

**ANALYSIS AND DESIGN FOR PROTECTING
THE
INTELLECTUAL PROPERTY RIGHTS
ASSOCIATED WITH MULTIMEDIA DATA**

A THESIS SUBMITTED

FOR THE AWARD OF THE DEGREE OF

Doctor of Philosophy

IN

TECHNOLOGY AND ENGINEERING

BY

SAMIR B. PATEL



**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING,
INSTITUTE OF TECHNOLOGY, NIRMA UNIVERSITY,
AHMEDABAD-382481,
GUJARAT, INDIA**

MAY, 2011.

Copyright by
Samir B. Patel
2011

Certificate

1. This is to certify that I, Samir B. Patel am registered as a student under Registration No. 05EXTPHDE01 for Doctoral programme under the Institute of Technology under the Nirma University. I have completed the research work and also the pre-synopsis seminar as prescribed for the Ph. D. study
2. It is further certified that the work embodied in this thesis is original and contains the independent investigations carried out by me. The research is leading to the discovery of new facts/techniques/correlation of scientific facts already known.

Date :

Signature of the student

Forwarded by guide :

Guide

Remarks of Head of the Department(if any):

Date :

Signature : _____

Remarks of Dean Faculty Concerned:

Date :

Signature : _____

Remarks of Dean Faculty of Doctoral Studies & Research (if any):

Date :

Signature : _____

To,

The Executive Registrar,

Nirma University

Certificate

This is to certify that the contents of this thesis entitled Analysis and Design for Protecting the Intellectual Property Rights Associated with Multimedia Data is the original research work of Samir B. Patel carried out under my supervision.

I further certify that the work has not been submitted either partly or fully to any other university or body - in quest of a degree, diploma or any other kind of academic award.

Date:

Place: Ahmedabad

Name of Guide : Dr. Shrikant N. Pradhan

Full Designation: Professor

Name of Institute: Institute of Technology,
Nirma University.

Candidate's Statement

The work included in this thesis entitled Analysis and Design for Protecting the Intellectual Property Rights Associated with Multimedia Data is an independent investigation carried out by me under the guidance and supervision of Dr. Shrikant N. Pradhan. In the thesis, references of the work done by others which have been used are cited at appropriate places.

I hereby declare that the work incorporated in the present thesis is original and has not been submitted to any other university or body - in quest of a degree, diploma or any other kind of academic award.

Date:

Place: Ahmedabad

Name of Student : Samir B. Patel

Reg. No. : 05EXTPHDE01

Brief Synopsis

Different authors are using different meanings for the word “watermark”, it is mostly agreed that the watermark is one, which is unnoticeably added to the cover object in order to convey the hidden data. It is the process of inserting information into another object/signal and can be termed as watermarking.

Watermarking (now-a-days) is mainly used for copy-protection and copyright-protection. Historically, watermarking has been used to send “sensitive” information hidden in another signal which must withstand a few intentional and unintentional attacks. One way to protect the copyright of digital image/video is to add an invisible structure to the image to identify the owner.

Copy protection attempts to find ways, which limits the access to copyrighted material and/or inhibit the copy process itself. Examples of copy protection include encrypted digital TV broadcast, access controls to copyrighted software through the use of license servers and technical copy protection mechanisms on the media.

Copyright protection inserts copyright information into the digital cover object without the perceptual loss in it. Whenever the copyright of a digital object is in query, this information is mined to identify the rightful owner. It is also possible to encode the identity of the original user along with the information of the copyright holder, which allows tracing of any illegal copies. Whereas copy protection seems to be difficult to implement, copyright protection protocols based on watermarking and strong cryptography are feasible. These have been the motivation for the present study and design of unique approach which includes data mining algorithm such as ID3 and image transformation technique like Discrete Cosine Transform (DCT), which is perhaps the first of its kind.

Watermarking has been increasingly recognized as a highly effective means of protecting the intellectual property rights associated with multimedia data. This inner detail describes information hiding methods for the DCT and data mining that is used to embed the message and to retrieve the message.

“To analyze and design a technique for intellectual property rights associated with multimedia data” The design and implementation of an encoder and decoder algorithm using the decision tree and transformation technique, which is robust and secure for performing digital watermarking has been designed and tested with steganalyzer, Image processing tool like Matlab and data mining tool such as Weka. For information hiding and recovery inside the cover image along with decision tree, discrete cosine transform and Arnold transforms are also used. The information stored at scattered places based on distributed hiding of information using decision rules, makes the algorithm more secure rather than the sequential or static one. The design approach is such that the information is recoverable even if cover images are damaged to a large extent. The design uses Arnold transform to provide an additional security along with the usage of keys in the design. During the design the attributes of the images remains unchanged. The robustness of the watermarked images, is evaluated with a steganalysis tool StegAlyzerAS version 3.2.

The design is scalable to have enough embedding capacity and speed. The usage of keys and the transformation techniques have made the design highly robust. The work is focused almost exclusively on the watermarking of digital images, however, most of these ideas could be applied to the watermarking in the digital video and audio domain as well.

Objectives and Methodology

1. To investigate the utility and effectiveness of the algorithms of Data mining and DCT technique.
2. To identify the image blocks which are most prominent within the cover image which satisfy the requirements for embedding.
3. To validate blocks and monitoring the block positions within the image and perform embedding within those blocks using decision tree mechanism.
4. Perform attack like signal processing operations, geometric operations like rotation, scaling, shearing, adding random local distortions and specialized attack based on knowledge of the method and obtain the result after such intentional/unintentional attacks.
5. To provide robustness against attacks like color transformations, cropping, image enhancement and restoration processing such as histogram equalization, salt and pepper noise, Gaussian noise, Poisson noise, gamma correction, etc. Applying filters like average, disk circular, Gaussian, motion blur, sharpen image, Laplacian, Prewitt and Sobel, Laplacian of Gaussian and other attacks including Random local distortions, addition of constants to an image, dithering, quantization, contrast stretching, etc.
6. To have very high security in the algorithm by making using keys and transformation based techniques.
7. The approach followed in the design is scalable to identify the required number of blocks for performing embedding.
8. To provide multiple time watermarking in all the image planes.
9. The watermarked images must look similar to the original one, without noticeable degradation of the cover image.

10. Obtain the recovery of watermark through the decoding procedure.

DEDICATION

I would like to dedicate this thesis to my parents and members of my family.

ACKNOWLEDGEMENT

I would like to express my sincere gratitude to my guide Dr. Shrikant N. Pradhan, for his guidance, nurturing, encouragement and support in every stage of my doctoral studies and for his valuable supervision and support during the preparation of this thesis. His knowledge, kindness, patience, open-mindedness and vision have provided me with lifetime benefits.

I am grateful to Dr. Ketan Kotecha and Prof. D. J. Patel in my thesis committee for their valuable comments and suggestions on the thesis drafts, as well as for the experience either as an expert or as a head, with these two outstanding teachers, I would also like to thank all the faculty members of Nirma University for their academic guidance and encouragement.

During this research, I collaborated with different researchers working in the field of image processing - that provided the background and foundations for my thesis research. Special thanks to, Dr. Asim Banerjee (Prof. at DAIICT, Gandhinagar), Dr. K. B. Kartikeyan (Sr. Scientist, ISRO, Ahmedabad) for their fruitful suggestions as member of Research Progress Committee and Dr. Prabhat Ranjan (Prof. at DAIICT, Gandhinagar) for their guidance in several matters that have expanded my vision. Beyond technical matters, I greatly appreciate the experience with Prof. Shrikant Pradhan in his High-tech classes and other faculty members of Nirma University.

The productive discussions that I had with them regarding data hiding have nurtured me in the course of becoming a professional researcher. Special thanks to Dr. Ketan Kotecha and Dr. Shrikant Pradhan for the valuable discussions, collaborations, encouragement, friendship and helps since the very beginning of work. I would like to thank Mrs. Daxa Vasoya for her unended support for formatting this thesis.

I am so grateful to Heena my wife, for her love and encouragement during my journey of this thesis. I also want to thank my family members Master Nihar and Master Kirtan for the year's care and support received from them.

In addition to the thanks to my teachers and classmates over the past two decades, I must mention two persons without whose love, nurturing and support, I could never accomplish all these. Thank Mom (Late) for her care and encouragement toward the pursuit of excellence. And thank Dad, not only for his patience and sacrifice, but also for being such a great role model for his son.

- Samir B. Patel

Contents

Certificate	iii
Candidate's Statement	v
Brief Synopsis	vi
Dedication	x
Acknowledgement	xi
List of Tables	xvii
List of Figures	xix
1 Introduction	1
1.1 Scope	1
1.2 Objective	3
1.3 The Need	3
1.4 Introduction to Steganography	5
1.4.1 LSB Technique	6
1.5 Introduction to Digital Watermarking	6
1.6 Introduction to Cryptography	10
1.7 Introduction to Steganalysis	11
1.8 Introduction to Compression	13
1.9 Developmental Challenges	13
1.10 Applications	14
1.11 Document Overview	16
Abbreviations	1
2 Intellectual Property Rights	18
2.1 Intellectual Property Rights (IPR)	18
2.1.1 Copyright	19
2.1.2 Patents	20

2.1.3	Trade Secrets	20
2.2	Introduction to DRM	21
2.2.1	Encryption	21
2.2.2	Watermarking	22
2.2.3	Other Limitations	23
2.3	Digital Watermarking and Steganography as Part of IPR	24
2.4	Current Scenario	25
2.5	Applications	25
3	Classification and Design Issues	27
3.1	Basis for Classification	27
3.2	Watermarking Design Issues	30
3.2.1	Other Properties	33
3.3	Introduction to Data Mining and its Related Concepts	36
3.4	The ID3	38
3.5	Summary	40
4	Transformation Based Techniques	41
4.1	Transformation Based Techniques for Performing Digital Watermarking DCT/DWT	41
4.2	Embedding Using DCT & DWT Transformation Based Techniques	45
4.2.1	Embedding Algorithm for MBCX (Mid Band Coefficient Exchange Method)	48
4.2.2	Recovery Algorithm for MBCX (Mid Band Coefficient Exchange Method)	50
4.2.3	DWT Based Embedding Technique	51
4.2.4	DWT Based Recovery Technique	52
4.2.5	Implementation Results of DCT Based Method	53
4.2.6	Implementation Results of DWT - Based Method of 1 Scale and 2 Dimensions	53
4.2.7	Implementation Results of DWT - Based Method of 2 Scale and 2 Dimensions	54
4.3	Summary	55
5	Watermarking Using Decision Tree and DCT	57
5.1	Mathematical Model	58
5.1.1	Training Before Actual Identification of Blocks	58
5.1.2	Actual Embedding of The Watermark Within the Host Image	62
5.1.3	Recovery of the Watermark from the Watermarked Image	64
5.2	Determining the Decision Tree	65
5.2.1	Data Pre-Processing and Training for Identification of Blocks	66
5.2.2	Algorithmic Details for The Application of ID3 in Digital Image Watermarking	73

5.2.3	Introduction to Arnold Transform	75
5.2.4	Embedding Algorithm	76
5.2.5	Extraction Algorithm	78
5.3	Comparison and Contribution	80
5.4	Summary	84
6	Experimental Results	85
6.1	Results from Datamining Approach	85
6.2	General Results and Discussion	87
6.3	Test for Robustness Against Various Image Processing Operations . .	98
6.4	Geometric Attack	104
6.4.1	Attack against Rotation	104
6.4.2	Attack against Scaling	104
6.4.3	Attack Against Shearing	105
6.4.4	Random Local Distortion	105
6.5	Signal Processing Operations	105
6.5.1	Attack against JPEG Lossy Compression	106
6.5.2	Averaging Filter	106
6.5.3	Disk Circular Averaging Filter	106
6.5.4	Gaussian Filter	107
6.5.5	Motion Blur Filter	107
6.5.6	Sharpen Image Filter	108
6.5.7	Laplacian Filter	108
6.5.8	Prewitt and Sobel Filter	108
6.5.9	Laplacian of Gaussian Filter	109
6.5.10	Dithering of pixels	109
6.5.11	Uniform Quantization	109
6.5.12	Minimum Variance Quantization	109
6.6	Specialized Attack based on Knowledge of method	110
6.6.1	Contrast Stretching	110
6.6.2	Addition of Constant to an Image	110
6.7	Storage of Keys	129
6.8	StegAlyzer AS 3.2 Evidence Report	133
7	Conclusions and Future Work	136
7.1	Conclusions	136
7.2	Future Work	139
A	Publications	140
A.1	Complete List of Publications	140
A.2	Publications with Abstract	143

B Prerequisite	148
B.1 DCT Basics	148
B.2 DWT : Discrete Wavelet Transform	149
B.3 PSNR	152
C Test Images	153
References	158
Index	174

List of Tables

2.1	Analysis of IPR	19
4.1	List of block based DCT algorithms without any perceptual modeling	42
4.2	List of block based DCT algorithms using implicit perceptual modeling	43
4.3	List of block based DCT algorithms using explicit perceptual modelling	43
4.4	List of DWT based non-blind watermarking algorithms	44
4.5	List of DWT based blind watermarking algorithms	45
4.6	Quantization values used in JPEG compression scheme	46
4.7	DCT results for mid band	53
4.8	DWT results for 1-Scale and 2-Dimensions	53
4.9	DWT results for 2-Scale and 2-Dimensions	54
5.1	Generalized range of attributes for selecting the class	68
5.2	Sample records with low frequency coefficient, PSNR and target class	68
5.3	Sample records with target class as ‘A’ and ‘N’	73
5.4	Comparison of work done by different authors with our algorithm . .	81
6.1	Summary of results (Output from Weka)	86
6.2	Detailed accuracy by class	86
6.3	Confusion matrix	86
6.4	PSNR values obtained in the RGB planes in color images after water- marking	89
6.5	Watermarking block where embedding is performed in the RGB planes	91
6.6	Recovered watermark from the cartoon image in RGB plane	91
6.7	Cropped and recovered watermarked images from Blue plane	92
6.8	Blocks identified as ‘A’ category blocks against different DC strength of test images in red plane.	93
6.9	Blocks identified against different DC strength in Bridge images in Red, Green and Blue image planes.	93
6.10	PSNR values obtained after addition of noise in watermarked images	99
6.11	PSNR values obtained after addition of noise in watermarked images	100
6.12	PSNR values obtained against various parameter values for gamma correction	101

6.13 PSNR values obtained against various parameter values for gamma correction	102
6.14 Average filter mask	106
6.15 Gaussian filter mask with $\sigma = 0.1$	107
6.16 Gaussian filter mask with $\sigma = 0.5$	107
6.17 Gaussian filter mask with $\sigma = 0.9$	107

List of Figures

1.1	Overall Block Diagram for Steganography and Digital Watermarking [1]	7
1.2	Classification of watermarking algorithms based on domain used for embedding Process	17
3.1	Classification of watermarking algorithms	29
4.1	Definition of DCT regions	46
4.2	2 Scale 2-Dimensional discrete wavelet transform	48
4.3	Watermarked images using DWT with $k = 0.5$	54
4.4	Recovered watermark images from LENA image	55
4.5	Recovered watermark images from MOON image	55
4.6	Recovered watermark images from CAMERAMAN image	55
4.7	Watermarked images using DWT with $k = 5$	56
4.8	Recovered watermarked images for LENA, MOON & CAMERAMAN using DWT for 2 Scale 2 Dimension with $k = 5$	56
5.1	A major activities performed by the functions to perform digital watermarking	58
5.2	A partial decision tree	67
5.3	Plot of <i>DC</i> coefficients	69
5.4	Plot of <i>AC1</i> coefficients	69
5.5	Plot of <i>AC2</i> coefficients	70
5.6	Plot of <i>AC3</i> coefficients	71
5.7	Plot of <i>AC4</i> coefficients	71
5.8	Training and identification of blocks	75
5.9	Arnold images after various iterations	76
5.10	Block diagram for embedding of the watermark	78
6.1	Gray scale Lena and Barbara images	87
6.2	Watermarked Lena and Barbara images	87
6.3	Extracted watermark	88
6.4	Extracted watermark with different JPEG quality factor	88
6.5	Correlation graph: Correlation Vs JPEG Quality factor	88

6.6	Graphical representation of the PSNR values obtained during watermarking in different image planes	90
6.7	Color logo watermark	90
6.8	Sample watermarked images of cartoon image in RGB plane	92
6.9	Blocks identified in image plane(s)	94
6.10	Double watermarked Bridge image in red plane	96
6.11	Recovered second watermark from Figure 6.10 using key set-2	97
6.12	Recovered first watermark from Figure 6.10 using key set-1	97
6.13	Plot of PSNR value after gamma correction	103
6.14	PSNR and recovered watermark after geometric rotation of varying degree	111
6.15	PSNR and recovered watermark after geometric scaling of different sizes	112
6.16	PSNR and recovered watermark after geometrically shearing in horizontal, vertical and both sides	113
6.17	PSNR and recovered watermark after intentionally added random local distortions	114
6.18	PSNR values for different images with varying quality factor (Q) and the recovered watermarks for JPEG compression attack	115
6.19	Averaging filter attack with PSNR and recovered watermark images .	116
6.20	Disk circular averaging filter attack with PSNR and recovered watermark images	117
6.21	Attack of Gaussian filter on watermarked images with PSNR and recovered watermarks	118
6.22	Motion blur attack with varying values of length and θ along with PSNR and recovered watermarks	119
6.23	Contrast enhancement attack with varying α along with PSNR and recovered watermarks	120
6.24	Laplacian filter with varying α along with PSNR and recovered watermarks	121
6.25	Prewitt and Sobel filter along with PSNR and recovered watermarks .	122
6.26	Laplacian of Gaussian filter with varying σ along with PSNR and recovered watermarks	123
6.27	Dithering and Nodithering on watermarked images with different colors with its PSNR and recovered watermarks	124
6.28	Uniform quantization with varying tolerance with its PSNR and recovered watermarks	125
6.29	Minimum variance quantization with different colors cubes with the PSNR and recovered watermarks	126
6.30	PSNR and recovered watermark after contrast stretching	127
6.31	PSNR and recovered watermark after constant value addition	128
6.32	Embedding location	129
6.33	Upper right location for storage of keys within the image	130
6.34	Bottom left location for storage of keys within the image	130

6.35	Bottom right location for storage of keys within the image	131
6.36	Simple LSB way to embed the keys into the pixel of R, G and B image plane.	131
6.37	User screen to store keys within the image.	132
6.38	Screen after selection of points where storage will take place.	132
6.39	Double steganography within the same image.	133
B.1	2D DWT decomposition	150
B.2	Decomposition of Image	151
C.1	Lena Gray Scale [2]	153
C.2	Barbara Gray Scale [3]	154
C.3	Lena Color [4]	154
C.4	Sea Shore Color	155
C.5	Cartoon Color Image [5]	155
C.6	Sail Ship Color [6]	156
C.7	Bridge Color Image [6]	156
C.8	Sail Boat Color Image [7]	157
C.9	Watermark Image	157

Chapter 1

Introduction

The amount of data generated by the computer professionals is increasing manifold. With the increase in the communication technologies, the users using these multimedia data have also increased. The development of internet has played a vital role in the distribution of multimedia data. The owner of the multimedia data, most of the time is not aware of the usage of data by others and sometimes the user using it is not knowing the legal aspects of it. There is a need to protect the individuals digital property by some mechanism. The need to solve the copyright protection issues by making use of Digital watermarking as a technique, is attempted in this chapter. In this Chapter, Section 1.1 provides the scope of the work involved and the objective is discussed in Section 1.2. A brief introduction to data hiding and its related concepts are also examined in this Chapter.

1.1 Scope

Digital watermarking addresses issues related to intellectual property and copyright protection. Digital watermarks can be considered as commercial applications of steganography. To trace, identify and locate digital media across networks digital watermarking and steganography is used. Digital watermarks are the additional data send with the carrier, as a watermark typically includes information about the owner

[N.F. Jhonson et al.][8] to protect the rights of the owner for the multimedia data. There has been a considerable work done by different authors using various transformation based techniques in this field. The requirement is such that, even if the intruder or user, applies few transformation to the data, the owner can still prove his/her ownership. The work focuses in developing a mechanism, through which concealment of data within multimedia content is made feasible. The design of a novel approach to perform digital watermarking is attempted using the concept of data mining which includes the usage of decision tree. The decision tree is prepared using the data mining algorithm such as ID3. The training is provided within this approach before the actual embedding is performed. The key elements in the design of data hiding system includes the following points to be considered.

- A model that ensures imperceptibility.
- Embedding of data in multiple planes of the image and multiple times within the image plane.
- Embedding in those parts of the cover media, where embedding is difficult and to handle uneven embedding capacity.
- The approach, to be such that it is robust to intentional and unintentional attacks.
- To embed data of any type.
- Recovery of the message on the receiver side.
- To check for the noise free recovery on the receiver side.
- Usage of keys along with transformations to make the system highly secure.
- To design a system which is scalable for embedding data of different size.

1.2 Objective

As digital watermarking is an emerging copyright protection mechanism. The objective is to provide a novel digital watermarking technique, which provides all the ideal watermarking characteristics in the design. In the design all features of the ideal watermarking system has to be integrated. Once the requirement analysis is done the choice of watermarking technique is to be focused. As part of the design technique a new robust watermarking technique based on combining the power of transform domain technique; the Discrete Cosine Transforms (DCT) and the data mining technique such as decision tree induction (ID3) is to be evolved. Further the work is to focus on the transformed vectors to build the decision tree. The training of images is provided for deriving the classification tree. The resulting decision tree provides the decision making rules to identify good quality image blocks for insertion of watermark. The major objective of the work is to design an ideal system to have robustness, security, capacity, speed, multiple watermark embedding capacity, etc. Watermarking, till now is not performed using data mining and transformation based technique[Patel et al.][9] and hence the approach is novel of its kind.

1.3 The Need

With the proliferation of digital media such as images, audio and video, a robust digital watermarking techniques for data hiding are needed for copyright protection, copy control, annotation, and authentication of multimedia data

A well-known term that “seeing is believing” is no longer true due to the pervasive and powerful multimedia manipulation tools. Such development has decreased the trustworthiness of multimedia data such as photos, video or audio clips, printed documents, and so forth used earlier. To ensure this trustworthiness, multimedia authentication techniques are being developed to protect multimedia data by verifying the information integrity, the suspected source of data and the reality of data.

As information is becoming widely available through internet. These global networks allow cross-references between databases. The advent of multimedia is allowing different applications to mix sound, images, and video and to interact with large amounts of information (e.g., in electronic business, distance education and human-machine interface). The industries provides audio, image and video data in electronic form to customers. The broadcast television companies, major corporations and photo archivers are converting their content from analogue to digital form. This movement from traditional content, such as paper documents, analogue recordings to digital media is due to several advantages of digital media over the traditional media. Some of these advantages are:

1. The trustworthiness of digital data is higher than that of their corresponding analog data. As time passes the traditional assets degrades in quality. Analogue data require high-priced systems to obtain good quality copies, whereas digital data can be easily copied without loss.
2. Digital data (audio, image and video data/signals) can be easily transmitted over networks such as internet. A large amount of multimedia data is easily available to users across the world. Such expansion will continue to improve at higher rate with the widening availability of advanced multimedia services like electronic commerce, advertising, interactive TV, digital libraries and a lot more.
3. Accurate copies of digital data can be made easily. This is very useful, but it also creates inconvenience to the owner of valuable digital data like precious digital images. Replicas of a given piece of digital data cannot be separated and their origin cannot be proved. It is impossible to determine which piece is the original and which is duplicate.
4. It is possible to hide some information within digital data such that the modifications within the data cannot be observed by the human senses.

1.4 Introduction to Steganography

Figure 1.1 provides the block diagram for performing steganography and digital watermarking. Frequently evolving technologies and expansion in communication system have made data accessibility faster.

Steganography in real sense is to perform hiding of data and transmitting it in such a way that the existence of the hidden data is not revealed. The word steganography is a Greek word and it means to cover. Computer based techniques have been designed to embed data within the cover object. This information can be communicated in the form of binary files, text or to provide additional information about the cover and its owner such as digital watermarks or fingerprints.

Criminals, who seek ways to conceal their activities in cyberspace, use data hiding techniques to distribute child pornography, steal intellectual property and converse covertly with the conspirators.

The most popular technique for performing steganography is Least Significant Bits (LSB) modifications. Within the image each pixel consist of 3 planes of 8 bit each, this image planes could be used to perform least significant bit insertion to store the hidden messages.

However this LSB technique is not robust, since a forceful deletion at these LSB positions could remove all the stored hidden data. The only advantage of this technique is that, it is very fast and easiest way for performing the embedding of messages. The embedding capacity of LSB technique is very high. But as said earlier, one can remove the content by brute force approach, which is a major concern.

1.4.1 LSB Technique

LSB insertion is one of the simplest techniques and has been in use since long (1980s). Given an image, the lower order bits are replaced with meaningful data. Since only the lower-order bits are altered, the resulting color shifts are typically invisible [Shoe-maker C.][10].

Steganography is a related field in which an attempt is made to attach information covertly, such that concealed information is not only ciphered or encrypted, but also not noticeable. The two techniques steganography and watermarking are almost similar but with slight difference. Watermarking only demands that the hidden data is hardly noticeable to the human eyes; based on the statistical analysis performed at the decoder side, the presence of the watermark can be known. On the other side steganography assumes that the end user is an opposition who may attempt to change or remove the hidden message. In most applications, the end user may be aware of the existence of a watermark without compromising its usefulness. A few steganography tools [11] have been designed by different authors on the basis of their needs. Encryption and compression can be performed before embedding the message into the cover object. However, both the encryption and compression are optional.

1.5 Introduction to Digital Watermarking

Different authors are using different meanings for the word ‘watermark’, it is mostly agreed that the watermark is one, which is unnoticeably added to the cover-signal, in-order to convey the hidden data. The process of embedding information within another object/signal is termed as watermarking.

Watermarking is mainly used for copy-protection and copyright-protection. It is observed historically that the watermarking systems have been used to send “sensitive” information hidden within another object/signal. Watermarking has its major

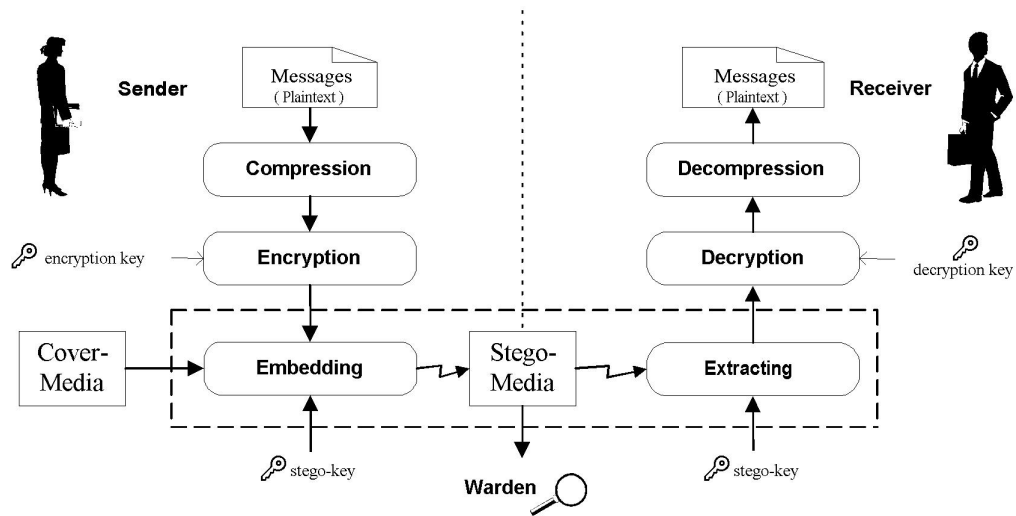


Figure 1.1: Overall Block Diagram for Steganography and Digital Watermarking [1]

applications in image/video copyright protection.

Copy protection attempts to find ways, which limits the access to copyrighted material and/or inhibit the copy process itself. Examples of copy protection include encrypted digital TV broadcast, access controls to copyrighted software through the use of license servers and technical copy protection mechanisms on the media. A recent example is the copy protection mechanism on DVDs. However, recent incidents show that the copy protection is very difficult to achieve.

Copyright protection inserts copyright information into the digital object without losing the quality of the object. Whenever the copyright of a digital object is in question, this hidden information is extracted to identify the rightful owner. It is also possible to encode the identity of the original buyer along with the identity of the copyright holder, which allows identification of the unauthorized copies.

Whereas copy protection seems to be difficult to implement, copyright protection protocols based on watermarking and strong cryptography are feasible.

The characteristics of watermarking algorithm are normally followed differently with the different types of application for which it is designed. The following part explain the words used in the context of watermarking.

Imperceptibility: In watermarking, we traditionally seek high quality, i.e. the watermarked work must look or sound, similar to the original.

Robustness: It is more a property and not a requirement of watermarking. The watermark should be able to survive any reasonable processing inflicted on the carrier.

Security: The watermarked image should not reveal any clues of the presence of the watermark, with respect to un-authorized detection, or (statistical) undetectability or unsuspecting (not the same as imperceptibility).

Different Types of Watermarks

Even though this Chapter does not relate to all kinds of watermarks that will be defined, it is important to state their existence in order to later derive some of the possible applications of watermarking systems.

- Robust watermarks are simply watermarks that are robust against attacks. Even if the existence of the watermark is known, it should be difficult for an attacker to destroy the embedded information without the knowledge of the keys. An implication of this fact is that the amount of data that can be embedded (also known as the payload) is usually smaller than in the case of steganographic methods. It is important to say that watermarking and steganographic methods are more complementary than competitive.
- Fragile watermarks are marks that have only very limited robustness [Kutter et al.][12]. They are used to detect modifications of the cover data, rather than

convey inerasable information, and usually become invalid after the slightest modification of a work. Fragility can be an advantage for authentication purposes. If a very fragile mark is detected intact in a work, we can infer that the work has probably not been altered since the watermark was embedded [Cox et al.][13]. Furthermore, even semi-fragile watermarks can help localize the exact location where the tampering of the cover work occurred.

- Perceptible watermarks, as the name states, are those that are easily perceived by the user. Although they are usually applied to images (as visual patterns or logos), it is not uncommon to have an audible signal overlaid on top of a musical work, in order to discourage illegal copying. As an example, the IBM Digital Libraries project [Memon et al.][14] [Braudaway et al.][15] has developed a visible watermark that modifies the brightness of an image based on the watermark data and a secret key. Even though perceptible watermarks are important for some special applications, our focus will be on imperceptible watermarks, as they are the most common.
- Bitstream watermarks are marks embedded directly into compressed audio (or video) material. This can be advantageous in environments where compressed bitstreams are stored in order to save disk space, like Internet music providers.
- Fingerprinting and labeling denote special applications of watermarks. They relate to watermarking applications where information such as the creator or recipient of the data is used to form the watermark. In the case of fingerprinting, this information consists of a unique code that uniquely identifies the recipient, and that can help to locate the source of a leak in confidential information. In the case of labeling, the information embedded is a unique data identifier, of interest for purposes such as library retrieving.

1.6 Introduction to Cryptography

While cryptography is about protecting the content of messages (their actual meaning), steganography is about concealing their very existence. It is often thought that communications may be secured by encrypting the traffic, but this has rarely been adequate. In the cryptography the encrypting algorithm is applied to transform the data into another domain. This data is inverse transformed on the receiver side to get back to the original one. For doing this, the keys are required to be made available on the receiver side, where as for doing compression of the data, we do not require keys. The compression techniques could also be used for performing encryption. However, by both of this techniques the data is converted into different domain which is not readily understood by the human being. A specialized software is required for retrieving the useful information. Many tools and techniques [D. Salomon][16] are available and are in use today.

Many authors and other classical writers still concentrate on methods for hiding messages rather than for enciphering them because it arouses less suspicion. This preference persists in many operational contexts to this day. For example, an encrypted email message between a known drug dealer and somebody not yet under suspicion, or between an employee of a defence contractor and the embassy of a hostile power, has obvious implications.

As the purpose of steganography is having a covert communication between two parties whose existence is unknown to a possible attacker, a successful attack[Patel et al.][17] consists in detecting the existence of this communication (e.g., using statistical analysis of images with and without hidden information). Watermarking, as opposed to steganography, has the (additional) requirement of robustness against possible attacks.

Copyright marks do not always need to be hidden, as some systems use visible digital watermarks, but most of the literature has focused on imperceptible (e.g., invisible, inaudible) digital watermarks which have many applications. Visible digital watermarks are strongly linked to the original paper watermarks which appeared at the end of the XIII century to differentiate paper makers of that time. Recent visible watermarks may be visual patterns (e.g., a company logo or copyright sign) overlaid on digital images and over televisions. The intent of use is also different: the payload of a watermark can be perceived as an attribute of the cover-signal (e.g., copyright information, license, ownership, etc.). In most cases the information hidden using steganographic techniques is not related at all to the cover. These differences in goal, leads to very different hiding techniques [Arnold et al.][18].

1.7 Introduction to Steganalysis

Several digital steganography applications are readily available on internet, and most of these are available as freeware or shareware, for use by criminals and terrorists. Computer security, law enforcement and intelligence professionals need the capability to both, detect the use of digital steganography applications to hide information and then extract the hidden information. Accordingly, there is much current interest in steganalysis, or the detection and extraction of embedded information. Steganography analysis and research center (SARC) has came up with such a tool such as StegAlyzerAS 3.2 to engage both of these steganalysis [11] techniques to effectively detect the use of digital steganography applications and subsequently extract the hidden information from it.

Blind Steganography Detection

The blind detection approach to steganalysis has been over the years. Blind detection attempts to determine if a message may be hidden in a file without any prior knowledge of the specific steganography application used to hide the information.

Several techniques may be employed to inspect, suspect files including various visual, structural and statistical methods.

Visual analysis methods, attempts to perceive the presence of steganography through visual inspection, either with the naked eye or using automated processes of the systems. Visual inspection with the naked eye can succeed, when steganography is inserted in relatively smooth areas, with nearly equal pixel values. Automated computer processes can, for example, decompose an image into its individual bit planes. A bit plane consists of a single bit of memory for each pixel in an image, and is a typical storage place for information hidden by steganography applications. Any unusual appearance in the display of the LSB-plane would be expected to indicate the existence of steganography.

Structural analysis methods attempt to reveal alterations in the format of the data file. For example, a steganography application may append hidden information past an image's end-of-file marker. An image that has been modified using this appending technique is interpreted by the operating system just as if, it were the original carrier file. The two files are visually and digitally identical, because the image's data bits have not been altered. The hidden information that is embedded past the end-of-file marker is simply ignored by the operating system. Many automated methods for conducting structural analysis have been developed in addition to the manual process of investigating images with a hex editor.

Statistical analysis methods attempt to detect tiny alterations in a file's statistical behavior caused by steganographic embedding. Statistical analysis of files can be difficult and time consuming, since there are a many ways for embedding- each modifying the carrier file in a various way. Therefore, unified techniques for detecting steganography using this method are hard to find. Determining statistics such as means, variances, and chi-square tests can measure the amount of redundant infor-

mation and/or deviation from the expected file characteristic. Current research in blind detection steganalysis is focused on these statistical methods.

1.8 Introduction to Compression

Data Compression is the process of encoding information using fewer bits than an unencoded representation would use. Compression is used to reduce the consumption of expensive resources, such as hard disk space or transmission bandwidth. On the sender side the encoder would convert the original data to a reduced size and on the receiver side the decoder would extract the meaningful information from it. When the data is encoded and converted into compressed domain, in one way, the data is made secure. If steganographic applications are using such type of compressed data, one level of security will be automatically provided into it. Also, if the volume of data is high, compressing the data will help to reduce the embedding capacity of the medium, resulting into computing efficiency during retrieval on the receiver side. A technique based on steganography, cryptography and compression could be obtained at [Patel et al.][19].

1.9 Developmental Challenges

Watermarking technology has become increasingly important as more vendors wish to sell their digital works on the Internet. This includes all manners of digital data including books, images, music and movies. Last few years has seen tremendous development in communication system and design systems of watermarking. However, despite this development and improvement in the digital image watermarking field, current technologies are far from what the end user is expecting. Lack of standardization and lack of a set of precise and realistic requirements for watermarking systems are two aspects that hinder further developments of copy protection mechanisms and digital watermarking techniques. Also, the lack of agreement on the definition of a

common benchmark for method comparison and on the definition of the performance related concept is the third aspect for this hindering.

- Perceptual transparency
- Security
- Robustness
- Capacity
- Speed
- Statistical invisibility
- Scalability

are some of the developmental challenges which will be discussed in Chapter 3.

1.10 Applications

Watermarking is not restricted to just retaining information of the author in the work, there are various other purposes for which watermarking may be incorporated into an object. Some of them are:

Copyright Protection: For the protection of intellectual property, the data owner can embed owner's signature as watermark representing the copyright information in his data.

Fingerprinting: To trace the source of illegal copies, the owner can use a fingerprinting technique. This requires the owner to embed different information onto copied of the work provided to different customers. The information embedded can be a serial number, customer id etc.

Copy protection: The watermark represents a single copy prohibit bit and the watermark detectors in the recording device determine whether the data offered to the recorder can be stored or not.

Broadcast Monitoring: By embedding watermarks in the commercials, an automated monitoring system can determine whether the commercial was broadcasted or not. Also other TV programs which might represent significant intellectual property such as the News.

Data Authentication: Introducing fragile watermarks into the data can help ensure that the data is not processed or modified in anyway by the user.

Indexing: Introducing watermarks in video mail, movies, news items can be used to index the data.

Data Hiding: Watermarking may be used to embed longer bits of information in the data. The earliest form of this was in ancient Greece, where an author could hide his name in the text of the literary work. The term used to describe data hiding, “Steganography” originated in Greece. This was also used by the Germans/Allies in World War II to send sensitive information to outposts by hiding it in postcards.

Medical Safety: Watermarks containing the name of the patient can be embedded onto the X-Rays, MRI Scans & other test results help in instant identification of the result as belonging to a patient and thus avoid mix-ups which can lead to catastrophic consequences.

Watermarking has been proposed in the literature as a means for different applications. The main digital watermarking applications for covert communications include:

1. Even though it contradicts the definition of watermark given before, some people may use watermarking systems in order to hide data and communicate

secretly. This is actually the realm of steganography rather than watermarking, but many times the boundaries between these two disciplines have been blurred. Nonetheless, in the context of this Chapter, the hidden message is not a watermark but rather a robust covert communication.

2. The use of watermarks for hidden annotation [Burgett et al.][20], or labeling, constitutes a different case, where watermarks are used to create hidden labels and annotations in content such as medical imagery or geographic maps and indexes in multimedia content for retrieval purposes.
3. In these cases, the watermark requirements are specific to the actual media where the watermark will be embedded. Using a watermark that distorts a patient's radiography can have serious legal consequences, while the recovery speed is crucial in multimedia retrieval.

Figure 1.2 shows the applications of watermarking with some examples for each of these applications. Also, digital watermarking is proposed for tracing images in the event of their illicit redistribution. The need for this has arisen because modern digital networks make large-scale dissemination simple and inexpensive. In the past, infringement of copyrighted documents were often limited by the nonfeasibility of large-scale photocopying and distribution. In principle, digital watermarking makes it possible to uniquely mark each image distributed or sold to rightful customer. If a purchaser then makes an illicit copy, the illicit duplication may be convincingly demonstrated [Swanson et al.][21].

1.11 Document Overview

The next Chapter (Chapter 2) describes the Intellectual Property Rights found in literature with an emphasis on copyright protection, Patents and Trademark. The reader is also introduced to the DRM. Chapter 3 provides the classification and the measures to be taken care when performing a Digital Watermarking Techniques.

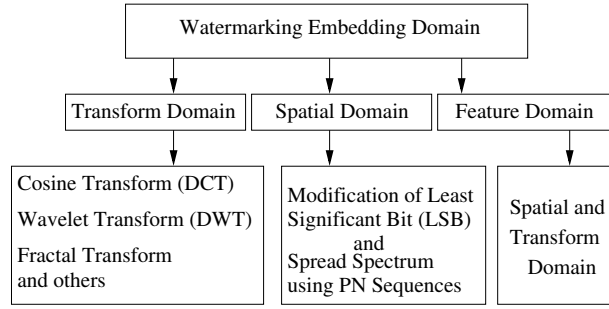


Figure 1.2: Classification of watermarking algorithms based on domain used for embedding Process

Chapter 4 describes the detailed analysis of the transformation based techniques and a few preliminary work done in this domain. Chapter 5 provides the mathematical model of the unified approach along with the detailed description of the newly designed algorithm. The topics that will be researched are also outlined along with the methodologies used to approach them. A detailed comparison and contribution of our work with other authors is explored in this chapter. The Chapter 6 gives detailed results of the topics that were investigated. Chapter 7 makes conclusions based on the results obtained and suggestions for future work. Appendix A provides a few publications details for the work done in this domain, Appendix B gives a few prerequisites for the approach and Appendix C shows all the original images used for testing.

Chapter 2

Intellectual Property Rights

Intellectual property (IP) is a term that refers to number of different creations of the human mind for which property rights are documented and the corresponding laws are formed for each one of them. This Chapter provides the basics of Intellectual property rights in Section 2. Section 2.1.1 discusses about the copyright. Patents and Trade Secrets are discussed in Section 2.1.2 and Section 2.1.3 respectively. An Introduction to DRM is explored in Section 2.2. Current scenario, applications of IPR in watermarking and steganography are explored in the subsequent Sections.

2.1 Intellectual Property Rights (IPR)

Under intellectual property law, proprietors are granted certain exclusive rights to a variety of intangible assets, such as musical, literary, and artistic works, discoveries and inventions, and words, phrases, symbols, and designs. Common types of intellectual property include copyrights [22][Pfleeger][23], trademarks, patents, industrial design rights and trade secrets. Table 2.1 provides a short summary of analysis of IPR.

Attributes	Copyright	Patent	Trade Secret
Protects	Expression of idea, not idea itself	Invention-the way something works	A secret, competitive advantage
Availability to Others	Yes, intention is to promote publication	Design filed at Patent Office and made available at a cost	No
Requirement to distribute	Yes	Saleable	No
Owner Ship	Very easy, do-it-yourself	Very complicated, specialist lawyer suggested	No filing
Duration	Life of human originator plus 70 years, or total of 95 years for a company	19 years	Indefinite
Legal protection	Sue if unauthorized copy sold (Law Suite)	Sue if invention copied (law Suite)	Sue for theft

Table 2.1: Analysis of IPR

2.1.1 Copyright

Copying a digital image is as easy as a mouse click so how do you protect the rights of the owner? According to the U.S. Copyright office [24]: Copyright is a form of protection provided by the laws of the United States (title 17, U. S. Code) to the authors of “original works of authorship”, including musical, artistic, literary, dramatic and certain other intellectual works. This protection is available to both published and unpublished works.

The copyright [23] in the work of authorship which immediately becomes the property of the author who created the work. Only the author or those deriving their rights

through the author can rightfully claim copyright and use it. In the case of works made for hire, the employer and not the employee is considered to be the author.

Anyone, who violates any of the exclusive rights of the copyright owner as provided by laws, is an infringer of the copyright.

An infringer of copyright is liable for either - (1) the copyright owner's actual damages and any additional profits of the infringer, or (2) statutory damages. So, how can the authors protect their creations? The answer to this is "Digital Watermarking".

2.1.2 Patents

Patents [23] are rights that are granted exclusively for inventions - inventions being a product or a process. The process itself must be a whole new approach or method to do something i.e. manufacturing, using or selling the patented product or from using the patented process, without due permission of the patent holder is a crime. Most countries have established patent offices and laws enacted to protect the rights of inventors over their inventions. After a maximum period of 18 months, the patent is granted, if approved. Trademarks and Copyrights as well as domain names are also to be registered at this office.

2.1.3 Trade Secrets

A trade secret [23] is information that gives one company, a competitive edge over others. For example, the formula for a soft drink is a trade secret, as is a mailing list of customer information about a product, due to be announced in a few months. Employees and outsiders who have access to the secret must be required not to divulge the secret. The owner must take precautions to protect the secret, such as storing it in a safe, encrypting it in a computer file, or making employees sign a statement that they will not disclose the secret. The trade secret protection evaporates in case of independent discovery. If someone else happens to discover the secret independently,

there is no infringement and trade secret rights are gone.

2.2 Introduction to DRM

The Internet is revolutionizing multimedia content distribution, offering users unprecedented opportunities to share digital images, audio, and video but it also presents severe challenges for digital rights management (DRM).

2.2.1 Encryption

As a fundamental information security technology, encryption is the process of scrambling confidential data into an unintelligible form. Digital content providers can apply various encryption techniques to protect the confidentiality of, and prevent unauthorized access to, digital content. Multimedia encryption, involves numerous technical complexities not encountered in encrypting text or other data. In addition, this approach has several limitations.

Lack of interoperability. Typically, different DRM systems employ their own encryption and rights management techniques. This makes interoperation of these systems difficult. Moreover, DRM system vendors might refuse to share their systems' inner workings or license their technologies, resulting in competing and incompatible systems. For example, Apple's FairPlay system is incompatible with Microsoft's Windows Media system.

Fair use and public availability restrictions. DRM restricts fair use rights. For example, users cannot transfer and play content protected by encryption-based DRM on arbitrary devices. Moreover, encryption-based DRM hampers public availability of multimedia content even after copyright expiration.

Deployment cost and complexity. Encryption-based DRM is most effective in a closed-content system. In large, open environments like the Internet, encryption-based content distribution requires the deployment of costly and

complex security mechanisms in a wide range of consumer devices. Furthermore, if an encryption system gets cracked, fixing the damage or upgrading the security infrastructure will incur additional cost.

2.2.2 Watermarking

A digital watermark [Shoemaker C.][10] [Johnson][25] is a signal embedded in multimedia content. In addition to being perceptually invisible or inaudible to humans, watermarks should be statistically undetectable and resistant to any malicious attempts to remove them [Natarajan][26]. In copyright protection applications, watermarks can carry information to assert the owner's copyright, licensing data for access control, or user-related information (such as a user's identity) to track illegal copy transfer. Researchers have developed different types of digital watermarks—for example, robust [Cox et al.][27], semifragile, and fragile—to assist in DRM, but the technology still faces several fundamental challenges.

Insufficient Robustness. Despite considerable efforts to develop watermarks resistant to content transformations such as JPEG compression, rotation, cropping, and additive noise, current watermarking techniques are not sufficiently robust for many DRM applications.

Inevitable Degradation of Quality. Multimedia content will inevitably degrade after watermarking. In general, it is easy to create either robust or imperceptible watermarks, but creating watermarks that have both qualities has proven to be quite difficult.

Incompleteness. Even robust watermarking technology cannot authenticate ownership of multimedia content on its own, as anyone can embed watermarks in the content. That is to say, a third-party content registration and authentication authority is needed.

2.2.3 Other Limitations

In addition to the problems unique to encryption and watermarking, both DRM solutions are vulnerable to the so-called analog hole—that is, protected digital content can be recorded and copied through analog means, then redigitized and distributed to bypass the protection systems.

However, perhaps the biggest impediment to proactive techniques like encryption and watermarking is not their robustness but their coverage. For example, the same video can be distributed via DVD, satellite and cable broadcasting, online streaming, or digital download, to name only a few ways. Using encryption-based DRM to protect content in all forms and channels is practically impossible. Indeed, Steve Jobs attributes FairPlay’s ineffectiveness to the coexistence of unprotected and protected music content. Similarly, most digital content on the Internet is not watermarked and therefore cannot be tracked or protected using this approach.

The limitations of both encryption and watermarking motivated the development of mediaprinting, a new DRM approach that attempts to retroactively protect copyrights by identifying multimedia content and checking whether it has been illegally distributed and shared on the Internet. Mediaprints are compact descriptors that, unlike extrinsic identifiers affixed to multimedia such as watermarks, or assigned identifiers such as International Standard Recording Code numbers for music, are extracted from the content. A mediaprint thus cannot be erased or faked because it can be always recomputed from the content. Unlike cryptographic hashes computed from binary data, which are extremely fragile and data-sensitive, mediaprints are robust (unchanging) across a wide range of modifications and transformations of the same content but sufficiently different for every unique content item.

2.3 Digital Watermarking and Steganography as Part of IPR

A digital watermark as you know is like inserting bits into a digital file - image, audio or video. Such messages more often carry copyright information of the file, control instructions or links to digital rights information, a unique content asset ID or a hyperlink relating the content to network services. Digital watermarking got its name from watermarking of paper or money. But the main difference between them is that digital watermarks are supposed to be perceptually invisible by the HVS (Human Visual System), unlike paper watermarks, which are visible. Digital image watermarking can be categorized into two main groups - visible and invisible watermarks.

A visible watermark is a visible semi-transparent text or image placed on top of the original image. It allows the original image to be viewed, but it still provides copyright protection by marking the image as its owner's property. Visible watermarks are more robust against image transformation and are preferable for strong copyright protection of intellectual property that's in digital format.

An invisible watermark is an embedded image which cannot be seen by human eyes. Only specialized software or electronic systems designed to extract the hidden information to establish the copyright owner [22] [28] [29][Pfleeger][23]. Invisible watermarking techniques are used to prove the digital contents legality.

Although the copyright protection is the main field of using digital watermarks, A few visible watermarking applications could be advertising (adding company's name and logo as a watermark for promotion rather than for protection) or even adding memo titles to digital photos, which cannot be removed from the image(s) or videos. It's obvious that only visible watermarks can satisfy these requirements. In many data communication systems, auxiliary data is typically transported with content and is

nearly always transparent to the user. For example, DVDs, CDs, Internet packets, wireless communications packets, and even telephone systems contain signalling, session and other control data. Digital watermarking is another method of distributing this auxiliary data with content, yet it is unique in that the auxiliary data is placed within the content itself rather than in an adjacent sideband channel such as the metadata that is commonly distributed with digital content. It persistently joins the auxiliary data to the content, and moves with it across channels. The fact that the data channel is created within the content allows the identification to survive content translation from one format or channel to another (e.g. DVD to HDTV, transitions from analog to digital, etc.). These characteristics are essential to effective content distribution and management, and carry features that benefit both the rights holder and the consumer [30].

2.4 Current Scenario

Copyright protection inserts copyright information into the digital media without affecting the PSNR, a measure of quality. Whenever the copyright of a digital object is in question, this information is extracted to identify the actual owner. It is also possible to hide the identity of the original buyer along with the identity of the copyright holder, which further allows tracing of any unauthorized copies and the most well-known way of embedding information in the multimedia data is by the use of digital watermarking [31] [Cox et al.][32]. Copyright protection protocols based on watermarking and strong cryptography are more feasible and easy to implement rather than copy protection process.

2.5 Applications

Watermarking (now-a-days) is essentially used for copy-protection and copyright protection. Here, copy protection tries to find ways, which restricts the right to use the

copyrighted material and/or restrains the copy process itself. Examples of copy protection include encrypted digital TV broadcasting, access controls to copyrighted software through the use of license servers and technical copy protection mechanisms by using encryption and decryption licenses in the transmission media, etc. There is a great demand to overcome the piracy issues of DVDs using copy protection mechanism.

Chapter 3

Classification and Design Issues

There are many watermarking techniques available in the literature. This Chapter reviews the important ones in the field of digital watermarking and steganography. This Chapter presents the analysis of work done by various authors in different domains and an evolution of a novel approach for performing digital watermarking in color images. Blind and Non blind are the types of techniques in which the complete analysis is categorized, outcomes and difficulties of this type of techniques are discussed in this Chapter. Introduction to data mining and concepts are explored in Section 3.3.

3.1 Basis for Classification

Watermarking [Vidyasagar et al.][33], which fit in to the information hiding area, has seen as research topic of great interest. There is a considerable amount of work being accomplished in this field. Steganography is used for secret communication with large amount of data to be communicated in less time [Johnson][25], whereas watermarking [Shoemaker C.][10] is used for content security, copyright management, content authentication and tamper detection. Every watermarking technique must ensure that embedding distortion is small enough to be imperceptible [Furht et al.][34]. There are a number of watermarking applications and they can be classified on the basis of

the type of information conveyed by the watermark [Nikolaidis et al.][35] are digital watermarking for copyright protection, copy protection, fingerprinting, content authentication, broadcast monitoring and system enhancement.

Different Types of image watermarking schemes can be classified as follows:

- By visibility - Visible vs. invisible
- By goals - Robust, Fragile, semi-fragile.
- By requirement of original for extraction - Blind or non-blind/Oblivious vs. non- oblivious/Public or private.
- By embedding process- Spatial domain, transform domain.
- By working region - Global or block-by-block.
- By characteristics - Image-adaptive, image-independent.
- By requirement of original for extraction - Oblivious and non-oblivious
Oblivious -Watermarking schemes that do not need the original image for watermark extraction are called oblivious. Non-oblivious – Watermarking schemes that need the original image for watermark extraction are called non-oblivious.
- By embedding - Spatial domain and Transform domain
Spatial domain techniques - Watermarking schemes that directly perform some transformation on the image pixels are called spatial domain watermarks or spatial watermarks.

Transform Domain Techniques - Watermarking schemes that transform the image in the frequency domain and then modify the transform coefficients are called transform domain techniques or spectral watermarks.

The Figure 3.1 below shows the classification of several image watermarking schemes under oblivious and non-oblivious techniques. A few transformation based techniques

are also available at [31][Shoemaker C][10] [S.C. Katzenbeisser et al.][36][37][J.R. Hernandez et al.][38][N. F. Johnson et al.][39][M.Kutter et al.][12][G. Langelaar et al.][40][P.Meerwald][41] [F. Petitcolas][42][D. J. Fleet et al.][43][A.H. Tewfik][44].

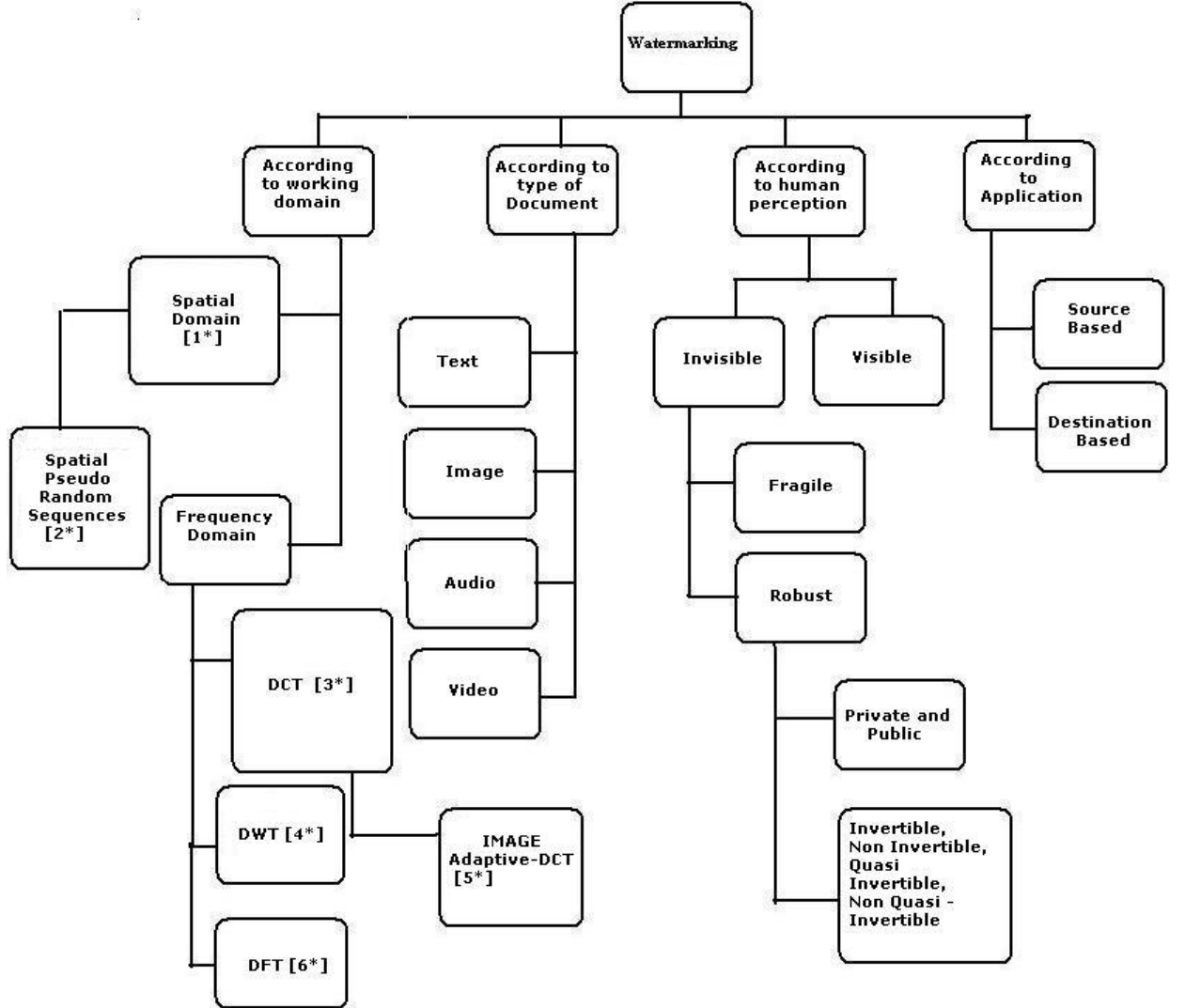


Figure 3.1: Classification of watermarking algorithms

Where,

[1*] = [45], [46], [43], [12], [47], [48], [49], [50], [26], [51], [52], [53]

[2*] = [54], [55], [56], [57]

$[3^*] = [58], [52], [59], [60], [61], [62], [63], [64], [65], [66], [67], [68], [69], [70]$

$[4^*] = [71], [72], [73], [74]$

$[5^*] = [68], [75], [76]$

$[6^*] = [77], [78]$

3.2 Watermarking Design Issues

There is a fundamental trade-off that restricts watermark designers. This fundamental trade-off exists among three key variables: robustness, payload & perceptibility [Miller et al.][79], [Czerwinski et al.][80], [N.F. Johnson et al.][8], [Kutter et al.][81], [Zhao et al.][82]. The relative significance given to each of these variables in a watermarking implementation depends on the desired application of the system. However, the fundamental characteristics remains the same for an ideal system.

Fundamental Characteristics

As per the review done by various authors [Arnold][83], [Boney, Tewfik & Hamdy][84], [Cox, Miller & Bloom][32], [Miller, Cox. & Linnartz][79], [Miller, Cox. & Bloom][13], [Kutter & Hartung][12], [Kutter & Petitcolas][81], [Swanson, Zhu, Tewfik & Boney][21] the ideal watermarking system should have the following features and these ideal features play a vital role in developing a quality test. The features are following:

Imperceptibility: Without perturbing the quality of the object, i.e. the embedding of the watermark should not be perceived by the end user. [Miller et al.][79]. In general, the term refers to the similarity between the original and watermark version of the covered object.

Perceptibility: The watermark must be imperceptible to the end user and [Miller et al.][79] also should imply that some sort of perceptibility criterion must be used not only to design the watermark, but also to quantify the distortion.

Moreover, it implies that this distortion can be measured at both the source and the destination. There is always a possibility of artifacts once the watermarking process is over and these produced artifacts are generally irritating and undesirable, which may lead to the reduction of the commercial value of the watermarked data in [S. Katzenbeisser] [36]. However, the perceptibility of the watermark can be added to when certain operations are performed on the cover signal.

Robustness: It is a kind of ability to perceive the watermark after common signal processing operations and unintentional attacks by the end user. A few common operations performed on multimedia data consists of addition of noise, volume adjustment or normalization, digital to analog conversion, histogram equalization, gamma correction, brighten or darken the color map, rotation, scaling and so forth [Rafael et al.] [85]. On the other hand, an attack is a process specifically targeted to remove the watermark. All watermarking applications do not need robustness against all possible signal processing operations. A serious consideration is required to be followed while designing the system. It is not compulsory to remove a watermark to make it useless; if the detector is unable to detect the presence of the mark, the attack can be considered as successful. It implies that a watermarking scheme is robust when it is able to withstand a series of attacks that attempts at the modification of the, quality of the embedded watermark, till it's completely removed, or its recovery process goes unsuccessful. "No such perfect method has been proposed so far, and it is not clear yet whether an absolutely secure watermarking method exists at all "[Kutter et al.][12] in [36].

The effective watermarking system produces the watermarked images after embedding. In other words, there is a probability that a watermark detector will recognize the watermark immediately after inserting it in the cover object.

What is most interesting in this definition is the implication that a watermarking system can have an effectiveness of less than 100%. That is, it is possible for a system to generate watermarks that are not fully recoverable even if no processing is done to the cover signal. This happens because perfect effectiveness comes at a very high cost with respect to other properties, such as perceptibility [Cox et al.][13]. When a known watermark is not successfully recovered by a detector, it is implied that a false negative error has occurred [S. Katzenbeisser][36].

Depending on the application, one may compromise some performances in place of other characteristics. For example, if an extreme level of fidelity is to be achieved, one may not successfully watermark certain type of works without creating some sort of distortion. In some cases, the effectiveness can be judged analytically, but most of the time it has to be evaluated by embedding a large chain of works with a given watermark and then trying to extract that mark. However, the statistical features of the test set must be the same when compared to those of the works that will be marked in the real world using the algorithm.

Data Payload: This term here refers to number of embedded bits that are transmitted in the cover object. A watermark that encodes N bits is considered as an N -bit watermark, and an attempt can be made to embed multiple watermarks of varying sizes. It is evident that there is a huge difference between the encoded message m , and the real bitstream that is embedded in the cover object. The latter is normally named as a pseudorandom (PN) sequence. Many systems are proposed where only one possible watermark has an opportunity to be embedded. The detector then tries to decide whether the watermark is available or not. These systems are called as one-bit watermarks, as only two diverse values can be encoded inside the watermark message.

While analyzing the data payload of a watermarking system, it is also necessary to differentiate the number of distinct watermarks that may be inserted, and the number of watermarks that may be detected by a single iteration with a given watermark detector. In many watermarking applications, each detector need not be tested for all the watermarks that might possibly be available [Cox et al.][79]. For example, one may insert two different watermarks into the same cover object, but the interest may be in recovering only in the last one which is embedded.

3.2.1 Other Properties

Some of the characteristics reviewed in the literature may not be crucial for testing purposes; however, they must be mentioned, so as to make the analysis of watermarking systems as comprehensive as possible.

False positive rate: A false positive error is the detection of a watermark in a work that does not actually contain one. Thus, a false positive rate is the expected number of false positives in a given number of runs of the watermark detector. Similarly the probability can be detected that a false positive will occur in a given detector run. In some applications a false positive can be highly dangerous & devastating. An example, of a DVD player that incorrectly determines that a legal copy of a disk (for example a homemade movie) is a non-factory-recorded disk and therefore refuses to play it. If such error becomes common, then the reputation of DVD players can be at stake and consequently their market can be seriously affected.

Statistical invisibility: This is required to prevent unauthorized detection or removal. Performing statistical tests on a set of watermarked files should not give any fact about the embedded information, nor about the technique used for watermarking [Swanson et al.][21]. The work by [Johnson et al.][8] provide a long and detailed description of known signatures that are generated by well known

information hiding tools. Their techniques can also be extended for further use in some watermarking systems.

Redundancy: To ensure robustness, the watermark information is embedded at multiple places in the cover object. This implies that the watermark can usually be retrieved from just a minor portion of the watermarked file.

Compression ratio: While performing compression the characteristics of the originality must remain the same in case of lossy techniques. Multimedia files are usually compressed using different schemes, such as MPEG-Layer 3 for audio compression, JPEG for image, MPEG-4 for video and so on. Any image with an embedded watermark should yield a similar compression ratio as its unmarked counterpart, for ensuring that its value does not get degraded. Moreover, the compression process should not affect or remove the watermark.

Multiple watermarks: Multiple users or even a single user should embed a watermark into the cover object. This implies that a user should ideally embed a watermark without affecting or disturbing any preexisting watermark that may be already residing in the cover object. This must prove a true even if the watermarking algorithms used are not same.

Secret keys: In general, watermarking systems should use one or more cryptographically secure keys to ensure that the watermark cannot be manipulated or removed. This is necessary because once it is read by anyone, it can be altered or modified by the same person as both the location and embedding algorithm of the mark will be familiar [Kutter et al.][12] in [36]. It is not safe to presume that the embedding algorithm is not known to the attacker. As the security of the watermarking system depends on the utilization of secret keys, the key space must be large. so that a brute force attack becomes impractical. In most of the watermarking systems, the key is the PN-pattern itself, or at least is used as a seed to produce it. Moreover, the watermark message is generally

encrypted first using a cipher key, before it is embedded with the watermark key. This approach adds security at two levels. The highest level of secrecy, does not allow the user to interpret or decode the watermark, or even detect the very presence of it. In the second level of secrecy, user is allowed to detect the presence of the watermark, but the data cannot be decoded without the proper key. Watermarking systems in which the key is familiar to various detectors are known as unrestricted-key watermarks. Thus, algorithms for use as unrestricted-key systems must employ the same key for every piece of data [Miller et al.][79]. Those systems that use a different key for each and every watermark (and thus where the key is shared by only a few detectors) are known as restricted-key watermarks.

Computational cost The amount of time that it takes for a watermark to be embedded and detected works as a crucial factor in a watermarking system. Certain applications, like broadcast monitoring, need real time watermark processing and thus procrastination is not acceptable under any circumstances. On the other hand, for court disputes (which are rarely found), a detection algorithm that takes hours is fully acceptable so long as the effectiveness is high.

Moreover, the number of embedders and detectors differ according to the application. This fact affects the cost of the watermarking system. Applications such as DVD copy control needs few embedders but also requires detector on each DVD player so that cost of retrieval should be very low and that of embedding can be a little higher. Whether the algorithms are implemented as plug-ins or dedicated hardware also affects the economics of deploying a system.

3.3 Introduction to Data Mining and its Related Concepts

When there is large quantum of specific data related to some specific application, there is always a possibility to explore the hidden information in the database. The utilization of hidden information from a given database is referred to as data mining.

The output of the data mining is not just a subset of the database but it is output of some analysis of the contents of the database. Data mining involves many diverse algorithms to accomplish different tasks. All of these algorithms attempt at fitting a model to the data. The algorithm examines the data and selects a model which is nearest to the characteristics of the data being tested.

Classification: Classification converts data into predefined groups or classes. It is often known as supervised learning because the classes are determined before testing the data. The classification algorithms require the classes to be defined based on data attribute values and very often they describe these classes by looking at the key features of data already known to belong to that class.

Decision Trees: A decision tree is a predictive modelling technique exploited in classification, clustering and prediction process. The tree is created to model the classification process. The divide and conquer technique is used in decision trees to split the problem search space into subsets. This tree has one root and the leaves of the tree represent classifications and branches represent conjunctions of features that lead to such classifications.

Once the tree is constructed, it is applied to each tuple in the database and results in a classification for that tuple. The decision tree has the following properties:

- Each internal node is labelled with an attribute A_i .

- Each arc is labelled with a predicate that can be applied to the attribute associated with the parent.
- Each leaf node is labelled with a class, C_j

Solving any problem using decision tree is a two step process.

1. Decision tree induction: construct a DT using training data. In our work we have used frequencies and not the pixels.
2. For each tuple belonging to training data, apply the DT to determine its class.

The major factor in the performance of the DT building algorithms is the size of the training set. The following issues are faced by most DT algorithms.

- Choosing splitting attributes: which attributes to use for splitting attributes impacts the performance applying the built decision tree.
- Ordering of splitting attributes
- Splits: Associated with the ordering of the attributes is the number of splits to take.
- Tree structure: To improve the performance of applying the tree for classification, a balanced tree with the fewest levels is desirable.
- Stopping criteria: The construction of the tree definitely stops when the training data are perfectly classified. There may be situations when stopping earlier would be desirable to prevent the creation of larger trees.
- Training data: The structure of the DT created depends on the training data. If the training data set is too small, then the generated tree might not be specific enough to work properly with the more general data whereas if the training data set is too large, then the created tree may over fit.

- Pruning: The pruning phase is required to remove redundant comparisons or remove subtrees to achieve better performance.

3.4 The ID3

The ID3 [Dunham et al.][86] technique to create a Decision Tree is based on information theory and attempts at minimizing the expected number of comparisons. The basic strategy used by ID3 is to choose splitting attributes with the highest information gain first. The concept used to quantify information is called entropy. Entropy is used to measure the amount of uncertainty or surprise or randomness in a set of data. Certainly, when all the data is a set belonging to a single class, the Decision Tree classification is to iteratively partition the given data set into subsets where all elements in each final subset belong to the same class. We have used ID3 algorithm for preparing the Decision tree, other techniques that could further be applied are C4.5 - an improved ID3, CART (Classification and Regression trees) - which generates a binary decision tree and Neural Network based algorithms.

Weka [87] is an open source data mining tool which provides a collection of machine learning algorithms and the environment for knowledge analysis. The algorithms can either be applied directly to a data set or called from one's own code. Weka contains tools for data pre-processing, classification, regression, clustering, association rules and visualization. It is also well-suited for developing new machine learning schemes. Weka is open source software available under the GNU General Public License. This particular tool is available on Windows, Mac OS x and Linux platforms.

In this work Weka 3.4 has been used as a Data mining tool for generating the Decision tree. This tool along with the decision tree also provides a few additional statistical results. The observations of the following statistical measures, obtained through the weka software are:

- **True positives** - The number of items correctly labelled as belonging to the positive class.
- **True negatives** - equivalence to correct rejections.
- **False Positives** - equivalent to false alarm.
- **False Negatives** - equivalent to miss.
- **TP Rate** - is given by $TP/P = TP/(TP + FN)$ equivalent with hit rate, recall, sensitivity.
- **FP Rate** - is given by $FP/N = FP/(FP + TN)$ equivalent with false alarm rate, fall-out.
- **Precession** - is termed as the number of relevant items retrieved by a search divided by the total number of documents. Precession is given by the ratio of number of True positives to the Total number of elements labelled as belonging to the positive class.
- **Recall** - is defined as the number of relevant documents retrieved by a search divided by the total number of existing relevant documents (which should have been retrieved) Recall is given by the number of true positives to the total number of elements that actually belong to the positive class.
- **F-measure** - is the harmonic mean of prevision and recall is given by $2*(P*R)/(P+R)$
- **Kappa** $= (\text{observed agreement} - \text{chance agreement}) / (1 - \text{chance agreement})$ The value of kappa if it is +1 gives a perfect agreement, a value of zero is agreement and a value of -1 is complete disagreement.

3.5 Summary

This Chapter started with a general view to classify the digital watermarking techniques in spatial domain and transformed domain for digital images. The concepts of data mining and ID3 algorithm have been reviewed, which have been used in the design approach in the subsequent Chapters. The watermarking design issues have been explored in detail, keeping in view the characteristics to be provided during the design of watermarking techniques explored in Chapter 5 and Chapter 6.

Chapter 4

Transformation Based Techniques

This Chapter deals with latest transformation based techniques that are employed for digital watermarking. A DCT and DWT based transformation techniques are explained and its applications in Image watermarking are explored in this Chapter. This Chapter also provides detailed summary of the work done by various authors in this field. Experimental results obtained using DCT and DWT are shown in this Chapter. This initial work was done to further evolve with the new algorithm.

4.1 Transformation Based Techniques for Performing Digital Watermarking DCT/DWT

An advantage of the spatial techniques is that they can be easily applied to any image, regardless of subsequent processing. A possible disadvantage of spatial techniques is that, these techniques do not allow for the exploitation of the image processing operations in order to increase the robustness of the watermark.

In addition to this, adaptive watermarking techniques are considerably more difficult in the spatial domain. Both the robustness and quality of the watermark could

be improved, if the properties of the cover image could be exploited similarly. For instance, it is generally preferable to hide watermarking information in noisy regions and edges of images, rather than in smoother regions. The benefit is two-fold, degradation in smoother regions of an image is more noticeable to the Human Visual System (HVS), and becomes a prime target for lossy compression schemes.

Taking these aspects into consideration, working in a frequency domain of some sort becomes very attractive. The classic and still most popular domain for image processing is that of the Discrete-Cosine-Transform (DCT). The DCT allows an image to be broken up into different frequency bands, making it much easier to embed watermarking information, into the middle frequency bands of an image. The middle frequency bands are chosen, such that they, have similar values and it does not over-expose themselves to removal through compression and noise attacks in high frequencies [Langelaar et al.][40].

A detailed summary of the work done by various authors using DCT as a basic transformation are shown in Table 4.1, 4.2, 4.3 and those using DWT are shown in Table 4.4 and 4.5 respectively.

Name of the Author	Year	Embedding Type	Watermark Type	Transform Domain	Embedding Location
Noore A. [88]	2003	Additive	Binary Image	Modified DCT	Not Specified
Fotopoulos and skodras [89]	2000	Additive	Not Specified	Subband DCT	Not Specified

Table 4.1: List of block based DCT algorithms without any perceptual modeling

Name of the Author	Year	Embedding Type	Watermark Type	Watermark Detection	Embedding Location
Choi and Aizawa [90]	2002	Additive	Gaussian Vector	Correlation	Luminance Domain
Suhail and Obaidat [91]	2003	Additive	Gaussian Vector	Correlation	Not Specified
Golikeri and Nasiopoulos [92]	2004	Additive	Gaussian Vector	Correlation	Luminance Domain

Table 4.2: List of block based DCT algorithms using implicit perceptual modeling

Name	Embedding Type	Perceptual Modeling Strategy	Watermark Type	Watermark Detection	Embedding Location
Tao[64]	Additive	Regional Classifier	Not Specified	Correlation	Luminance Domain
Hsu [65]	Additive	Frequency Classifier	Gaussian Vector	Correlation	Mid Frequency
Huang et al [93]	Additive	Luminance Texture masking	Gaussian Vector	Correlation	DC Component
Wong et al [94]	Additive	Band pass Filtering	Gaussian Vector	Correlation	DC Component

Table 4.3: List of block based DCT algorithms using explicit perceptual modelling

Name	Filter Used	Level	Embedding Type	Watermark Type	Watermark Embedded
DWT Based Non-Blind Watermarking Algorithms					
Ganic and Eskjcioglu [95] 2005	Haar	1	Additive	Grey Scale Image	High and Low Pass bands
Tao and Eskicioglu [96] 2004	Haar	2	Additive	Binary Image	All Bands
Kundur and Hatzinakos [97] 2004	Daubechies 10 pt wavelet	3	Additive	Grey Scale Image	High pass bands
Raval and Rege [98] 2003	Not Specified	2	Additive	Binary Image	2nd level LL and HH bands
Kang et al. [99] 2003	Daubechies 9/7 biorthogonal	3	Additive BCH (61,8), 2D interleaving	Gaussian Vector	Low Pass LL
Hsieh and Wu [100] 2001	Not Specified	3	Additive	Gaussian Vector	High and Low pass bands
Niu et al. [101] 2000	Not Specified	2	Additive	Grey Scale Image	Not Specified
Hsu and Wu [102] 1998	Daubechies tap-6	Multi	Neighboring Relationship	Binary Logo	Not Specified
Chae and Manjunath [103] 1998	Haar	1	Additive	Grey Scale Image	Not Specified
Xia et al. [104] 1998	Haar	Multi	Additive	Gaussian Vector	High pass band
Xia et al. [71] 1997	Haar	2	Additive	Gaussian Vector	High pass band
Kundur and Hatzinakos [74] 1997	Not Specified	4	Additive	Binary Image	Low Pass Bands

Table 4.4: List of DWT based non-blind watermarking algorithms

Name	Filter Used	Level	Embedding Type	Watermark Type	Watermark Embedded
DWT Based Blind Watermarking Algorithms					
Xiao et al. [105] 2002	Not Specified	Not Specified	Additive	Gaussian Vector	Mid Freq. Component
Kaewkamnerd [106] 2000	Quadrature Mirror	Multi	Additive	Gaussian Vector	High pass bands
Kaewkamnerd [107] 2000	Quadrature Mirror	4	Additive	Gaussian Vector	High pass bands
Lu et al. [108] 1999	Not Specified	Not Specified	Additive	Binary and Grey scale image	Visually significant coef.
Zhu et al. [109] 1999	Not Specified	Multiple	Additive	Gaussian Vector	High pass bands
Kundur and Hatzinakos [110] 1998	Daubechies 10 pt wavelet	Not specified	Additive	Binary watermark (-1,1)	High pass band

Table 4.5: List of DWT based blind watermarking algorithms

4.2 Embedding Using DCT & DWT Transformation Based Techniques

In this preliminary work, the technique utilizes the comparison of middle-band DCT coefficients to encode a single bit into a DCT block. To begin, we define the middle-band frequencies (F_M) of an 8 x 8 DCT block as shown below in Figure 4.1. F_L denotes the lowest frequency components of the block, while F_H denotes the higher frequency components. F_M is chosen as the embedding region as to provide additional resistance to lossy compression techniques, while avoiding significant modification of the cover image [Hernandez et al.][38].

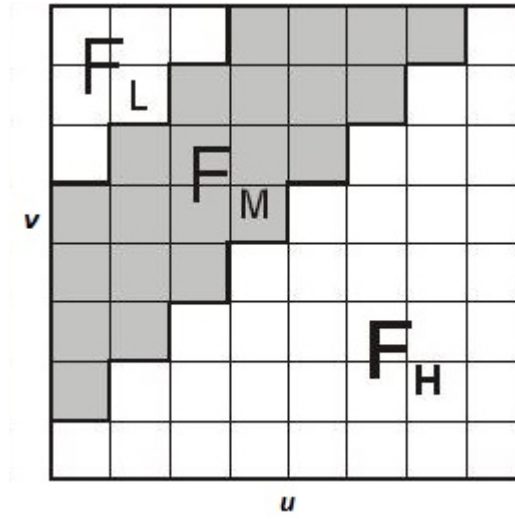


Figure 4.1: Definition of DCT regions

Next, two locations $B_i(u_1, v_1)$ and $B_i(u_2, v_2)$ are chosen from the F_M region for comparison. Rather than arbitrarily choosing these locations, extra robustness to compression can be achieved if we base the choice of coefficients on the recommended JPEG quantization table shown in Table 4.6. If two locations are chosen such that they have identical quantization values, we can feel confident that any scaling of one coefficient will scale the other by the same factor preserving their relative size. Based

16	11	10	16	24	40	51	61
12	12	14	19	26	58	60	65
14	13	16	24	40	57	69	56
14	17	22	29	51	87	80	62
18	22	37	56	68	109	103	77
24	35	55	64	81	104	113	92
49	64	78	87	103	121	120	101
72	92	95	98	112	100	103	99

Table 4.6: Quantization values used in JPEG compression scheme

on the table, we can observe that coefficients (4,1) and (3,2) or (1,2) and (3,0) would make suitable candidates for comparison, as their quantization values are equal. The DCT block will encode a “1” if $B_i(u_1, v_1) > B_i(u_2, v_2)$; otherwise it will encode a “0”. The coefficients are then swapped if the relative size of each coefficient does not agree with the bit that is to be encoded [Johnson et al.][39].

The swapping of such coefficients should not alter the watermarked image significantly, as it is generally believed that DCT coefficients of middle frequencies have similar magnitudes. The robustness of the watermark can be improved by introducing a watermark “strength” constant k , such that $B_i(u_1, v_1) - B_i(u_2, v_2) > k$. Coefficients that do not meet this criteria are modified through the use of random noise as to then satisfy the relation. Increasing k thus reduces the chance of detection errors at the expense of additional image degradation.

$$I_{W_{x,y}}(u, v) = \begin{cases} I_{x,y}(u, v) + k * W_{x,y}(u, v), & u, v \in F_M \\ I_{x,y}(u, v), & u, v \notin F_M \end{cases} \quad (4.1)$$

Wavelet Watermarking Techniques Another possible domain for watermark embedding is that of the wavelet domain. The DWT (Discrete Wavelet Transform) separates an image into a lower resolution approximation image (LL) as well as horizontal (HL), vertical (LH) and diagonal (HH) detail components. The process can then be repeated to compute multiple “scale” wavelet decomposition, as in the 2 scale wavelet transform shown in Figure 4.2.

One of the many advantages of the wavelet transform is that, it is believed to more accurately model the aspects of the HVS, as compared to the FFT or DCT. This allows us to use higher energy watermarks in regions, that the HVS is known to be less sensitive to, such as the high resolution detail bands (LH, HL, HH). Embedding watermarks in these regions allow us to increase the robustness of our watermark, at little or no additional impact on image quality [Langelaar et al.][40].

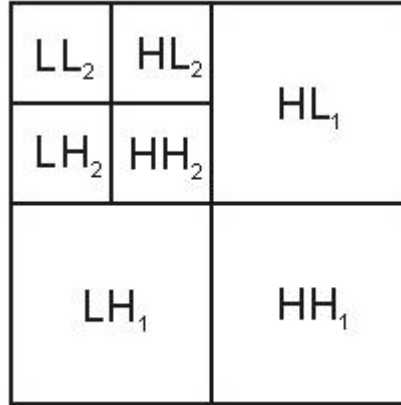


Figure 4.2: 2 Scale 2-Dimensional discrete wavelet transform

A few implementation results obtained during the preliminary understanding phase of watermarking is as shown below.

4.2.1 Embedding Algorithm for MBCX (Mid Band Coefficient Exchange Method)

Mid Band Coefficient Exchange Method

- Set minimum coefficient difference.
- Set the size of the block in cover to be used for each bit in watermark.
- Read in the cover object.
- Determine the size of cover object.
- Determine the maximum message size based on cover object, and block size.
- Read in the message image.
- Reshape the message to a vector.
- Check that the message is not too large for cover, if length of message is greater than maximum message size error message is displayed.

- Pad the message out to the maximum message size with ones.
- Generate shell of watermark image.
- Process the image in blocks. Encode such that $(5,2) > (4,3)$ when $\text{message}(x) = 0$; and that $(5,2) < (4,3)$ when $\text{message}(x) = 1$.
- In loop
 - Transform block using DCT.
 - If message bit is 0 then $(5,2) > (4,3)$
If message bit is 1 then $(5,3) < (4,3)$
If the above cases are not satisfied then we need to exchange them.
 - Now we adjust the two values such that their difference \geq coefficient difference set in step 1.
 - Transform block back into spatial domain using *IDCT2*.
 - Move on to next block, at end of row move to next row.
- Convert to 8 bit integer using *uint8* and write the watermark image out to a file.
- Display processing time.
- Calculate PSNR and display it.
- Display the watermarked image.

4.2.2 Recovery Algorithm for MBCX (Mid Band Coefficient Exchange Method)

- Clear all.
- Save start time.
- Set the size of the block in cover to be used for each bit in watermark.
- Read in the watermarked object.
- Determine the size of watermarked image.
- Determine maximum message size based on cover object, and block size.
- Read in the original watermark.
- Determine the size of the original watermark.
- In loop process the image in blocks.
 - Transform block using DCT
 - If $dctblock(5,2) > dctblock(4,3)$ then $message(x) = 0$ else $message(x)=1$.
 - Move to the next block and at the end of the row move to the next row.
- Reshape the embedded message.
- Display processing time.
- Display the Recovered message.

Here for the implementation of low band, nothing but the coefficient of the standard JPEG table is chosen such that the coefficients have similar values.

4.2.3 DWT Based Embedding Technique

- Save CPU start time.
- Set the gain factor for embedding.
- Read in the cover object.
- Determine the size of cover image.
- Read in the message image and reshape it into a vector.
- Set the value of key for PN generator (may be Image or Number.)
- Reset MATLAB's PN generator to "key", as set in above step.
- Perform Transformation using *DWT2* on the cover image using haar as the mother wavelet.
- In a loop Add PN sequences to H1 and V1 components when message =0.

If message(x) = 0 then

$$CH1 = CH1 + K * pn_sequence.$$

$$CV1 = CV1 + K * pn_sequence.$$

- Perform *IDWT2* to get watermarked image.
- Convert back to *uint8*.
- Write watermarked image to file.
- Display the elapsed time.
- Calculate the PSNR.
- Display the watermarked image.

4.2.4 DWT Based Recovery Technique

- Save CPU start time.
- Read in the watermarked object
- Determine the size of watermarked image.
- Read in the original watermark.
- Determine the size of original watermark.
- Read in key for PN Generator.
- Reset MATLAB's PN generator to "key", set in above step.
- Initialize message to all ones.
- Perform *DWT2* on watermarked image using haar as mother wavelet.

Generate *pn_sequence* for horizontal and vertical components.

Calculate correlation between *CH1* and *pn_sequence*.

Calculate correlation between *CV1* and *pn_sequence*.

Calculate average of *CH1* and *CV1* correlation and store it in array.

- In a loop if average correlation $>$ mean of correlation then message (x) = 0.
- Reshape the message vector and display the recovered watermark.
- Display elapsed time.

For second scale operation to be performed, *DWT2* is performed twice on the cover image and the required embedding takes place in H2 and V2 components. On the receiver side, the extraction of the information is done from H2 and V2 components, rest of the above algorithmic steps remain the same.

4.2.5 Implementation Results of DCT Based Method

In MBCX (Mid Band Coefficient Exchange) method, K is the difference in coefficient and B is the block size. Table 4.7 shows the value of PSNR obtain with different block sizes of 8 and 16 in DCT with different watermarking gain constant K.

Mid Frequency Band Results				
		LENA	MOON	CAMERA MAN
Block Size	K	PSNR	PSNR	PSNR
$B = 8$	10	2.12E+04	4.42E+04	1.80E+04
	50	2.41E+03	2.92E+03	2.45E+03
	200	183.4009	201.5846	190.9553
$B = 16$	10	1.17E+04	3.88E+04	8.43E+03
	50	5.32E+03	8.41E+03	4.81E+03
	200	624.7721	744.1055	6.37E+02

Table 4.7: DCT results for mid band

4.2.6 Implementation Results of DWT - Based Method of 1 Scale and 2 Dimensions

Table 4.8 shows PSNR value obtain using DWT for 1-scale in 2-dimension with different images, where K is the watermarking gain constant and embedding is done in HL and LH band.

DWT Result using PN sequence(HAAR)			
Gain Constant	LENA	MOON	CAMERA MAN
K	PSNR	PSNR	PSNR
0.2	4.5852E+04	5.0465E+04	4.8105E+04
0.5	9.0000E+03	1.0831E+04	9.4811E+03
1	2.3216E+03	3.0520E+03	2.4488E+03
2	5.8312E+02	8.2148E+02	6.3395E+02
10	2.5822E+01	3.8053E+01	2.9130E+01
50	4.8272E+00	4.3320E+00	4.8499E+00

Table 4.8: DWT results for 1-Scale and 2-Dimensions

4.2.7 Implementation Results of DWT - Based Method of 2 Scale and 2 Dimensions

Table 4.9 shows PSNR value obtain using DWT for 2-scale in 2-dimension with different images and watermarking gain constants.

All the results were obtained in Matlab and since DCT and DWT techniques takes

DWT Result using PN sequence (HAAR)			
Gain Constant	LENA	MOON	CAMERA MAN
K	PSNR	PSNR	PSNR
0.2	1.0584E+05	1.1390E+05	1.1103E+05
0.5	3.2000E+04	3.5818E+04	3.3576E+04
2	3.0005E+03	3.0375E+03	2.44E+03
5	3.7190E+02	5.4171E+02	4.1143E+02
10	9.4006E+01	1.4112E+02	1.0727E+02

Table 4.9: DWT results for 2-Scale and 2-Dimensions

more time, the best way is to use $nLSB$ approach to embed more information in the cover image. Figure 4.4, Figure 4.5 and Figure 4.6 show the recovered watermarked images with different values of the watermarking gain constant K for LENA, MOON and CAMERAMEN images respectively.

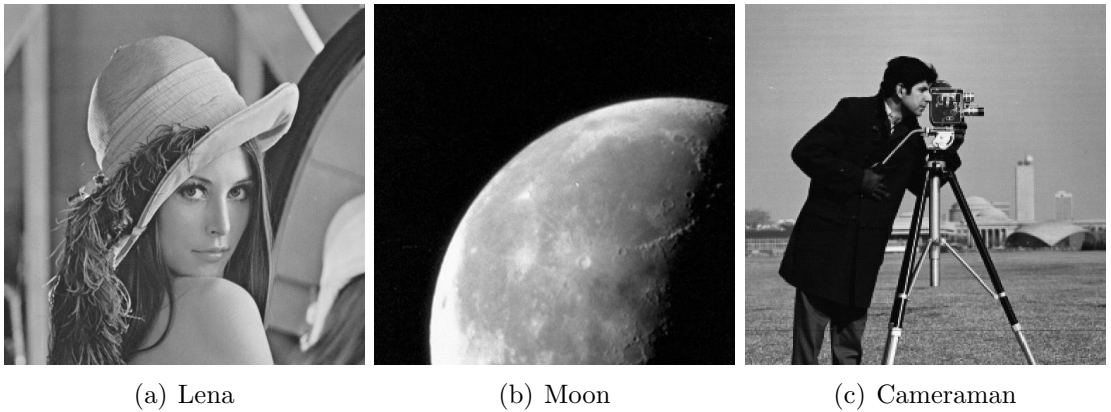


Figure 4.3: Watermarked images using DWT with $k = 0.5$

RECOVERED WATERMARKED IMAGES USING DWT

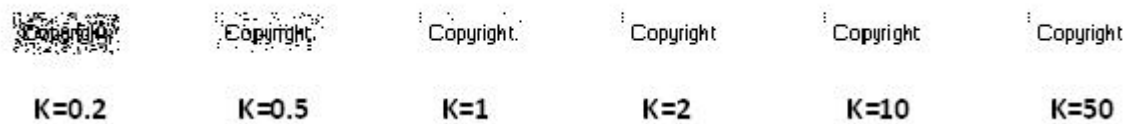


Figure 4.4: Recovered watermark images from LENA image



Figure 4.5: Recovered watermark images from MOON image

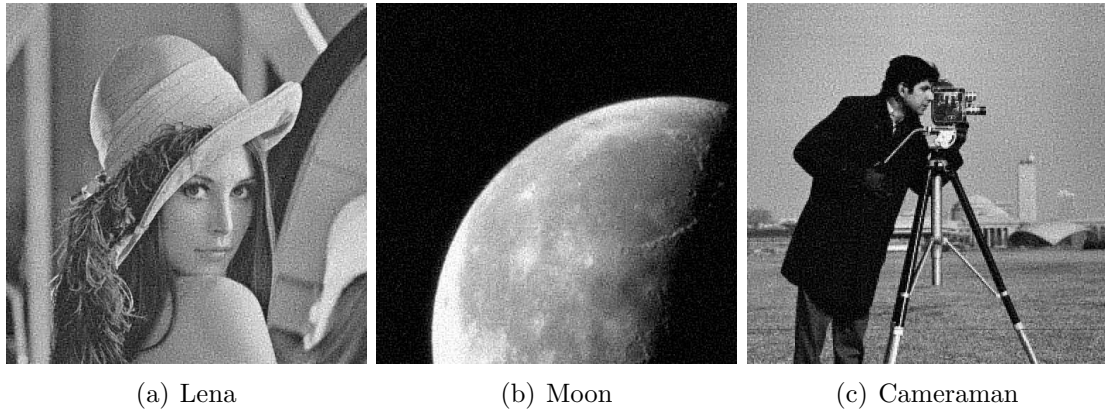


Figure 4.6: Recovered watermark images from CAMERAMAN image

Figure 4.7 shows the watermarked images of LENA, MOON and CAMERAMAN using DWT as the transform and with K as 5. Recovered watermarked images for LENA, MOON & CAMERAMAN using DWT for 2 Scale 2 Dimension with $K = 5$ is as shown in Figure 4.8

4.3 Summary

In this Chapter we surveyed the current literature on digital image watermarking and classified watermarking algorithms based on the transform domain in which the

Figure 4.7: Watermarked images using DWT with $k = 5$ Figure 4.8: Recovered watermarked images for LENA, MOON & CAMERAMAN using DWT for 2 Scale 2 Dimension with $k = 5$

watermark is embedded. The analysis of techniques for information hiding in images covers both steganography and digital watermarking. The analysis involves investigating available tools and techniques for hiding information, classifying these techniques and understanding the impact of it. It is observed that transform domain techniques for watermarking, are better than spatial domain techniques for both reasons of robustness as well as visual impact. Based on the discussion made in the preceding Sections, it is observed that, no single common solution to all problems presently exist. The solutions are more likely to remain application specific and the trade-offs between the conflicting requirements of low distortion and robustness to acceptable manipulations, still have to be made. Next Chapter 5 provides a novel approach to watermarking technique in images.

Chapter 5

Watermarking Using Decision Tree and DCT

This Chapter discusses a new approach to perform Digital Watermarking using the techniques of Data Mining. The design objective of the work is to have a new robust digital watermarking technique by combining the power of transform domain technique like the Discrete Cosine Transform (DCT) [Zeng et al.][111] [Huang et al.][112] [Eyadat][113] [Sayood][114] and the data mining technique such as Decision Tree Induction (ID3)[Dunham et al.][86]. The approach focuses on a technique through which the notion of decision tree (DT) can be applied on transformed vectors to build the decision tree. Based on the training provided, using images for the identification of image blocks, the classification tree or decision tree is obtained. The resulting decision tree provides, decision making rules to classify good quality image blocks for insertion of the watermark. A mathematical model is provided in this Chapter. Complete model with flowchart and the necessary explanation is given in this Chapter along with the comparison of work with other authors.

5.1 Mathematical Model

A mathematical model for performing the digital watermarking in color images using a data mining approach is shown in this part of discussion. The overall block diagram for the major activities performed by the approach followed is shown in Figure 5.1.

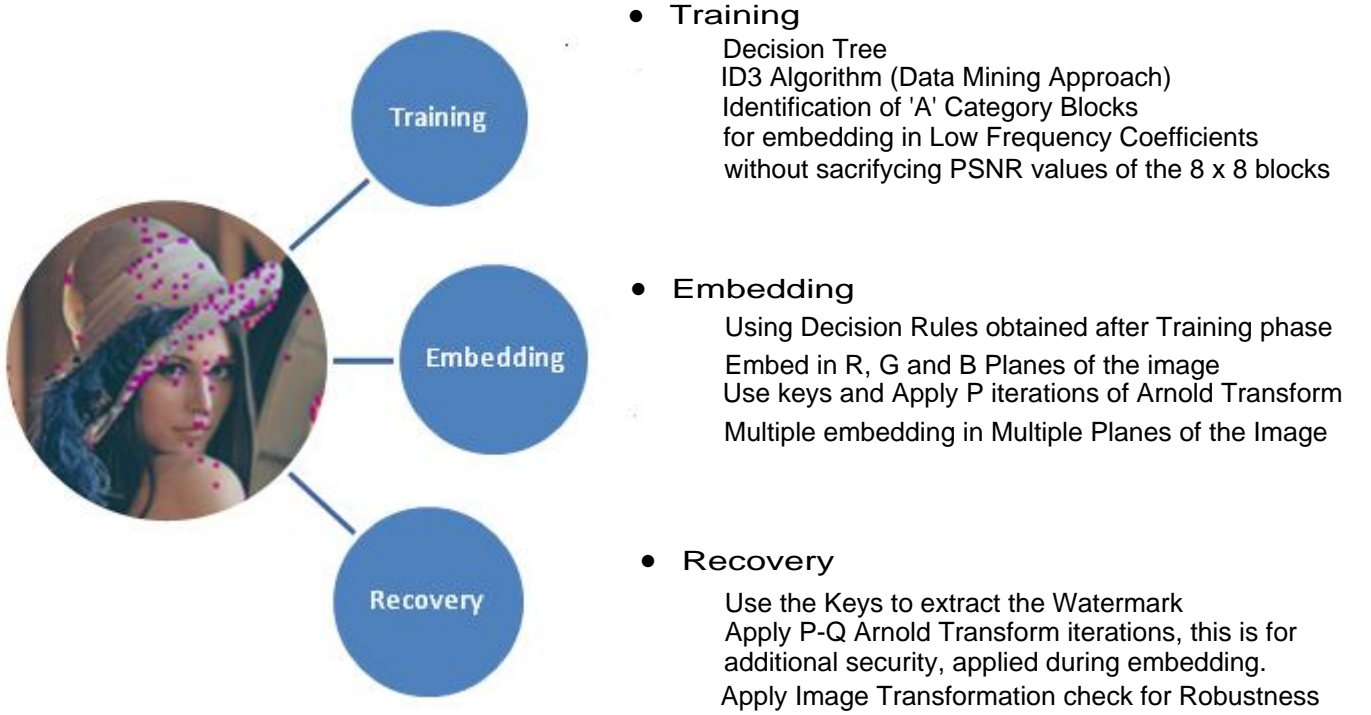


Figure 5.1: A major activities performed by the functions to perform digital watermarking

5.1.1 Training Before Actual Identification of Blocks

Let the watermark image be W_i , for each image block of size 8 x 8, obtain the DCT of it, resulting into

$$W_i^* = DCT(W_i) \quad (5.1)$$

Each vector within W_i^* will have N coefficients, where N is 64, we are interested in first 4 coefficients of these vectors,

$$\begin{aligned}
AC1_i &= \{AC1_1, AC1_2, AC1_3, AC1_4, \dots, AC1_N\} \\
AC2_i &= \{AC2_1, AC2_2, AC2_3, AC2_4, \dots, AC2_N\} \\
AC3_i &= \{AC3_1, AC3_2, AC3_3, AC3_4, \dots, AC3_N\} \\
AC4_i &= \{AC4_1, AC4_2, AC4_3, AC4_4, \dots, AC4_N\}
\end{aligned} \tag{5.2}$$

Based on these 4 AC coefficients, we calculate α_1 , α_2 , α_3 and α_4 as follows,

$$\begin{aligned}
\alpha_1 &= \sum_{i=1}^N AC1(W_i^*) \\
\alpha_2 &= \sum_{i=1}^N AC2(W_i^*) \\
\alpha_3 &= \sum_{i=1}^N AC3(W_i^*) \\
\alpha_4 &= \sum_{i=1}^N AC4(W_i^*)
\end{aligned} \tag{5.3}$$

Let β be the average of all AC coefficients and is calculated as follows.

$$\beta = \frac{1}{4} \sum_{i=1}^4 \alpha_i \tag{5.4}$$

During the training phase we use Training images, let these training images be represented as TI_i

Watermarking is performed using the equation as

$$TI_{watermarked} = TI_{original} + \beta TI_{original} \tag{5.5}$$

DC value of the host image i.e. $TI_{original}$ is not modified. This is just to minimize the distortion of the watermarked image. Therefore we keep the DC value un-watermarked.

Let the Host image used during the training be TI , for each image block of 8×8 obtain the DCT of it, resulting into $TI_i^* = DCT(TI_i)$ for all image blocks of size 8×8 .

The equation could then be represented as

$$DCT(TI_i^T) = \begin{cases} DCT(TI_i) + \beta DCT(TI_i) & \forall AC1, AC2, AC3, AC4 \text{ Coeff.} \\ DCT(TI_i) & \forall DC \text{ value} \\ DCT(TI_i) & \forall \text{ Otherwise} \end{cases} \quad (5.6)$$

Where TI_i^T is the watermarked signal of TI_i after modifications to the low frequency coefficients with the value of β .

Inverse DCT transformation is then performed to get back to the image form.

$$TI_i = IDCT(TI_i^*) \quad (5.7)$$

The calculation of PSNR (Peak Signal to Noise Ratio) value for each of this block is performed.

A file is created to store all these coefficient vector consisting of DC , $AC1$, $AC2$, $AC3$, $AC4$, PSNR and the TARGET. As there are 4096 blocks available because of the image size of 512×512 . Taking 5 images of 512×512 size, for the required training, approximately 20480 records are available for training classifier.

Generalization of Coefficients into classes as ‘A’, ‘B’, ‘C’, ‘D’ and ‘N’ Classes

The JPEG quantization tables are designed, in such a way that it grows as we move from the upper left corner to the bottom right. The DC coefficient [at position (0,0), the top left corner] is a measure of the average value of the 64 original pixels, constituting the data unit. A few low frequency coefficients could have nonzero values and these, along with DC coefficient, would be enough to reconstruct the original data units to a high precision.

If $DC > 1200$, DC is categorized as 'A', if between $500 \leq DC \leq 1200$ as 'B' else to 'C'

If $AC1 \leq 0$ as 'N', $AC1 > 130$ as 'A', if between $100 \leq AC1 \leq 200$ as 'B' else as 'C'.

If $AC2 \leq 0$ as 'N', $AC2 > 200$ as 'A', if between $100 \leq AC2 \leq 200$ as 'B', if between $50 \leq AC2 < 100$ as 'C' else as 'D'.

If $AC3 \leq 0$ as 'N', $AC3 > 130$ as 'A', if between $100 \leq AC3 \leq 130$ as 'B', if between $50 \leq AC3 < 100$ as 'C' else as 'D'.

If $AC4 \leq 0$ as 'N', $AC4 > 130$ as 'A', if between $100 \leq AC4 \leq 130$ as 'B', if between $50 \leq AC4 < 100$ as 'C' else as 'D'.

The categorization of the blocks into A , B , C and R (Rejected) based on the PSNR value is then performed.

If $PSNR \leq 10$ then this classes are identified as 'R', where 'R' stands for rejection.

If $10 \leq PSNR \leq 20$ then this class is identified as 'C', if $20 < PSNR \leq 30$ it is identified to be 'B' class and if $PSNR > 30$ it is categorized as 'A'.

The outcome till now is that, based on the values of PSNR, the categorization of all the blocks 'A', 'B', 'C', 'D' and 'N' are mapped into 'A', 'B', 'C' or 'R'.

We are interested in these 'A'(Yes) category blocks only, rest categories are to be rejected and hence are mapped to 'N'(No).

This vector bank is now sufficient to provide input to the ID3 algorithm. Based on the target attribute set as 'A', the decision tree is generated [24] [Dunham et al.][86]. Based on this decision tree, decision rules are formed for actually embedding the watermark. Till now, whatever the images have been used, are only for the training purpose.

Blocks thus identified are the ones in which even after modifying low frequency coefficients the PSNR value remains high, which gives us a very good intuition to use only these ‘A’ category blocks for actual embedding. These will further increase the robustness of the embedded watermark because even though one goes for compression of the image the low frequency values remains undisturbed. We have tested the embedding of watermark with various set of images and the results are shown in Chapter 6.

5.1.2 Actual Embedding of The Watermark Within the Host Image

Let the host image be $H_{original}$

The host image has three image planes *R-Red*, *G-Green* and *B-Blue*. So we call this host image as $H_{original,i}$ where i is 1,2 and 3 for *R*, *G* and *B* planes respectively. Let the watermark image be W_i . Embedding is performed as

$$I_{water,i} = (H_{original,i}, W_i) \quad (5.8)$$

The embedding is similar to the one we did during the training phase. We add the constant β which is the average of watermark image into the low frequency coefficients $AC1$, $AC2$, $AC3$ and $AC4$. The *DC* value remains un-watermarked as before.

From the decision tree obtained as the outcome of the training phase, decision rules are prepared to identify ‘A’ category blocks which are at the leaf nodes.

An Arnold transform of the watermark image is performed during the embedding phase to make the system secure, as Arnold transform is periodic in nature, we used the concept of breaking it into two part. A few iterations are executed during the

embedding and the remaining iterations are performed during the decoding phase.

Let the total iterations be P , out of which few iterations say Q are performed during encoding phase, then the P minus Q iterations needs to be executed on the decoding phase. These iterations counts of P and Q are known only to the developer and not to others, hence this provides an additional level of security to the system design. So, even though if the location, where data is hidden is known to some intruder, may be by compromise, the number of iterations to roll over for the Arnold transform is not known to the intruder.

The Arnold transform is performed as follows:

$$\begin{aligned} X' &= (2X + Y) \text{MOD} 32 + 1 \\ Y' &= (X + Y) \text{MOD} 32 + 1 \end{aligned} \tag{5.9}$$

The size of the watermark we have used to embed is 32 x 32.

The overall transformation is like $Z(X', Y') = i(X, Y)$ this is similar to scrambling or ciphering performed in cryptography.

The individual image planes (R , G and B) allows performing embedding of the watermark. One very interesting observation is that this mechanism permits us to perform embedding of multiple watermarks in each of these planes. Since the embedding is performed only in the low frequency coefficients, these embedding are going to be robust against compression. So this approach provides the ability to perform multiple watermarks in each image planes.

The keys for individual watermark performed are stored in a file. The keys are also encrypted to provide one more level of security. These keys will be required by

the decoder to get back the actual watermark data.

5.1.3 Recovery of the Watermark from the Watermarked Image

The decoding is performed as

$$W' = D_i(R_i, I_{original}) \quad (5.10)$$

Where, R_i is the watermarked image and $i = 1, 2 \text{ and } 3$, for the Red, Green and Blue planes of the image respectively.

From these image planes using the keys, get the 4 coefficient values $AC1$, $AC2$, $AC3$ and $AC4$ and generate the vector by subtracting for this four coefficients the marked image and original image. The vector is transformed back into 32×32 and mapping is performed to 0 or β value. The remaining iterations $P \text{ minus } Q$ are applied to undergo the Arnold transform on this image. The resulting image is the recovered watermark image W' . For multiple watermarks the same value of β is repeated with the different set of keys.

However, as the key is to be safely communicated because the design is of non blind type, what is done here is, A special Region of Interest (ROI) based software is designed for inserting keys into the same watermarked image, resulting into a combination of steganography and watermarking for storing information within the watermarked medium (image). This model is specially designed to take care of identification of cropping from different sides, if at all it is done. Here a specialized vector is prepared for different shapes and is stored at four different places for necessary processing. A two level information hiding support is also provided within this module. This multi-level concept could further be used in forensic science for other applications. A successful steganalysis [11] check is also done for the identification of hidden

signatures within the watermarked and stego images. Along with the previous work additionally a software for hiding files of large size containing any type of document including word, excel, image, video, etc. in the cover medium (image, audio, video) is also designed. Here the information is encrypted using a key and this key will be required on the receiver side for necessary extraction. The design also includes the required interface for such application.

A detailed literature on the multimedia files and header formats can be obtained at [115], [Bjontegraard et al.][116].

5.2 Determining the Decision Tree

The model provides the hybrid approach of decision tree induction for watermark block identification. The decision tree induction is used in the process of embedding watermark. Principally the watermark size is less than the original cover image. Using decision tree, we identify significant image blocks.

Discrete Cosine Transform (DCT) is applied on each block. The DCT is closely related to the Discrete Fourier Transform. One can often reconstruct a sequence very accurately from only a few DCT coefficients, a very useful property for applications requiring data compression.

The next step is to pre-process the selected coefficient-data resulting into generalized data. Data generalization is required to avoid over-fitting classification rules, increasing classifier accuracy and to extract meaningful decision rules.

The image is divided into 8×8 non-overlapping blocks. The cell value of each block is scanned in zigzag fashion and stored in vector V . In Vector V there are 63 different AC coefficients and one DC coefficient present in each block.

If the size of host image is 512×512 then total numbers of 4096 non overlapping blocks are available. If 5 images are taken for training set, then $4096 \times 5 = 20,480$ vectors with its block co-efficient values are available with us. This tell us that voluminous information is available to carry out some data mining work and to extract some meaningful information out of it.

5.2.1 Data Pre-Processing and Training for Identification of Blocks

A partial decision tree is shown below in the Figure 5.2. where the root is specified as $DC=A$ category block and the leaves are classified as ‘A’ category blocks based on the training data set. The other nodes in the Decision tree are referring to the DCT coefficients. Based on the coefficients values as per the observation of the Table 5.1, the categorization of DC , $AC1$, $AC2$, $AC3$ and $AC4$ coefficients is done into A , B , C and N . Further, these blocks are classified into A , B , C and R based on the value of the PSNR. A file with the records of such five images with the categorized attributes is provided as input to the ID3 algorithm to form the decision tree. This decision tree is further used for generating the rules which are then used to identify the blocks in the image in which embedding can be performed.

After performing DCT of the images blocks of size 8×8 , we come up with 64 frequency coefficient values, which is not suitable for mining meaningful decision making rules. This gives rise to pre-process the available records. To assign the class labels initially, PSNR values are categorized which in turn forms target attribute for the classification of data set. As ID3 algorithm works only on the categorical attribute, we have generalized our range looking into all the 20,480 elements of DC , $AC1$, $AC2$, $AC3$ and $AC4$ as shown in Table 5.1.

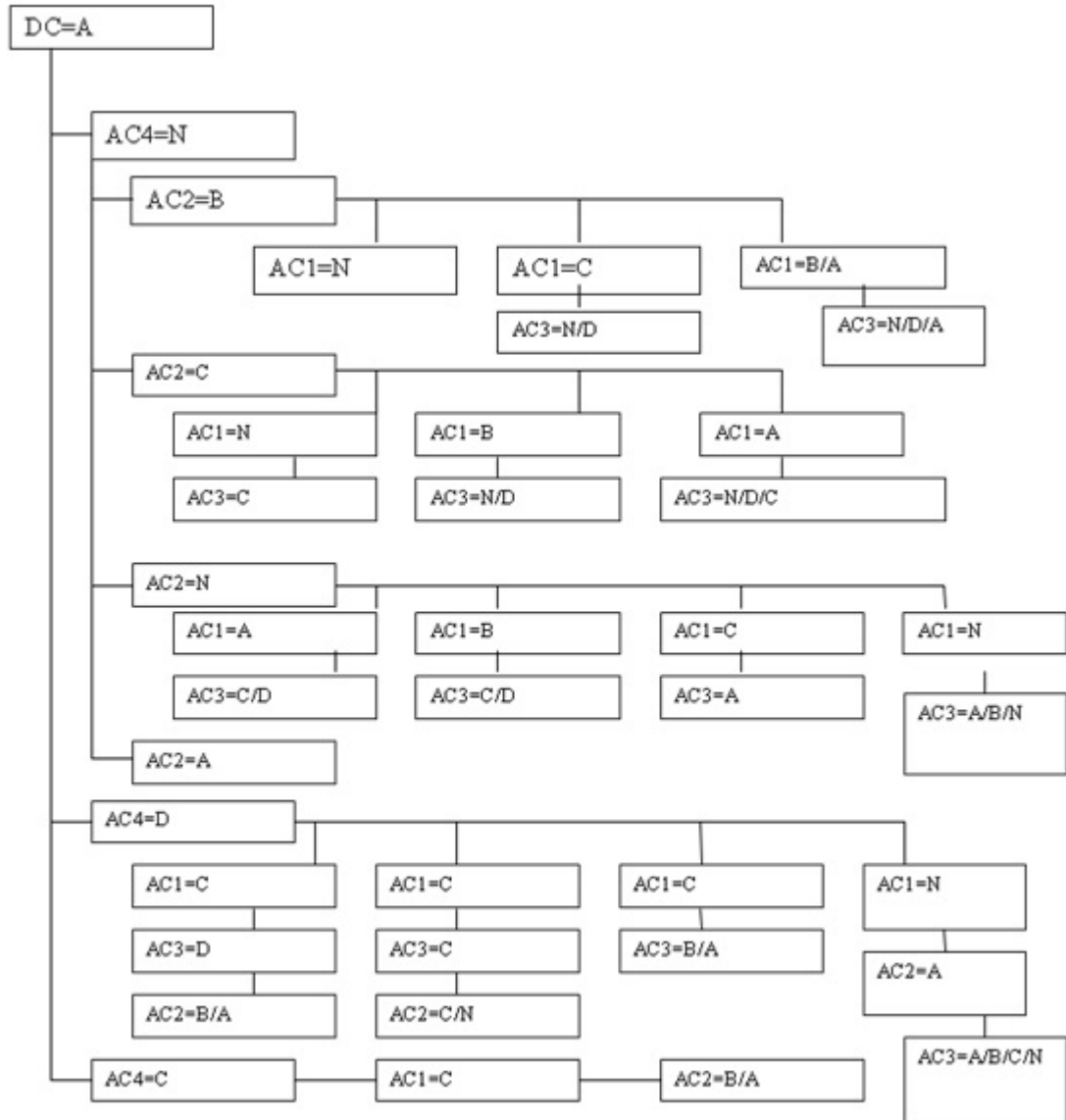


Figure 5.2: A partial decision tree

The coefficients are generalized based on the distribution of the values in different bands. Other generalization based on the actual requirement could also be designed having different band and conditions for classifications. We have considered five images for training, which are Lena, Flower, Baboon, Street and Barbara of size 512×512 and a watermark image of size 32×32 .

Coefficient	Range for Generalized Attribute				
	N	A	B	C	D
DC	-	$\geq 500 \ \& \ \leq 1200$	-	-	-
AC1	≤ 0	>130	$\leq 130 \ \& \ \geq 100$	<100	-
AC2	≤ 0	≥ 200	$\geq 100 \ \& \ <200$	<100	-
AC3	≤ 0	>130	$\geq 100 \ \& \ \leq 130$	$\geq 50 \ \& \ <100$	<50
AC4	≤ 0	>130	$\geq 100 \ \& \ \leq 100$	$\geq 50 \ \& \ <100$	<50

Table 5.1: Generalized range of attributes for selecting the class

A sample file is shown in Table 5.2 for storing the low frequency coefficients like *DC*, *AC1*, *AC2*, *AC3* and *AC4* along with class and the PSNR values of 20,480 records.

DC	AC1	AC2	AC3	AC4	Class	PSNR
1283.5	4.768263	7.947332	-5.03491	-0.7878727	B	29.22464
1251.75	4.851223	-2.694331	3.593048	-1.894951	B	28.8816
1246.25	2.797178	0.7849733	-1.059179	1.565153	B	28.93145
1246.25	-2.858218	-9.752873	-0.2870126	-2.952501	B	28.93145
1269.5	4.095252	-8.276431	-0.2705981	-0.4069359	B	28.981
1270.875	219.1655	-108.2987	-2.113549	-11.19557	A	30.18943

Table 5.2: Sample records with low frequency coefficient, PSNR and target class

By looking at the Figure 5.3, we have selected the cut off for *DC* ‘A’ category blocks as values greater than 1200. In Figure 5.3 good amount of coefficients are available having value greater than 1200. For *AC1* coefficients the plot is shown in Figure 5.4.

Based on the plot of values of *AC1* coefficients as shown in Figure 5.4, the values which are less than zero are straight away categorized as ‘N’, however for value of

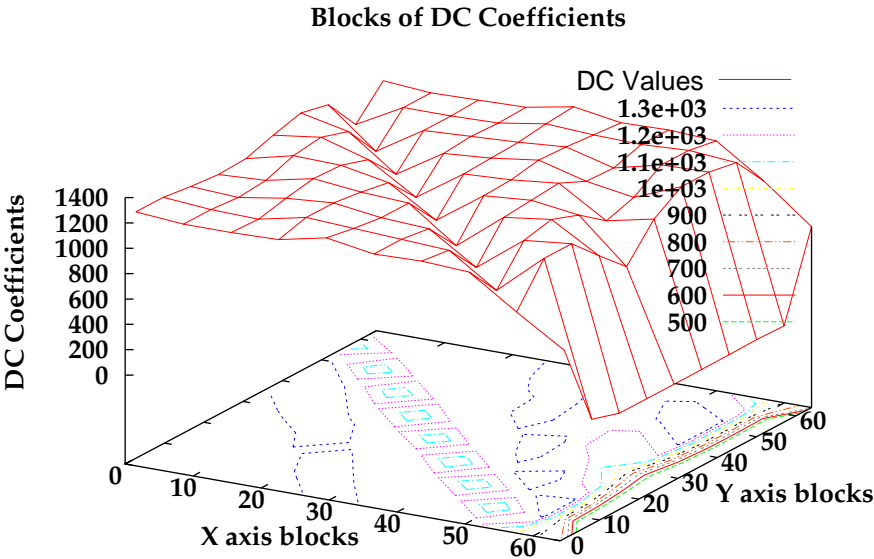


Figure 5.3: Plot of *DC* coefficients

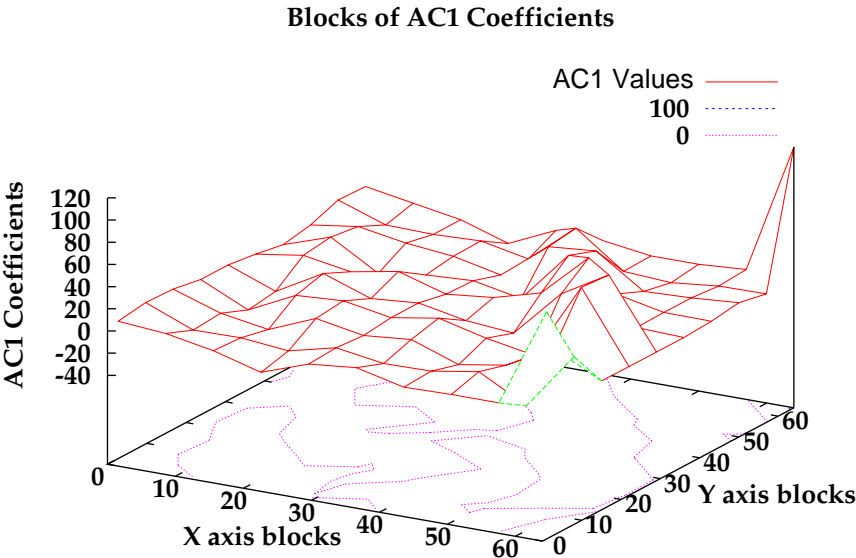


Figure 5.4: Plot of *AC1* coefficients

$AC1 > 130$, we categorized it to ‘A’ and $100 \leq AC1 \leq 130$, we categorized to ‘B’ and rest to ‘C’.

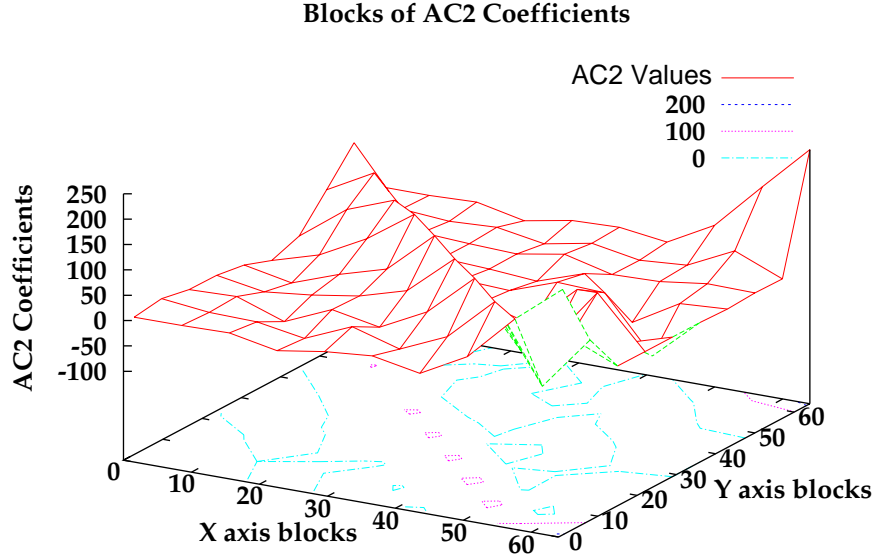


Figure 5.5: Plot of $AC2$ coefficients

Looking at Figure 5.5 the $AC2$ coefficient values ≥ 200 are categorized as ‘A’, $100 \leq AC2 \leq 200$ as ‘B’ and $AC2 < 100$ as ‘C’, values less than 0 as ‘N’.

Looking at Figure 5.6 the $AC3$ coefficient values ≥ 130 are categorized as ‘A’, $100 \leq AC3 \leq 130$ as ‘B’, $50 \leq AC3 < 100$ as ‘C’, < 50 as D and with values less than 0 as ‘N’.

Looking at Figure 5.7 the $AC4$ coefficient values ≥ 130 are categorized as ‘A’, $100 \leq AC4 \leq 130$ as ‘B’, $50 \leq AC4 < 100$ as ‘C’, < 50 as D and with values less than 0 as ‘N’.

Figure 5.3 to Figure 5.7 are of a single image, similar such plots could be obtained for other images used in the training. Initially all records of 5 training images are

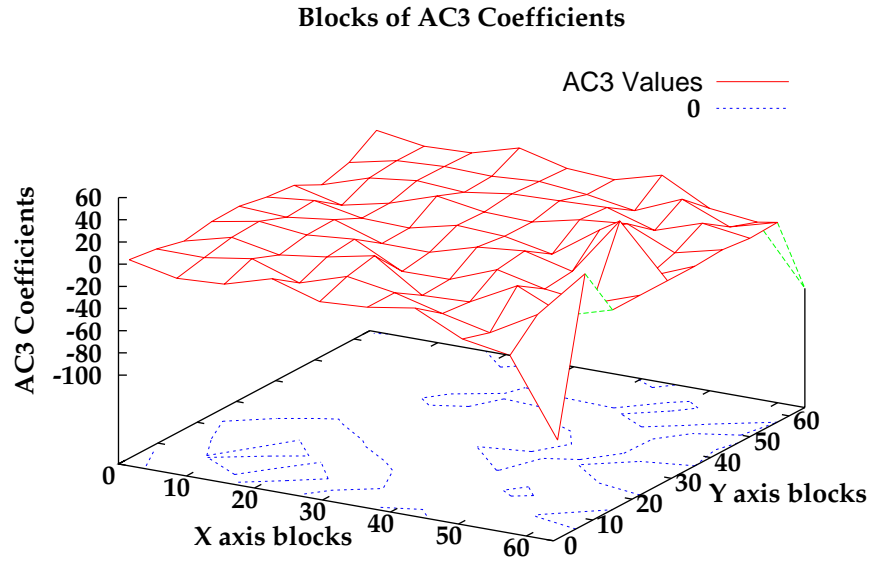


Figure 5.6: Plot of AC_3 coefficients

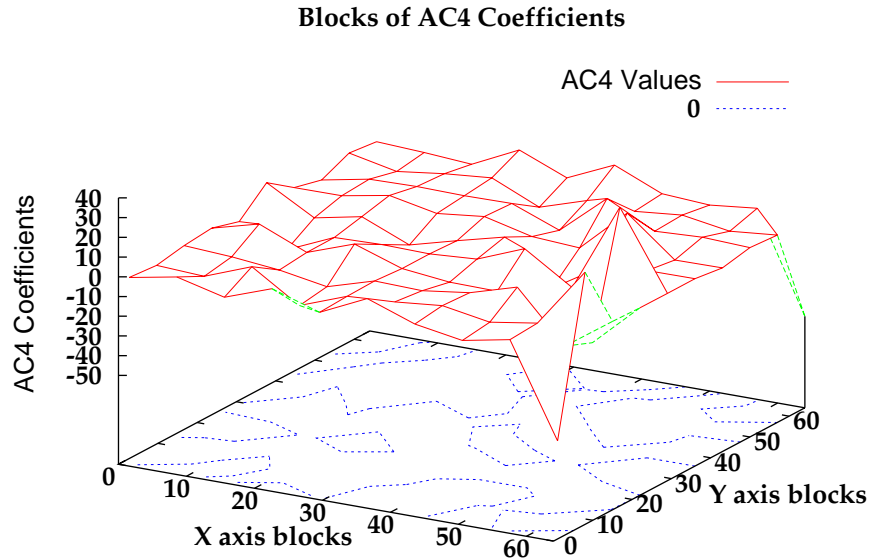


Figure 5.7: Plot of AC_4 coefficients

appended, followed by generalization. The above observations of the coefficients are used to classify into A , B , C , D and N categories. During the training phase we calculate the average of all the $AC1$, $AC2$, $AC3$ and $AC4$ coefficients of the watermark to be embedded, this average comes out to be approximately a value of 25 (a constant, which is represented as β in Section 5.1).

Once all such records are classified with respective labels, based on the value of PSNR it is classified into further class as A , B , C and R . If the value of PSNR is less than 10 it is identified as rejected class, if the value of PSNR is in between 10 to 20 the class is identified as C , if the PSNR value is in between 20 and 30 it is B class else for values greater than 30 class is A class which identifies that this is the best block for embedding a watermark.

These A category blocks are the blocks where we can embed data. The target vector will then have A , B , C and R only. The blocks that are so identified are the blocks having the PSNR in the required range of values and are in the low frequency band. Thus, these blocks are the blocks which are the potential candidates for embedding data. As the coefficients are low frequency coefficients robustness is also expected.

A generalized file as shown in Table 5.3 which consists of 20480 records is provided as input to the data mining software like Weka [87] to generate the decision tree. From this decision tree, the decision rules are formed for the identification of the A category blocks within the image for embedding. The block diagram for providing the training is as shown in Figure 5.8.

DC	AC1	AC2	AC3	AC4	PSNR based Classification	PSNR	Target Class
A	C	C	N	N	B	29.2	N
A	C	N	D	N	B	28.9	N
A	C	C	N	D	B	28.9	N
A	N	N	N	N	B	28.9	N
A	C	N	N	N	B	29	N
A	C	C	D	N	B	29.3	N
A	C	N	N	D	A	31.3	A
A	C	N	N	D	A	31.6	A
A	N	C	N	C	A	31.6	A
A	N	C	D	C	A	31.4	A
B	N	B	D	N	A	31.1	A

Table 5.3: Sample records with target class as ‘A’ and ‘N’

5.2.2 Algorithmic Details for The Application of ID3 in Digital Image Watermarking

The Data mining technique ID3 is used, which is a supervised learning algorithm, where the class label for each row should be known in advance. We have taken 4 classes A , B , C and R (Rejection) and the training was conducted for higher embedding strength watermark with 1.0 as scaling factor. Once the class label of the data is known then we use our decision tree algorithm on the data. The extracted knowledge is utilized for embedding the watermark at appropriate image block with different embedding strength.

A step wise algorithmic explanation to take decision for embedding watermark using decision tree induction algorithm (ID3) is as shown below:

After adding the data of all the training images, we create one column at the end of the file and named it to be Target, this column based on the PSNR classes will further classify it to A for Yes and N for No.

The above steps result into too many generalized vectors available with us as discussed earlier in Section 5.2.1.

Algorithm 1 Training Algorithm

- 1: Read the Training image using *imread* to provide the training.
 - 2: Read the watermark image.
 - 3: Open a file in append mode, to write all the vectors into it.
 - 4: Apply DCT2 on the watermark image.
 - 5: In the loop scan through every row of 8 in size and travel every column by step size of 8 for the complete image.
 - 6: **for** $i = 1 \rightarrow 4096$ **do**
 - 7: Apply DCT2 over the block.
 - 8: Apply zig-zag scan over the DCT coefficients and convert it into a vector of 64 coefficients.
 - 9: For vector coefficients $DC, AC1, AC2, AC3, AC4$ add a constant value beta to it, Keep the remaining vector elements i.e. DC and 6 to 64 as it is.
 - 10: Perform inverse zig-zag scan over the updated DCT coefficients.
 - 11: Apply Inverse DCT like $x = IDCT2(izigzag(y,8,8))$;
 - 12: Calculate PSNR value of the same block after adding beta to $DC, AC1, AC2, AC3, AC4$ coefficients.
 - 13: Generalize the coefficients into A, B, C, D and N using the generalization.
 - 14: Classify the block into using the PSNR Values such as value >30 to A , between 20 to 39 as B , 10 to 29 as C and less then 10 to R .
 - 15: Store the complete vector using *fprintf* function into a file.
 - 16: Go to next line within the file to store the new vector.
 - 17: **end for**
 - 18: Close the opened files and exit.
-

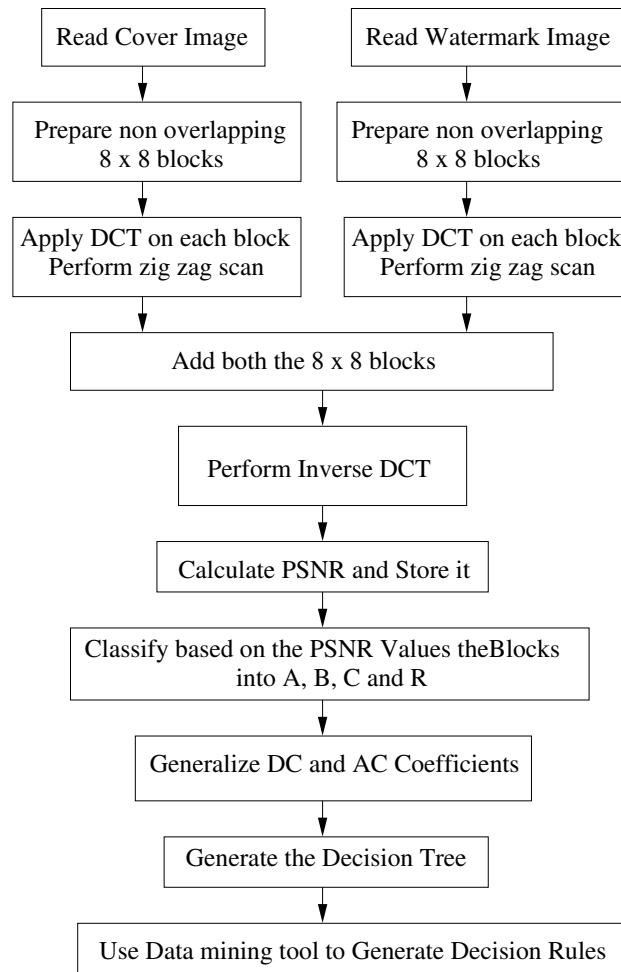


Figure 5.8: Training and identification of blocks

5.2.3 Introduction to Arnold Transform

The Arnold transformation (AT) stretches an image that is composed of $n \times n$ pixels and effectively wraps the stretched portions around to restore the original dimensions. It is of interest to people doing research in chaos and difference equations because it shows properties that many chaotic systems have. For instance, after a fixed number of iterative execution the original image is restored. It is then believed to be periodic with the given number of iterations. This interesting transform is shown in the Figure 5.9 (a) consisting of original logo image, Figure 5.9 (b) Arnold image in first iteration, followed by Figure 5.9 (c) Arnold image in second iteration, Figure 5.9 (d)

10th iteration and Figure 5.9(e) at 72nd iteration it resembles the original image: The



Figure 5.9: Arnold images after various iterations

following brief discussion covers the method used to generate the transformation. The transform that is applied to every pixel in the image is using the equation 5.9.

Here, the point is basically sheared. The *modulo* n is what causes the sheared image to wrap back around to restore the original $n \times n$ image.

As previously stated, images of certain size n , have a definite period of iterations where they will restore the original image. As you can see after the 2nd iteration, the image becomes barely discernable, but order re-emerges after few iterations. On the encoder side a few Arnold iterations could be executed and the remaining Arnold transform is to be applied during the extraction process till the original watermarked image is retrieved. Keeping this iteration number known only to the encoder and decoder makes the system highly secure against attacks, however, if the watermarked image is available to the intruder or to any individual, but the iteration count and location where embedding is done are not available, resulting into high level of security along with the robustness.

5.2.4 Embedding Algorithm

Figure 5.10 shows the block diagram for embedding the watermark in the image. The selected block numbers are encrypted and preserved in a file. These block numbers are referred to as key. These keys will be required to be furnished on the receiver side for necessary extraction of the watermark.

Algorithm 2: Watermark Embedding Algorithm Using Decision Rules

```

1: Read the watermark image using imread to perform Arnold Transform.
2: Read the image to be watermarked using imread() function.
3: Convert to double  $wm\_img \leftarrow double(z)$  , Open key.txt in write mode.
4: Initialize x, y, s, v1 and l2 to 1 and also decide upon the image plane to be R, G or B.
5: for  $k = 1 \rightarrow 4096$  do
6:    $dct\_img\_blk \leftarrow dct2(im2(y : y + 7, x : x + 7, 3))$ 
7:    $z\_dct\_blk \leftarrow zigzag(dct\_img\_blk)$ ,  $dccat \leftarrow z\_dct\_blk(1)$ ,  $ac1\_cat \leftarrow z\_dct\_blk(2)$ ,  $ac2\_cat \leftarrow z\_dct\_blk(3)$ ,  $ac3\_cat \leftarrow z\_dct\_blk(4)$  and  $ac4\_cat \leftarrow z\_dct\_blk(5)$ 
8:   if  $dc\_cat > 1000$  then
9:      $dc\_cat \leftarrow 'A'$ 
10:  else if  $dc\_cat \geq 200$  and  $dc\_cat < 800$  then
11:     $dc\_cat \leftarrow 'B'$ 
12:  else
13:     $dc\_cat \leftarrow 'C'$ 
14:  end if
15:  Similarly apply the generalization rules for AC1, AC2, AC3 and AC4 coefficients.
16:  Initialize mark_flag to zero.
17:  if  $dc\_cat == 'A'$  and  $ac4 == 'N'$  and  $ac2\_cat == 'B'$  and  $ac1\_cat == 'N'$  then
18:     $mark\_flag \leftarrow 1$ 
19:  end if
20:  Apply other rules in this manner to classify the A category blocks and mark such block with the mark_flag set to 1.
21:  if  $mark\_flag == 1$  then
22:    if  $l2 > 1024$  then
23:       $marked\_img(1 : 64) \leftarrow z\_dct\_blk(1 : 64)$ 
24:    else
25:      Store the block number in the key file using fprintf(f1, '
26:      Iterate q for 1 to 4 and if  $W(l2) > 0$  add the Beta value to  $z\_dct\_blk(q)$  and store it into marked_img(q) else Subtract the Beta value to  $z\_dct\_blk(q)$  and store it into marked_img(q)
27:       $l2 \leftarrow l2 + 1$ 
28:       $marked\_img(5 : 64) \leftarrow z\_dct\_blk(5 : 64)$ 
29:    end if
30:  else
31:     $marked\_img(1 : 64) \leftarrow z\_dct\_blk(1 : 64)$ 
32:  end if
33:  Perform inverse zigzag such that  $iz\_marked \leftarrow izigzag(marked\_img, 8, 8)$ 
34:  Perform inverse DCT using  $idct2(iz\_marked)$  and store it into  $im2(y:y+7,x:x+7,1)$ .
35:  Check for the last block in the row, if it is the last block go to the next row.
36:   $v1 \leftarrow v1 + 4$ 
37: end for
38: Display the block number and store the watermarked data to a file.
39: Read original and watermarked images, calculate PSNR and close the key file.

```

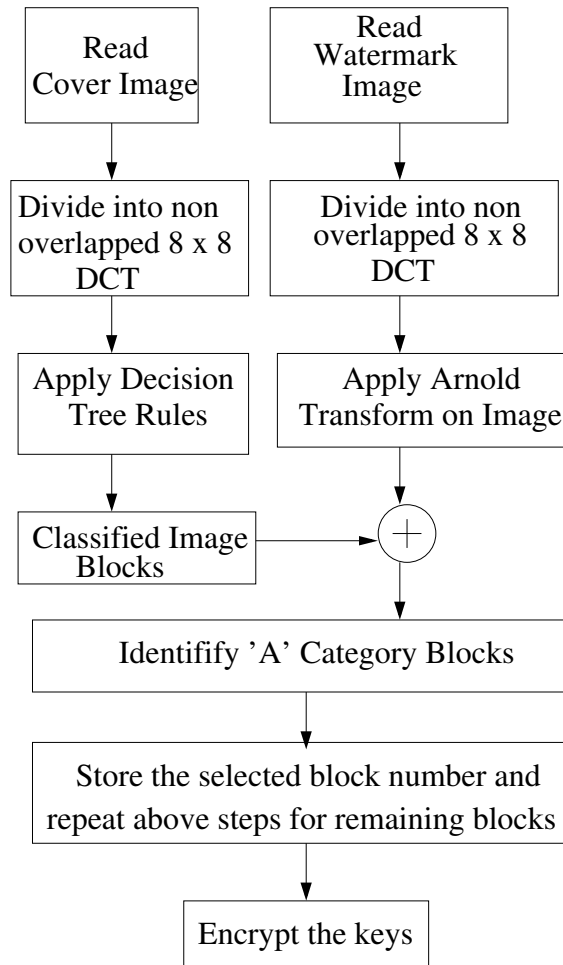


Figure 5.10: Block diagram for embedding of the watermark

5.2.5 Extraction Algorithm

- Use the keys to identify the block where the watermark was embedded.
- Take the difference of the watermarked block and the same cover image block.
- This block is just the scrambled information
- Using this approach the owner of the image can prove the identity of ownership of image because keys are available only with the owner.
- After all retrieval of blocks, apply the remaining steps of Arnold transform to get the watermarked image.

- As low frequency coefficients are utilized, robustness is expected. While marking an image, one wants to place the mark in the more robust areas of the image. The approach uses the areas with high gradient magnitude. In other words relatively strong edges with respect to the structure of the image and the luminance variances of the edges.

Algorithm 3: Watermark Extraction Algorithm

```

1: Read the watermarked image using imread.
2: Read the original watermark image and the original cover image.
3: Open a key file in read mode for retrieving the block numbers.
4: Initialize the variable value 1 to i, x1, y1, a, x, y and select the image plane.
5: Get the key element from the file and store it into key_ar
6: for  $i = 1 \rightarrow 4096$  do
7:   if  $i \leq 256$  and  $k == key\_ar(i)$  then
8:     Apply DCT2 over the block identified by the key element.
9:     Apply zigzag scan to form a vector.
10:    Apply DCT2 over the original image block.
11:    Apply zigzag scan to form a vector on the original image block.
12:     $rec\_img\_vector(a : a + 3) \leftarrow z\_marked(1 : 4) - z\_org(1 : 4)$ 
13:     $a \leftarrow a + 4$ 
14:    Repeat for all the key elements.
15:   end if
16:   Increment the block of the original image column wise and if the column reaches
      to the last block then go to the next row
17: end for
18: Reshape the vector using  $wm\_image \leftarrow reshape(rec\_img\_vector, 32, [])$ 
19: for  $i = 1 \rightarrow 32$  do
20:   for  $j = 1 \rightarrow 32$  do
21:     if  $wm\_image(i, j) \leq 0$  then
22:        $wm\_image(i, j) \leftarrow 0$ 
23:     else
24:        $wm\_image(i, j) \leftarrow Beta$ 
25:     end if
26:   end for
27: end for
28: Apply remaining Arnold Transform iterations.
29: Use  $i \leftarrow imresize(z, [3232], 'bicubic')$  in the iterations of Arnold Transform
30: Show the images to the user and save the recovered watermark.
31: Calculate the PSNR value for the original and reconstructed watermark.
32: Close the key file.

```

The same approach with a detailed Pseudocode is provided in the extraction algorithm.

5.3 Comparison and Contribution

The approaches for data hiding are performed by researchers under varying constraints. Due to desired characteristic of watermarking, we believe that traditional LSB algorithms are not suitable for watermarking. The proposed and implemented algorithm based on data mining and DCT satisfies number of ideal metrics for evaluation and the approach as being distinct from the work available in literature.

In general the research papers cover different aspect of watermarking systems. Since, there is no standardization in the approaches followed by various work, it adds some difficulty for the comparison. However, we have compared results of our algorithm with some of the well known work carried out by different authors in this area. Majority of the work covers LSB and transformations including DCT and DWT, which is listed in Chapter 3 and Chapter 4 as part of literature review.

The work done by various authors is on gray scale image and not on color image. The algorithm designed by us in this work is for both color images as well as for gray scale images. Most of the authors in their work use only one image to test their work, we have used six different color images for testing our algorithm. We have used a set of images having different characteristics such Lena image having uniform gray level distribution to cartoon image having constant colors in parts of images. The algorithm designed, contributes in the area of digital watermarking focusing on the characteristics of ideal watermarking system. The features of the algorithm along with the work done is highlighted in the Table 5.4.

Attacks/ Authors Contribution	[117]	[99]	[118]	[119]	[55]	[120]	[121]	[88]	[122]	[91]	[93]	[58]	[65]	[72]	[123]	Our Work
Rotation		Yes	Yes			Yes	Yes	Yes	Yes			Yes				Yes
Scaling		Yes	Yes	Yes		Yes	Yes	Yes		Yes		Yes		Yes		Yes
Shearing		Yes										Yes				Yes
Random Local Distortion													Yes		Yes	Yes
Cropping		Yes		Yes			Yes	Yes	Yes	Yes		Yes	Yes	Yes	Yes	Yes
Histogram Equalization	Yes				Yes											Yes
Salt and Pepper Noise	Yes						Yes									Yes
Gaussian Noise	Yes		Yes			Yes		Yes	Yes		Yes					Yes
Poisson Noise																Yes
Speckle Noise																Yes
Gamma Correction																Yes
JPEG Lossy Compression		Yes	Yes		Yes		Yes	Yes		Yes	Yes	Yes	Yes	Yes	Yes	Yes
Averaging Filter						Yes				Yes			Yes			Yes
Disk Circular Averaging Filter																Yes
Gaussian Filter		Yes		Yes			Yes		Yes							Yes
Motion Blur Filter																Yes
Sharpen Image Filter	Yes	Yes					Yes									Yes
Laplacian Filter																Yes
Prewitt and Sobel Filter																Yes
Laplacian of Gaussian Filter																Yes
Dithering of pixels				Yes								Yes				Yes
Median Filter	Yes				Yes	Yes	Yes			Yes					Yes	
Wiener filter			Yes							Yes						
Uniform Quantization															Yes	Yes
Minimum Variance Quantization																Yes
Contrast Stretching			Yes													Yes
Addition of Constant to an Image													Yes			Yes
Multiple Watermarks												Yes	Yes			Yes

Table 5.4: Comparison of work done by different authors with our algorithm

Proposed algorithm: The algorithm design is unique and makes use of data mining (ID3) and image processing operations to make it novel. Such an approach is used for the first time.

Visible/Invisible approach: Our algorithm is designed to work for invisible watermarking.

Perceptibility: The hidden watermark is imperceptible to the end user.

Keys: The algorithm uses keys for security.

Security: The algorithm is designed to include Arnold transform on the sender and receiver side for additional security.

Robustness: The algorithm is able to withstands various geometric attacks like rotation, scaling, shearing, addition of random local distortions, signal processing operations like applying filters of various types including average, disk circular, Gaussian, motion blur, sharpen, Laplacian, Prewitt and Sobel, Laplacian of Gaussian, JPEG lossy compression and addition of noise of different types including salt and pepper noise, Gaussian noise, Poisson noise, gamma correction. Along with color transformations, cropping, image enhancement and restoration processing such as histogram equalization, addition of constants to an image, dithering, quantization, contrast stretching, etc.

Universality: The algorithm is universal since, it could be further applied to other multimedia domain(s), with little or no modifications.

Multiple Watermarks: The algorithm is able to hide and maintain multiple watermarks successfully.

Scalability: The algorithm is designed to have enough embedding capacity in different image planes.

Image Planes: The algorithm is designed to have multiple watermarks in all the image planes.

Color/Gray scale images: The algorithm is designed to work with color as well as gray scale images of any type.

Change in file formats: Watermarks are retrievable after transformation of file formats like bmp, jpeg, tiff and png.

Statistics: The algorithm does not change the statistics of the cover image.

Tools used: The design outcome is tested using StegAlyzerAS version 3.2, the tool was unable to identify the hidden signatures.

Thus the algorithm represents a good solution to the protection of the ownership of data like images when it is made available in public domain.

5.4 Summary

In this Chapter a unified approach is used to design and embed digital watermark in the color images. This approach uses the Decision Tree and the Discrete Cosine Transform coefficients for the selection of blocks within the image. The approach is unique in the sense that it selects only those blocks whose PSNR values are high and based on the selection of such high PSNR blocks the embedding is performed. The design is scalable, since we can change the value of DC to get the required blocks, not only this, but it also permits the embedding to be performed in all the image planes. The technique designed is shown to be robust against intentional attacks. The design makes use of keys and Arnold transform to provide security. The Overall design proves to satisfy all the challenges of ideal digital watermarking system including perceptibility, security, robustness against various attacks, universality, multiple watermark insertion, file format conversions, statistics of the image and check with the benchmarking tools. The specific contributions and the work done are shown in Table 5.4 of Section 5.3. The experimental results obtained are elaborated in Chapter 6.

Chapter 6

Experimental Results

This Chapter covers experimental results and observations of applying theory or algorithm developed in Chapter 5. For implementation of the algorithm data mining tool namely Weka has been used. Matlab provided a fast implementing platform for verifying the concept. The initial Section of this Chapter deals with the results obtained as the outcome of data mining tool (Weka) and the subsequent results have been obtained in the Matlab software. Weka produced the Decision Tree to generate decision rules. This Chapter describes the results of various experiments to show the robustness of the algorithms against image processing operations.

6.1 Results from Datamining Approach

- Input to Weka software is the Excel file containing the following attributes: *DC*, *AC1*, *AC2*, *AC3*, *AC4*, Target
- Records produced for necessary training i.e. number of Instances is: 20,480

Generating the decision tree is done using Weka 3.4, an open source software in which an input file is provided which contains all the records and the target block is specified as 'A'. A partial decision tree obtained is as shown in Figure 5.2, where the

leaf nodes corresponds to target as ‘A’ class. The output generated from the software module is as follows:

Test mode: 10-fold cross-validation

Time taken to build model: 0.14 seconds

Description	Count/Result	Result in %
Total Number of Instances	20480	-
Correctly Classified Instances	17967	87.7395%
Incorrectly Classified Instances	2419	11.8115%
Unclassified Instances	94	0.459 %
Kappa statistic	0.597	-
Mean absolute error	0.1523	-
Root mean squared error	0.2838	-
Relative absolute error	46.19%	-
Root relative squared error	69.99%	-

Table 6.1: Summary of results (Output from Weka)

Based on stratified cross-validation the following results are obtained as the outcome of the training. Table 6.1 provides the summary of result produced by Weka. Table 6.2 gives the detailed accuracy by class and confusion matrix is shown in Table 6.3. the outcome of the training.

TP Rate	FP Rate	Precision	Recall	F-Measure	Class
0.964	0.432	0.894	0.964	0.928	N
0.568	0.036	0.806	0.568	0.667	A

Table 6.2: Detailed accuracy by class

N	A	Classified as
15549	583	N
1836	2413	A

Table 6.3: Confusion matrix

6.2 General Results and Discussion

For testing the designed algorithm well known images e.g. Lena, Bridge, Sail Boat, Sail Ship, Cartoon, etc. are used. Image blocks obtained in gray scale images, where embedding is carried out using Decision Tree rules are shown in Figure 6.1(a) and Figure 6.1(b) for Lena and Barbara images respectively. The watermarked images after embedding are shown in Figure 6.2(a) and Figure 6.2(b).

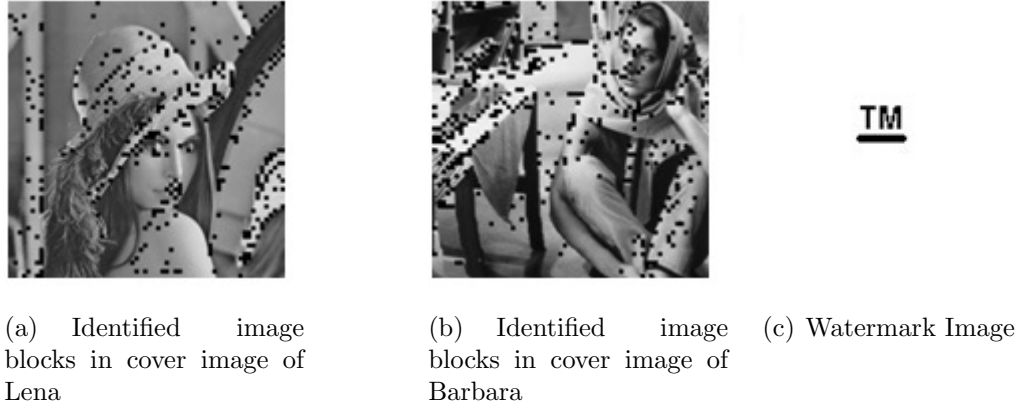


Figure 6.1: Gray scale Lena and Barbara images

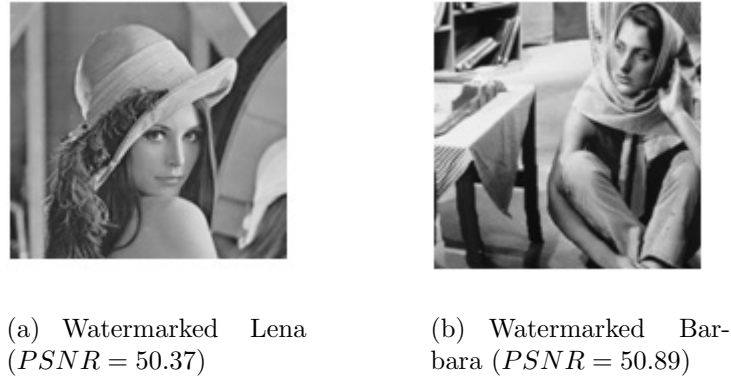


Figure 6.2: Watermarked Lena and Barbara images

Quality factor is the standard JPEG parameter. It is a number between 0 and 100; higher numbers mean higher quality (less image degradation due to compression), but the resulting file size, however would be larger. Figure 6.3 and Figure 6.4 shows the extracted watermark from various images with varying quality factors.

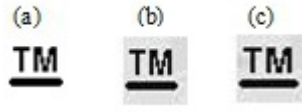


Figure 6.3: Extracted watermark



Figure 6.4: Extracted watermark with different JPEG quality factor

The correlation graph with variation in quality factor is as shown in Figure 6.5 It is observed that with the quality factor of 10, the watermark is almost destroyed.

Color logo image watermarking was also implemented and based on the training the

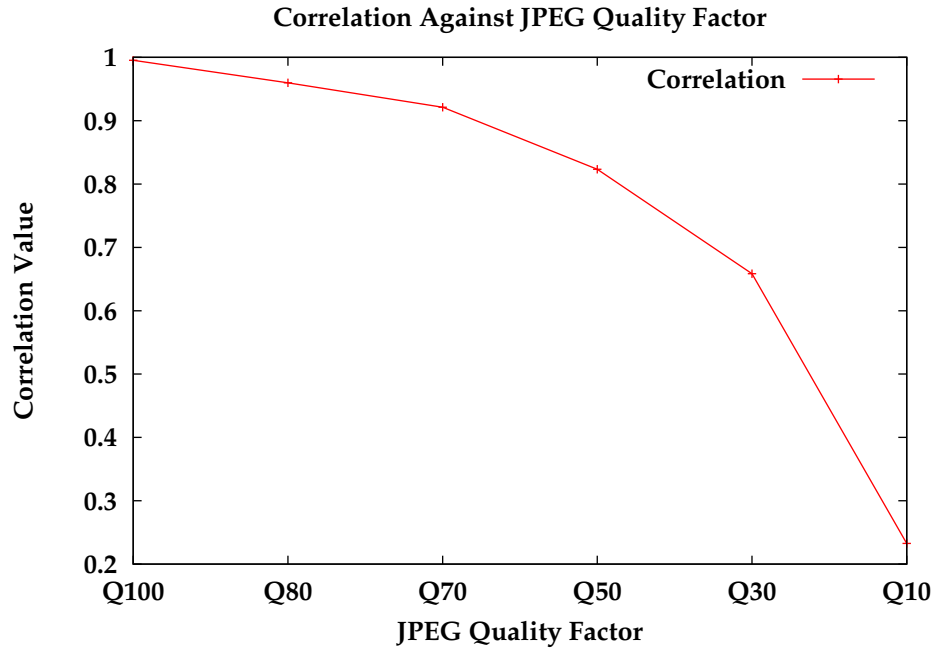


Figure 6.5: Correlation graph: Correlation Vs JPEG Quality factor

embedding was performed in to the R , G and B planes with different sets of images and the result obtained in PSNR is as shown below for images like Color Lena, Sail Boat, Cartoon, Bridge, Sea_Shore and Sail_Ship of size 512×512 as shown in Table

6.4 and Figure 6.6 shows the graphical representation for the same. Color planes

Images	Red Plane	Green Plane	Blue Plane
Color Lena	50.8609	47.9892	49.77
Sail Boat	47.32	47.31	47.43
Cartoon	47.43	49.75	47.55
Bridge	47.72	47.44	47.47
Sea_Shore	47.51	47.55	47.54
Sail_Ship	47.43	47.42	47.42

Table 6.4: PSNR values obtained in the RGB planes in color images after watermarking

in the Sail Boat, Color Lena, Cartoon, Bridge, Sea_Shore and Sail_Ship image where the embedding blocks have been identified is as shown in Table 6.5. As color image has R , G and B planes in each of these three planes, the intensity is different. So there is a need to choose proper threshold.

The threshold increases based on the intensity of the R , G and B components actually present to embed the watermark i.e. for reddish looking image such as Color Lena, we can keep higher threshold for getting suitable blocks in Red plane, but such higher threshold can not be kept for Green and Blue planes because this much higher threshold does not provide enough number of blocks for embedding.

In such cases threshold value can be lowered down to find sufficient number of blocks. This threshold according to our analysis is in the range of 500 to 1000.

Figure 6.8 shows the watermarked images of Figure 6.7 as the watermarked color logo inserted in the cartoon image in different image planes.

The visual results show that the data, could very well be inserted into the Blue Plane and some artifacts could be seen in the Red and Green plane in the background. Table 6.6 shows the result of the recovered watermark from the different planes. These

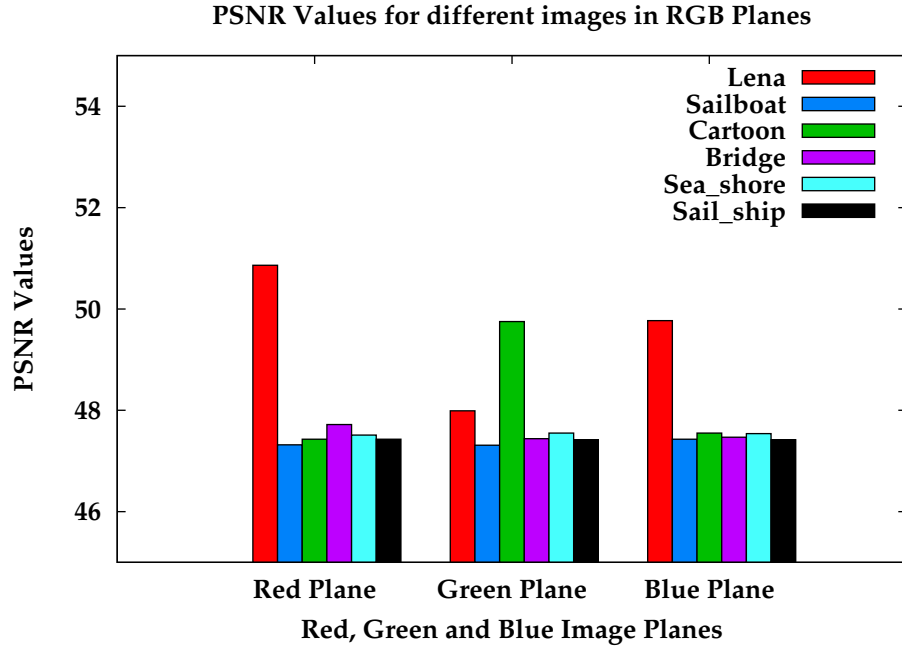


Figure 6.6: Graphical representation of the PSNR values obtained during watermarking in different image planes

C

Figure 6.7: Color logo watermark

watermarks are extracted after the storage of keys as shown in Section 6.7.

Table 6.7 shows a few cropped images of Color Lena, Sailboat and Cartoon along with the recovery of watermark. This attack is done intentionally to check for the recovery of watermark. The cropping is performed after the keys are stored within the image and the location of all these keys are preserved at distributed places as mentioned in Section 6.7.

A detailed analysis of how many blocks are available as ‘A’ category blocks for embedding in different images is as shown in Table 6.8. The watermark of size of 32×32 required such 256 blocks, however in the implementation it is observed that, this




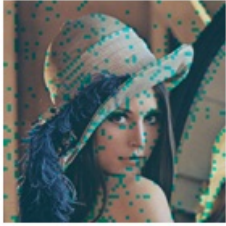
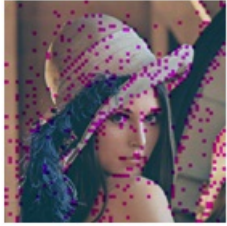
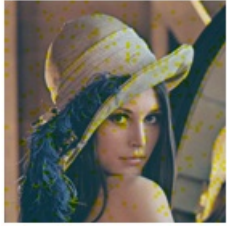
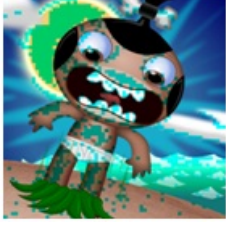
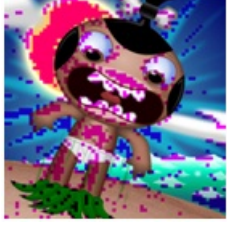
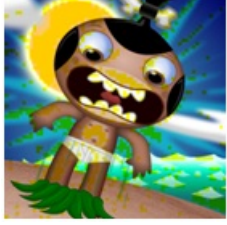
	R Plane	G Plane	B Plane
Sail Boat			
Color Lena			
Cartoon			

Table 6.5: Watermarking block where embedding is performed in the RGB planes

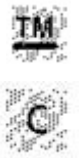


Red Plane (a)	Green Plane (b)	Blue Plane (c)
		

Table 6.6: Recovered watermark from the cartoon image in RGB plane

approach is having enough embedding places and few images have almost double the required capacity to embed. Thus depending upon the size of the watermark to be embedded this approach could be used in a scalable manner.

Table 6.9 shows the ‘A’ category blocks identified within the bridge image, which one can obtain for any images for embedding.



Figure 6.8: Sample watermarked images of cartoon image in RGB plane

Image	Cropped Image	Recovered Watermark
Color Lena (Cropped 20 % approximately)		
Sail Boat (Cropped 60 % approximately)		
Cartoon (Cropped 25 % approximately)		

Table 6.7: Cropped and recovered watermarked images from Blue plane

DC Coefficient	Color Lena	Sea Shore	Sail Ship	Bridge	Sail Boat	Cartoon
300	501	366	714	729	425	820
400	493	361	683	708	425	764
500	469	356	646	692	425	725
600	454	353	625	675	419	704
700	434	350	611	649	395	673
800	413	344	529	622	353	641

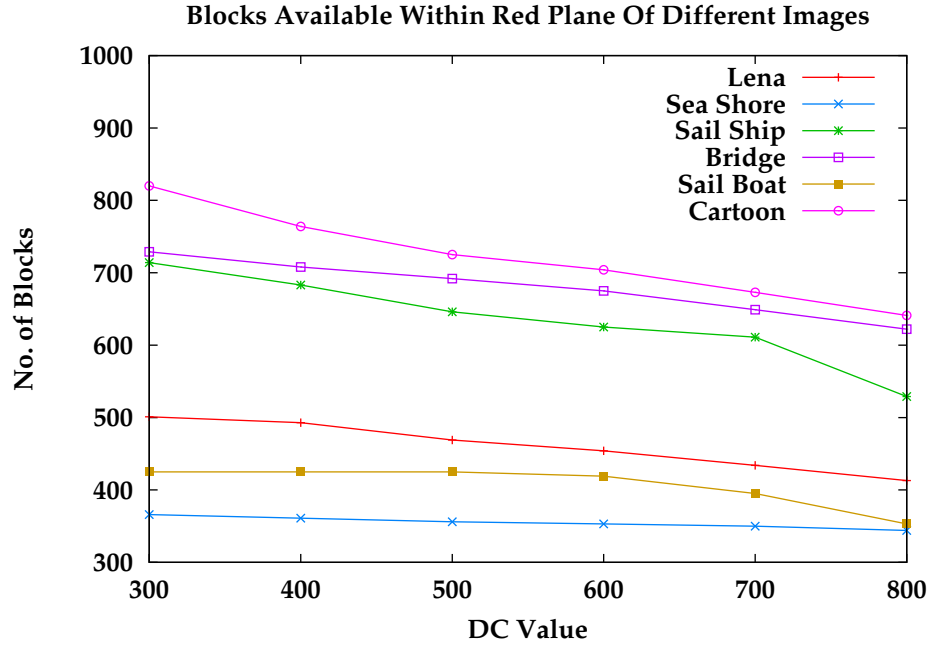
Table 6.8: Blocks identified as ‘A’ category blocks against different DC strength of test images in red plane.

DC Coefficient	Red	Green	Blue
300	729	742	693
400	708	729	683
500	692	712	661
600	675	689	637
700	649	666	608
800	622	640	574

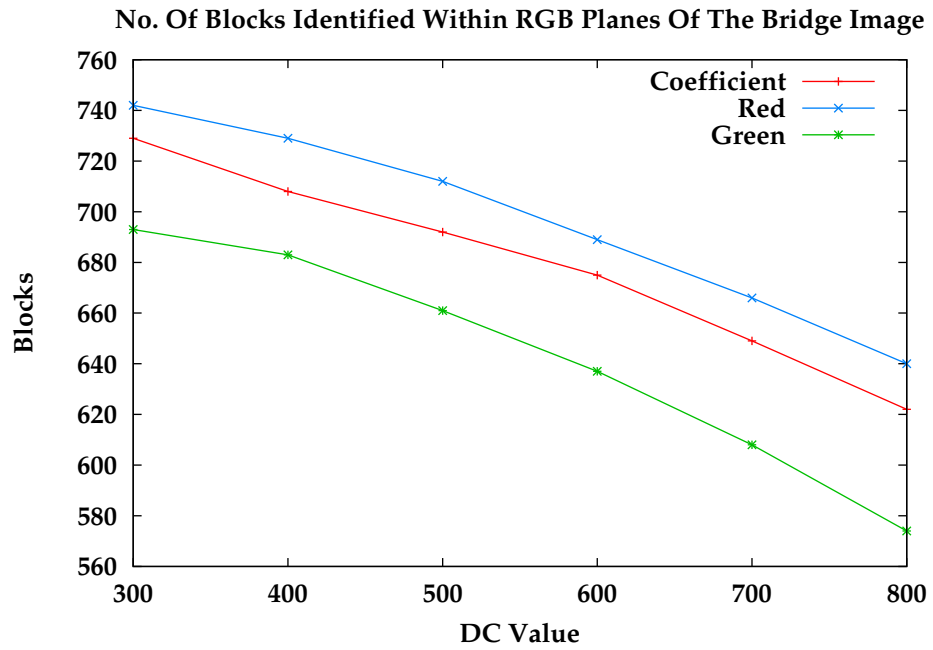
Table 6.9: Blocks identified against different DC strength in Bridge images in Red, Green and Blue image planes.

Figure 6.9(a) shows the plot against different DC value and the ‘A’ category blocks in various images and Figure 6.9(b) shows the plot against different DC value and the identification of ‘A’ category blocks in the bridge image in all the R , G and B image planes.

The keys generated by the embedding function are the place where the part of the watermark is stored. For a Bridge image the keys produced for embedding the watermark of size 32×32 is as shown below, these numbers are nothing but the block



(a) Graph showing all 'A' category blocks against different values of DC in all images based on Table 6.8



(b) Graph showing all 'A' category blocks against different values of DC in Different planes of the Bridge image based on Table 6.9

Figure 6.9: Blocks identified in image plane(s)

numbers identified row by row in multiples of 8 as DCT is applied on the image of size 8×8 thus the numbers generated are $1, 2, \dots, n$, where n is required number of blocks.

While embedding the first watermark following key set is obtained with DC as 800, let us say this key set as set-1

40,41,42,43,46,51,54,64,73,90,104,105,107,108,112,116,119,120,125,152,154,156,172,173,174,176,184,186,189,194,200,228,233,239,251,257,288,296,301,302,312,315,317,320,333,343,344,371,394,408,434,435,438,439,440,441,442,445,452,457,471,505,510,512,535,571,572,574,590,595,596,604,607,620,627,630,632,637,680,684,690,693,694,704,714,730,741,746,748,752,756,758,762,766,768,777,781,783,785,788,790,805,809,810,819,823,827,853,867,868,869,881,882,884,886,889,890,891,893,894,896,908,937,939,943,946,950,951,952,981,988,995,1001,1013,1014,1015,1017,1018,1023,1029,1036,1042,1047,1066,1074,1075,1085,1087,1088,1096,1099,1111,1113,1118,1121,1124,1129,1132,1135,1142,1144,1145,1148,1150,1171,1177,1178,1181,1188,1196,1198,1203,1204,1205,1206,1207,1212,1213,1234,1235,1238,1240,1248,1252,1255,1258,1267,1270,1271,1273,1276,1280,1298,1304,1311,1317,1318,1325,1327,1330,1334,1338,1341,1343,1344,1363,1367,1371,1375,1382,1390,1397,1401,1404,1406,1407,1425,1429,1430,1431,1433,1435,1436,1438,1440,1442,1444,1446,1450,1453,1459,1461,1463,1466,1468,1469,1470,1488,1491,1496,1503,1505,1511,1515,1526,1534,1536,1553,1557,1561,1563,1565,1566,1574,1583,1588

A similar set of keys is be obtained to embed within the watermarked image for doing double watermarking; here are the keys obtained with DC as 400, while performing embedding in the earlier watermarked image to embed a different watermark. Let us say this set of keys as set-2

36,40,46,51,64,86,100,116,119,120,149,155,156,164,176,184,186,228,239,283,294,302,312,315,346,349,351,361,422,434,435,438,445,479,505,559,571,572,613,620,627,632,637,677,680,690,693,694,704,741,748,752,758,762,768,785,805,815,819,827,870,871,882,884,889,890,

894,934,935,**937,943**,944,**946,950**,999,**1001**,1007,1008,**1014,1015,1023,1042**,1062,1063,1064,
 1072,**1074,1075,1085,1087,1088**,1126,1128,**1135**,1136,**1142,1144,1145**,1150,1190,1196,1198,
 1200,**1203,1205,1212,1213,1255,1258**,1262,1265,**1267,1270,1271,1273,1280,1286,1298**,
1318,1325,1327,1329,1330,1334,**1341,1343,1344**,1351,**1382**,1389,**1390,1401,1404,1407**,
 1448,**1450**,1457,**1461,1466,1468,1469,1470,1511**,1512,**1515**,1516,1521,1534,1545,**1553**,
1574,1576,**1583**,1585,1589,1590,1592,1593,1598,1611,1619,1620,1621,1624,1632,1635,1639,
 1640,1643,1645,1647,1650,1651,1653,1658,1661,1664,1665,1677,1688,1689,1690,1691,1693,
 1699,1704,1708,1714,1716,1723,1725,1726,1730,1739,1748,1751,1758,1761,1768,1769,1773,
 1774,1781,1787,1788,1810,1811,1813,1817,1818,1820,1823,1825,1831,1833,1834,1839,1841,
 1842,1845,1854,1874,1885,1886,1895,1897,1900,1903,1905,1915,1933,1948,1953,1965,1966,
 1967,1968,1969,1973,1981,1982,1988,2000,2005,2006,2012,2015,2019,2021,2022,2029,2030,
 2033,2048,2054,2070

The image after performing the double watermarking in the red plane of the bridge image is as shown in Figure 6.10. Figure 6.11 and Figure 6.12 shows the recovery of watermark using different key sets. The keys shown in bold in set-1 and set-2 are those blocks which holds data for multiple watermarks.



Figure 6.10: Double watermarked Bridge image in red plane



Figure 6.11: Recovered second watermark from Figure 6.10 using key set-2

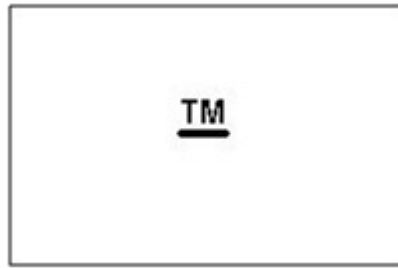


Figure 6.12: Recovered first watermark from Figure 6.10 using key set-1

Thus, we can say that the marked image itself could be used for embedding different watermark in the same image, resulting into multiple levels of watermarking, within the same image with different set of keys.

A similar set of keys for embedding a different watermark could be obtained for embedding in different planes of the image like green and blue, which further provides the flexibility for embedding multiple watermarks in different planes.

A very prominent feature observed in this implementation is that the previous watermark does not get damaged as new watermarks are embedded within the image. The overall effect is, that this mechanism allows multiple watermarks in multiple planes of the image.

However, a bigger size of watermark is to be embedded and if sufficient blocks are not available, the design will generate an error message in the encoder module and

will not proceed to embed. Normally the watermark to be inserted is always of less size, however for a special requirement of bigger size of watermark, the embedding algorithm can be modified by reducing the value of DC , so as to get more number of embedding blocks. By doing this, in the embedding side, one has to see that, by lowering the DC value to a very low level does not noticeably degrade the watermarked image. An attempt must always be to use higher DC value in the range of 700 to 1200. Thus, we can say that more the DC value the better the PSNR would be after embedding and lower will be the distortion observed by the Human Visual System. For performing multiple watermarking different values of DC in the encoder side will result into different identification of blocks and ultimately different set of keys. It is possible that the cover image itself is such that it is not giving good blocks for embedding even with the lower size of the watermark, in such cases embedding must be performed in different image planes.

6.3 Test for Robustness Against Various Image Processing Operations

Table 6.10 shows the test against histogram equalization and addition of Salt and Pepper noise of the order 2 % and 5 % having random distribution of pixels where noise is added. The extracted watermarks and their PSNR values are shown in Table 6.10. The watermarks are visually identifiable except one case of cartoon image. Similar thing is observed during addition of Gaussian noise, Poisson and Speckle noise, which are shown in Table 6.11. The presence of watermark is detected in all cases with different levels of noise and is visually recognizable.

Table 6.12 and Table 6.13 show the test for robustness against different values of gamma correction. The table also provide the recovered watermarks from the processed images as well as those from added noise. Watermarking is tested against gamma correction for values of γ ranging from 0.1 to 3.0. The results shows good

recovery of watermark for values of $\gamma > 0.4$ for all test images.

Figure 6.13 shows the plot of the PSNR values for the different images and different values of γ .



















Images	Histogram Equalization in Red plane	Recovered Water-mark	Salt and Pepper Noise 2 %	Recovered Water-mark	Salt and Pepper Noise 5 %	Recovered Water-mark
Color Lena	7.4975		8.1551		7.1716	
Sail Boat	6.6512		8.2391		7.3568	
Cartoon	7.3614		7.0195		6.3592	
Sea_Shore	7.216		8.2334		7.1893	
Bridge	7.1893		8.1385		7.4407	
Sail_Ship	7.2972		8.4654		7.343	

Table 6.10: PSNR values obtained after addition of noise in watermarked images



















Images	Gaussian Noise Mean 0 and Variance .01	Recovered Water- mark	Poisson Noise	Recovered Water- mark	Speckle Noise 3 %	Recovered Water- mark
Color Lena	7.9811		8.9569		6.96	
Sail Boat	7.9019		8.858		7.3753	
Cartoon	6.396		6.9433		6.0206	
Sea_Shore	7.638		8.6791		7.1938	
Bridge	7.7376		8.871		7.4501	
Sail_Ship	7.6776		8.9303		8.1385	

Table 6.11: PSNR values obtained after addition of noise in watermarked images











































Gamma Value	Color Lena PSNR	Recovered Water-mark	Sail Boat PSNR	Recovered Water-mark	Cartoon PSNR	Recovered Water-mark
0.1	5.4314		5.3087		6.3555	
0.2	5.8804		5.6845		6.6044	
0.3	6.2505		6.0037		6.6906	
0.4	6.5619		6.1585		6.7989	
0.5	6.7464		6.3445		6.9982	
0.6	6.8974		6.7224		7.0754	
0.7	7.0495		7.1057		7.279	
0.8	7.3614		7.3892		7.488	
0.9	7.5745		7.4313		7.5023	
1.0	9.0309		9.0309		8.2391	
1.5	7.5745		7.7025		7.6925	
2.0	7.4595		7.4595		7.488	
2.5	7.3753		7.3338		7.4548	
3.0	7.3109		7.2205		7.4079	

Table 6.12: PSNR values obtained against various parameter values for gamma correction











































Gamma Value	Sea_Shore PSNR	Recovered Water-mark	Bridge PSNR	Recovered Water-mark	Sail_Ship PSNR	Recovered Water-mark
0.1	5.473		5.4551		6.6122	
0.2	5.7318		5.7446		6.9307	
0.3	6.0003		6.141		7.0452	
0.4	6.2398		6.2937		7.0927	
0.5	6.4706		6.4108		7.1627	
0.6	6.7625		6.6277		7.1893	
0.7	6.9897		6.9939		7.27	
0.8	7.279		7.343		7.343	
0.9	7.4643		7.4785		7.4313	
1.0	9.0241		9.0309		9.0309	
1.5	7.5793		7.5696		7.6975	
2.0	7.2836		7.3476		7.6627	
2.5	7.1938		7.2474		7.5842	
3.0	7.1101		7.1188		7.5551	

Table 6.13: PSNR values obtained against various parameter values for gamma correction

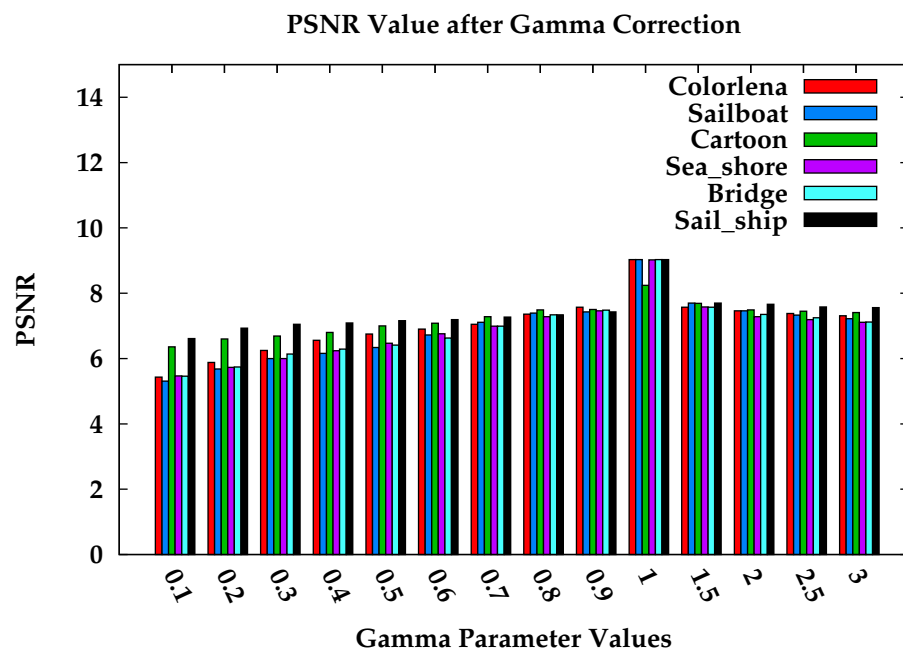


Figure 6.13: Plot of PSNR value after gamma correction

6.4 Geometric Attack

An attempt is made to check the algorithm against various geometric attacks like rotation, scaling, shearing and random local distortions. With all the geometric attacks applied, the algorithm is found to be sufficiently robust against such attacks. The outcome of these tests is shown in the following Subsections.

6.4.1 Attack against Rotation

The watermarked images are rotated by varying degrees in a counterclockwise direction around its center point. In the process of rotation, the output image becomes large enough to contain the entire rotated image. The pixels outside the rotated images are set to zero. We also cropped the rotated image to fit with the original size. The results with varying degree of rotation is shown in Figure 6.14. Result shows that the algorithm is able to withstand the attack against rotation. The watermark is clearly detectable in all cases and across test images.

6.4.2 Attack against Scaling

The watermarked images are scaled down to different size upto $1/2$, $1/4$ and even upto $1/8$. The watermarked image while scaling up, undergoes an interpolation using some methods. Our algorithm is tested with the bicubic and nearest neighbour interpolation methods. In the bicubic interpolation, the output pixel value is a weighted average of pixels in the nearest 4×4 neighborhood. Whereas in the nearest-neighbor interpolation, the output pixel is assigned the value of the pixel that the point falls within and no other pixels are considered. The results of scaling with different interpolation methods and varying sizes is as shown in Figure 6.15. The 512×512 image is down sampled by factor of 2, 4 and 8. The recovered watermarks are visually detectable for scaling by factor of 2. However the results may be acceptable for scaling by 4 and poor for scaling by 8.

6.4.3 Attack Against Shearing

A 2D affine transformation is applied on the watermarked image for performing horizontal and vertical shear of the image. The image size increased after applying shearing operation on the watermarked image. When horizontal and vertical both shearing operation is applied the image size gets reduced. We selected the actual image part and then converted it to the size of the actual image. The results of geometric shearing is as shown in Figure 6.16. Visual detection is possible for PSNR values above approximately 6 and it is also subject to type of image.

6.4.4 Random Local Distortion

An intentional attack is attempted to damage the watermarked images up to approximately 50 %, the recovered watermark shows noticeable recovery of watermark against such type of attack. Figure 6.17 shows the result obtained against random distortion attack applied across different images in different ways and the recovered watermark with the PSNR values obtained. The recovery of the watermark is visually recognizable considering the worst case damage of upto 50 % of different shapes and sizes distributed across the images.

6.5 Signal Processing Operations

The algorithm is checked for its robustness against various attacks like JPEG compression (Lossy and Lossless), various filtering attacks like averaging filter, disk averaging filter, Gaussian filter, motion blur filter, sharpened image filter, Laplacian filter, Prewitt and Sobel filter and Laplacian of Gaussian filter. Multiple watermarks are also attempted and found to be robust as shown in Section 6.2. Addition of noise is also tested and the design is found to be robust as shown in Section 6.3.

6.5.1 Attack against JPEG Lossy Compression

After the watermark is added within the cover image an attempt is made to check with various quality factor (Q) of JPEG compression algorithm in the range of 10 to 100. Figure 6.18 shows various values of Q and the PSNR obtained along with the recovered watermark. The presence of watermark is visually noticeable from the quality factor of 30 which increases proportionally with the increase in the value of quality factor. An attempt to check with lossless mode of JPEG is also performed on the watermarked images, the PSNR value of approximately 7.0927 is obtained with no loss in the recovered watermark with all the images. In the images like cartoon where there is a constant color, let us say blue as the dominating color and if we embed the watermark in the red plane of the image, which is not the dominating color, the watermark is visually perceived with the increase in the quality factor of 50 in the presence of noise.

6.5.2 Averaging Filter

An averaging filter of 3×3 has been used to perform averaging. The filter is shown in Table 6.14 and the results are shown in Figure 6.19. The results show very good recovery of watermark image in all cases.

0.1111	0.1111	0.1111
0.1111	0.1111	0.1111
0.1111	0.1111	0.1111

Table 6.14: Average filter mask

6.5.3 Disk Circular Averaging Filter

The algorithm was checked against a circular averaging filter (pillbox) within the square matrix of side $2*radius+1$. Our method retains the watermark with a radius of

3, 4 and 5 as shown in Figure 6.20. The recovered watermarks are visually noticeable for radius of 3 and degrades for radius greater than 4.

6.5.4 Gaussian Filter

Rotationally symmetric Gaussian low pass filter of size 3×3 with standard deviation σ (positive) is applied on the watermarked images. The default value for σ is 0.5. The design is tested against 0.1, 0.5 and 0.9 values of σ as shown in Figure 6.21. The recovered watermark images are very clear. Table 6.15, Table 6.16 and Table 6.17 provides the mask applied with varying values of σ as 0.1, 0.5 and 0.9 respectively.

0	0	0
0	1	0
0	0	0

Table 6.15: Gaussian filter mask with $\sigma = 0.1$

0.0113	0.0838	0.0113
0.0838	0.6193	0.0838
0.0113	0.0838	0.0113

Table 6.16: Gaussian filter mask with $\sigma = 0.5$

0.0673	0.1248	0.0673
0.1248	0.2314	0.1248
0.0673	0.1248	0.0673

Table 6.17: Gaussian filter mask with $\sigma = 0.9$

6.5.5 Motion Blur Filter

A motion blur filter is applied on the watermarked images which is a convolution with shifted version of the image. The linear motion of a camera represented by linear shift in pixels and with an rotation of angle θ in a counter clockwise direction is applied

over the images. Different values of length/shift and θ have been attempted on the watermarked images and the approach is found to be sufficiently robust for moderate shifts and rotation such as length of 10 and rotation up to 20 degrees, above which the watermarks is not recoverable as shown in Figure 6.22.

6.5.6 Sharpen Image Filter

A 3×3 unsharp contrast enhancement filter is applied on the watermarked images. The unsharp filter from the negative of the Laplacian filter with parameter α is applied on the watermarked images. The value of α controls the shape of the Laplacian and must be in the range 0.0 to 1.0. The default value for α is 0.2. An attempt is made to check with various values of α and is found to be highly robust as shown in Figure 6.23. Very good recovery of watermarks is observed in all the cases.

6.5.7 Laplacian Filter

A 3×3 filter approximating the shape of the two-dimensional Laplacian operator is applied on the watermarked images. The parameter α controls the shape of the Laplacian and must be in the range 0.0 to 1.0. The default value for alpha is 0.2. The presence of hidden message is noticeable in two cases, where as in other cases watermarks are not recovered. The results after applying Laplacian filter on the watermarked images is shown in Figure 6.24.

6.5.8 Prewitt and Sobel Filter

A 3×3 Prewitt and Sobel filter is applied on the watermarked images that emphasizes horizontal edges using the smoothing effect by approximating a vertical gradient. The presence of watermark as shown in Figure 6.25 is rarely noticeable in some of the images. This may be attributed to the fact that majority of the blocks identified by algorithm are near edges [9].

6.5.9 Laplacian of Gaussian Filter

A rotationally symmetric Laplacian of Gaussian filter of size 5×5 with standard deviation σ (positive) is applied on the watermarked images. Different values of σ has been attempted as shown in Figure 6.26. The design is observed to show most of the extracted images having some hidden message with σ in the range of 0.1 to 0.5.

6.5.10 Dithering of pixels

Dithering changes the colors of pixels in the neighbourhood, so that the average color in each neighbourhood, approximates the original RGB color. Here no dithering means quantization in colors only. The extracted watermarks do not give any meaningful information as shown in Figure 6.27.

6.5.11 Uniform Quantization

Uniform quantization and minimum variance quantization differ in the approach used to divide the RGB color cube. The tolerance determines the size of the cube shaped boxes into which the RGB color cube is divided. For example, specifying a tolerance of 0.1, the edges of the boxes are one-tenth the length of the RGB color cube. The approach is not found to be robust against uniform quantization as shown in Figure 6.28.

6.5.12 Minimum Variance Quantization

With minimum variance quantization, the color cube is cut into boxes (not necessarily cubes) of different sizes, the size of the boxes depend on how the colors are distributed in the image. The approach is not found to be robust against this type of quantization. Watermarks could not be recovered as shown in Figure 6.29.

6.6 Specialized Attack based on Knowledge of method

Changes to the color or intensity falls under this attack. The design is tested with contrast stretching and adding constant values to the watermarked images. The results obtained for contrast stretching are shown in Section 6.6.1 and the results for addition of constant value to the image plane are as shown in Section 6.6.2

6.6.1 Contrast Stretching

Contrast Stretching is applied on the watermarked images. The stretching is performed in the range of 0.01 to 0.99 and the results obtained are shown in Figure 6.30

6.6.2 Addition of Constant to an Image

Watermarked images were added with different constant value in the Red image plane. The design is found to be robust against this attack as shown in Figure 6.31. However in the case of cartoon image the watermark could not be detected as the red color is totally missing in the image.











































Images	Degree of rotation						
	5	15	30	45	60	75	90
Color Lena	7.0280	6.8396	6.9015	6.6986	6.7585	6.7786	7.0927
Recovered Watermark							
Sail Boat	7.0495	6.9224	6.9015	6.8602	6.8396	6.9643	7.0927
Recovered Watermark							
Cartoon	6.9643	6.7786	6.7184	6.6591	6.8192	6.9015	7.0927
Recovered Watermark							
Sea_Shore	6.9643	6.8602	6.8192	6.7989	6.7786	6.9015	7.0927
Recovered Watermark							
Bridge	6.9224	6.8602	6.7786	6.7585	6.7384	6.8192	7.0927
Recovered Watermark							
Sail_Ship	7.0495	6.8602	6.8396	6.8808	6.9433	7.0067	7.0927
Recovered Watermark							

Figure 6.14: PSNR and recovered watermark after geometric rotation of varying degree





































Images	512→256→512 Scaling by $\frac{1}{2}$ Bicubic	512→256→512 Scaling by $\frac{1}{2}$ Nearest	512→128→512 Scaling by $\frac{1}{4}$ Bicubic	512→128→512 Scaling by $\frac{1}{4}$ Nearest	512→64→512 Scaling by $\frac{1}{8}$ Bicubic	512→64→512 Scaling by $\frac{1}{8}$ Nearest
Color Lena	7.0927	6.6200	6.3923	5.8217	5.2600	4.7248
Recovered Watermark						
Sail Boat	7.0927	6.5812	6.3009	5.6159	5.4195	4.9575
Recovered Watermark						
Cartoon	7.0280	6.7184	6.4669	6.3555	5.7894	5.7543
Recovered Watermark						
Sea_Shore	7.0067	6.6986	6.3190	6.0376	5.2743	4.8655
Recovered Watermark						
Bridge	6.9643	6.4481	6.1936	5.7255	5.4492	4.8138
Recovered Watermark						
Sail_Ship	7.0927	6.8192	6.4669	6.3555	5.7414	5.5698
Recovered Watermark						

Figure 6.15: PSNR and recovered watermark after geometric scaling of different sizes



















Images	Horizontal Shear 0.5	Vertical Shear 0.5	Horizontal and Vertical Shear 0.5
Color Lena	6.7585	6.2469	5.5851
Recovered Watermark			
Sail Boat	5.9868	6.3190	5.8542
Recovered Watermark			
Cartoon	6.5237	6.7786	6.0037
Recovered Watermark			
Sea_Shore	6.5046	6.3739	5.6626
Recovered Watermark			
Bridge	6.0206	6.2648	5.9868
Recovered Watermark			
Sail_Ship	6.2828	6.6395	6.6591
Recovered Watermark			

Figure 6.16: PSNR and recovered watermark after geometrically shearing in horizontal, vertical and both sides













Images	PSNR	Intentional Attack (Approximately 50 % damage) Images scaled to ¼ size	Recovered Watermark
Color Lena	5.7733		
Sail Boat	5.8055		
Cartoon	6.6200		
Sea_Shore	6.1760		
Bridge	6.0547		
Sail_Ship	6.3555		

Figure 6.17: PSNR and recovered watermark after intentionally added random local distortions





























































Images	Q=10	Q=20	Q=30	Q=40	Q=50	Q=60	Q=70	Q=80	Q=90	Q=100
Color Lena	5.0244	5.6314	6.1063	6.3739	6.7184	6.7786	6.9433	6.9643	7.0927	7.0927
Recovered Watermark										
Sail Boat	5.1199	5.4940	5.8217	6.1236	6.2828	6.3009	6.4481	6.7184	6.9015	7.0280
Recovered Watermark										
Cartoon	5.9200	6.0547	6.0376	6.1236	6.0890	6.5427	6.5237	6.6986	6.9015	7.0067
Recovered Watermark										
Sea_Shore	5.0515	5.5698	6.1585	6.6986	6.8396	6.9224	7.0280	7.0710	7.0067	7.0927
Recovered Watermark										
Bridge	5.0110	5.4940	5.9533	6.5427	6.5427	6.7786	6.9224	7.0280	7.0927	7.0927
Recovered Watermark										
Sail_Ship	5.4940	5.5698	6.1936	6.3555	6.8192	6.9224	7.0067	7.0710	7.0927	7.0927
Recovered Watermark										

Figure 6.18: PSNR values for different images with varying quality factor (Q) and the recovered watermarks for JPEG compression attack







Images	PSNR	Recovered Images
Color Lena	7.0927	
Sail Boat	6.9433	
Cartoon	6.9855	
Sea_Shore	6.8396	
Bridge	6.8192	
Sail_Ship	7.0710	

Figure 6.19: Averaging filter attack with PSNR and recovered watermark images































Images	Radius for circular averaging				
	3	4	5	6	10
Color Lena	6.6005	6.0376	5.5242	5.3029	5.0515
Recovered Watermark					
Sail Boat	6.2828	5.8217	5.5393	5.2886	5.1754
Recovered Watermark					
Cartoon	6.6986	6.3009	6.0890	5.8217	5.4940
Recovered Watermark					
Sea_Shore	6.2648	5.8705	5.6470	5.3754	4.9708
Recovered Watermark					
Bridge	6.2469	5.8542	5.7414	5.5091	5.1199
Recovered Watermark					
Sail_Ship	6.6788	6.4108	6.1760	5.9533	4.9047
Recovered Watermark					

Figure 6.20: Disk circular averaging filter attack with PSNR and recovered watermark images



















Images	Standard Deviation (σ)		
	0.1	0.5	0.9
Color Lena	7.0927	7.0927	7.0927
Recovered Watermark			
Sail Boat	7.0927	7.0927	7.0710
Recovered Watermark			
Cartoon	7.0280	7.0280	7.0280
Recovered Watermark			
Sea_Shore	7.0927	7.0927	6.9855
Recovered Watermark			
Bridge	7.0927	7.0927	6.9224
Recovered Watermark			
Sail_Ship	7.0927	7.0927	7.0927
Recovered Watermark			

Figure 6.21: Attack of Gaussian filter on watermarked images with PSNR and recovered watermarks

























Images	Length = 2 Theta = 5	Length = 10 Theta = 10	Length = 10 Theta = 20	Length = 20 Theta = 20
Color Lena	7.0927	6.3372	6.3923	5.6159
Recovered Watermark				
Sail Boat	7.0927	6.2828	6.1410	5.4195
Recovered Watermark				
Cartoon	7.0280	6.6200	6.6591	5.7573
Recovered Watermark				
Sea_Shore	7.0927	6.3739	6.2469	5.4641
Recovered Watermark				
Bridge	7.0927	6.2648	6.1760	5.6470
Recovered Watermark				
Sail_Ship	7.0927	6.384	6.6986	5.9366
Recovered Watermark				

Figure 6.22: Motion blur attack with varying values of length and θ along with PSNR and recovered watermarks
















Images	α for Laplacian filter					
	0.1	0.2	0.4	0.6	0.8	0.9
Color Lena	7.0067	7.0067	7.0067	7.0067	7.0495	7.0495
Recovered Watermark						
Sail Boat	6.7384	6.7184	6.6986	6.6986	6.6986	6.6986
Recovered Watermark						
Cartoon	7.0067	6.9855	6.9643	6.9643	6.9855	6.9855
Recovered Watermark						
Sea_Shore	6.8602	6.8396	6.8396	6.8602	6.8808	6.8602
Recovered Watermark						
Bridge	6.7989	6.7786	6.7786	6.7786	6.7989	6.7786
Recovered Watermark						
Sail_Ship	6.9855	6.9643	6.9643	6.9643	6.9643	6.9643
Recovered Watermark						

Figure 6.23: Contrast enhancement attack with varying α along with PSNR and recovered watermarks































Images	α for Laplacian filter				
	0.05	0.1	0.2	0.5	0.9
Color Lena	4.8526	4.8396	4.8526	4.8526	4.8655
Recovered Watermark					
Sail Boat	5.0651	5.0651	5.0651	5.0924	5.0924
Recovered Watermark					
Cartoon	4.5763	4.5763	4.5763	4.5763	4.5763
Recovered Watermark					
Sea_Shore	5.0379	5.0379	5.0651	5.0651	5.0651
Recovered Watermark					
Bridge	5.1476	5.1476	5.1476	5.1337	5.1337
Recovered Watermark					
Sail_Ship	4.6376	4.6376	4.6376	4.6499	4.6499
Recovered Watermark					

Figure 6.24: Laplacian filter with varying α along with PSNR and recovered watermarks













Images	Prewitt	Sobel
Color Lena	5.8542	5.8870
Recovered Watermark		
Sail Boat	5.6470	5.7097
Recovered Watermark		
Cartoon	5.0244	5.1061
Recovered Watermark		
Sea_Shore	5.7414	5.8055
Recovered Watermark		
Bridge	5.8870	5.9200
Recovered Watermark		
Sail_Ship	6.0890	6.1410
Recovered Watermark		

Figure 6.25: Prewitt and Sobel filter along with PSNR and recovered watermarks

















































Images	Standard Deviation (σ)							
	0.1	0.2	0.3	0.4	0.5	0.6	0.7	1.0
Color Lena	4.5885	4.6499	4.7501	4.7882	4.7882	4.8396	4.8786	4.9443
Recovered Watermark								
Sail Boat	4.6623	4.7248	4.8786	4.9443	4.9842	5.0924	5.1061	5.1476
Recovered Watermark								
Cartoon	4.4801	4.5521	4.5885	4.5885	4.5763	4.5763	4.5763	4.6007
Recovered Watermark								
Sea_Shore	4.7627	4.8786	4.9310	4.9047	4.9575	5.0651	5.1061	5.1337
Recovered Watermark								
Bridge	5.0651	5.0924	5.0787	5.0651	5.0924	5.1337	5.1476	5.1476
Recovered Watermark								
Sail_Ship	4.0607	4.4920	4.5885	4.6253	4.6376	4.6376	4.6376	4.6623
Recovered Watermark								

Figure 6.26: Laplacian of Gaussian filter with varying σ along with PSNR and recovered watermarks





































Images	Dither 8	No Dither 8	Dither 512	No Dither 512	Dither 2048	No Dither 2048
Color Lena	5.1476	5.1814	5.3458	5.1061	4.8010	4.8916
Recovered Watermark						
Sail Boat	5.1615	5.1754	4.7882	4.8396	4.6997	4.8786
Recovered Watermark						
Cartoon	5.9366	5.9700	6.0206	5.9868	5.6782	5.7097
Recovered Watermark						
Sea_Shore	5.2743	5.1894	5.6939	5.3463	5.1894	5.1476
Recovered Watermark						
Bridge	5.1615	5.1615	5.0515	5.1061	4.6747	4.5400
Recovered Watermark						
Sail_Ship	5.1476	5.7894	5.3901	5.0244	4.8138	4.6007
Recovered Watermark						

Figure 6.27: Dithering and Nodithering on watermarked images with different colors with its PSNR and recovered watermarks











































Images	Tolerance for uniform quantization						
	0.005	0.01	0.1	0.3	0.5	0.7	0.9
Color Lena	4.2484	4.2484	5.4492	5.2175	5.1337	5.1061	5.1061
Recovered Watermark							
Sail Boat	4.4327	4.4327	4.5642	5.2600	5.1892	5.1754	5.1754
Recovered Watermark							
Cartoon	5.4641	5.4641	5.8542	5.4048	5.2175	5.3463	5.3463
Recovered Watermark							
Sea_Shore	4.3976	4.3976	5.4641	5.2743	5.2175	5.1894	5.1894
Recovered Watermark							
Bridge	4.4210	4.4210	4.4445	5.2458	5.2316	5.2175	5.2175
Recovered Watermark							
Sail_Ship	4.9310	4.9310	5.0244	5.5393	5.3754	5.3608	5.3608
Recovered Watermark							

Figure 6.28: Uniform quantization with varying tolerance with its PSNR and recovered watermarks





































Images	RGB Color cube					
	4	8	16	32	128	256
Color Lena	5.0651	5.1476	5.1615	5.5242	5.6470	5.8055
Recovered Watermark						
Sail Boat	5.1615	5.1615	5.2175	5.2743	5.6005	5.2600
Recovered Watermark						
Cartoon	5.4048	5.9366	5.9700	6.0037	5.9034	5.7894
Recovered Watermark						
Sea_Shore	5.2034	5.2743	5.3754	5.4195	5.9034	5.9034
Recovered Watermark						
Bridge	5.1615	5.1615	5.2034	5.4790	5.7894	5.6159
Recovered Watermark						
Sail_Ship	5.6005	5.1476	5.6159	5.8379	6.3190	5.7414
Recovered Watermark						

Figure 6.29: Minimum variance quantization with different colors cubes with the PSNR and recovered watermarks







Images	PSNR	Recovered Images
Color Lena	6.6591	
Sail Boat	6.2469	
Cartoon	7.0280	
Sea_Shore	6.9643	
Bridge	6.6005	
Sail_Ship	5.9868	

Figure 6.30: PSNR and recovered watermark after contrast stretching































Images	Constant addition				
	20	40	60	80	100
Color Lena	5.9700	5.7255	5.0651	4.7755	4.5642
Recovered Watermark					
Sail Boat	6.0037	5.9034	5.6782	5.5091	5.0924
Recovered Watermark					
Cartoon	6.0376	6.0206	5.9200	5.8379	5.6782
Recovered Watermark					
Sea_Shore	5.8217	5.7573	5.3901	5.2458	5.0651
Recovered Watermark					
Bridge	6.0037	5.9533	5.6005	5.4492	5.0515
Recovered Watermark					
Sail_Ship	6.0037	5.9868	5.9868	5.9366	5.9200
Recovered Watermark					

Figure 6.31: PSNR and recovered watermark after constant value addition

6.7 Storage of Keys

A separate software module is designed in Matlab to perform steganography in Region of Interest (ROI). Here the user have to select the object image which in our case is the same in which we have done the embedding of the watermark. The keys are required to be used on the receiver side. So the key files which are prepared by the embedding module is stored inside the text file. This text file is now to be inserted into this image. The decoder module will unhide this text file and use it further to extract the embedded watermark. Since the keys are not known to the end user, also the usage of Arnold transform makes the mechanism more secure.

A major feature of this module is that the user interface allows the end user to

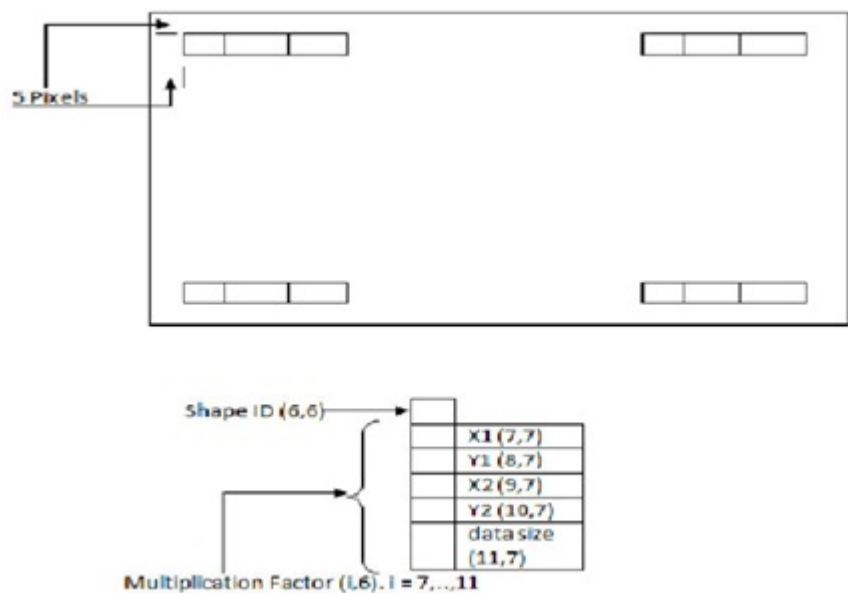


Figure 6.32: Embedding location

select any portion of the image to embed the keys. Also the information is managed in the image multiple times at different locations. The module under consideration identifies which portion of the image is attacked. If one of the corner information of the image is lost, then also the system locates the place where the keys are present.

This intelligence is provided into the system by embedding the location of storage at all the corners of the images in a specific way. Figure 6.32 provides a brief outline for the location where the storage of the keys is performed.

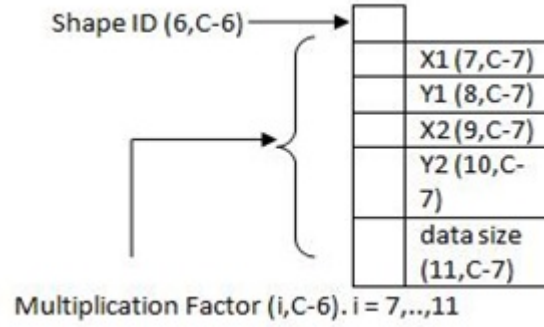


Figure 6.33: Upper right location for storage of keys within the image

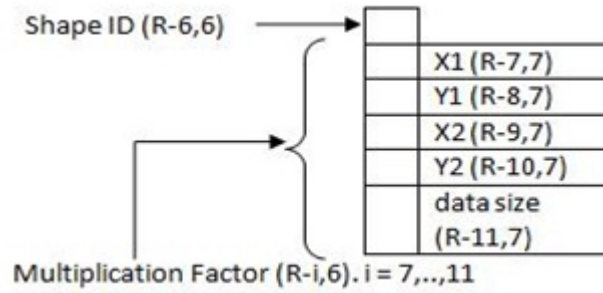


Figure 6.34: Bottom left location for storage of keys within the image

Figure 6.32 to Figure 6.35 show how the locations are stored into vector form at four different locations. Figure 6.36 shows the breakup of how 8 bit of information is stored within the R , G and B planes in the pixel of the image. Figure 6.37 and Figure 6.38 show interfaces provided to the user for storing the keys within the embedded image. So this particular image now contains both the watermark and also the keys. These keys are required on the decoder side to get back the watermark from the image.

The key storage module allows us to store multiple keys within the same image,

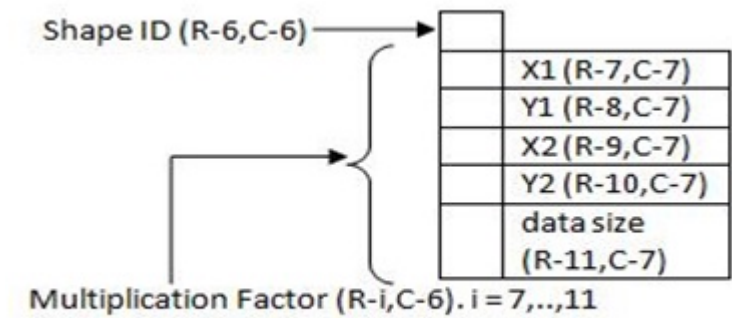


Figure 6.35: Bottom right location for storage of keys within the image

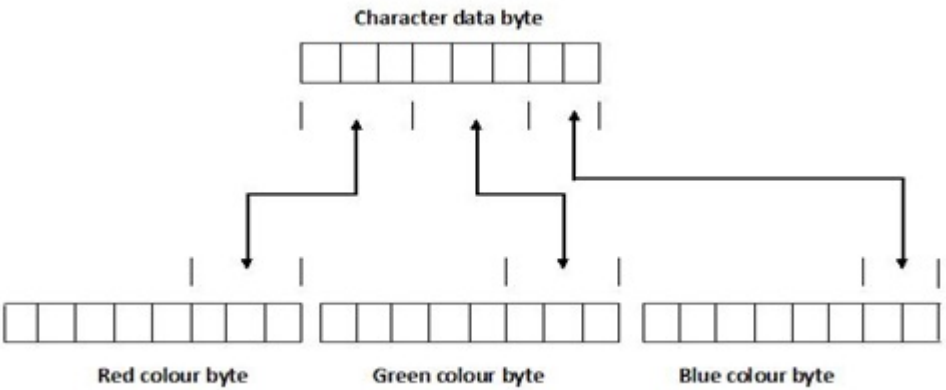


Figure 6.36: Simple LSB way to embed the keys into the pixel of R, G and B image plane.

and this is intimated to the user that the file in use is already having some information previously hidden. The design thus allows double steganography within the same image as shown in Figure 6.39.

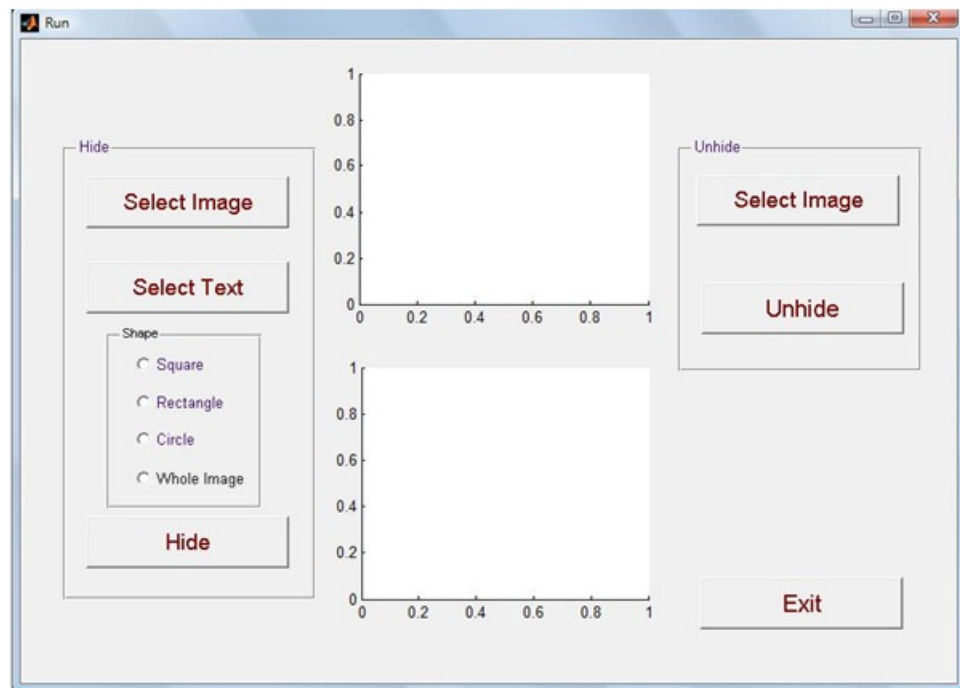


Figure 6.37: User screen to store keys within the image.

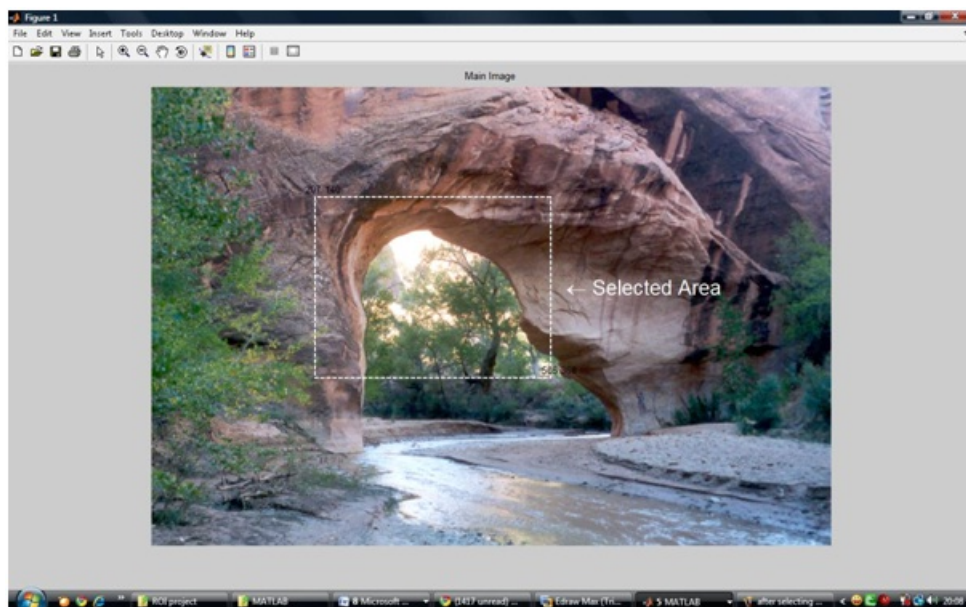


Figure 6.38: Screen after selection of points where storage will take place.

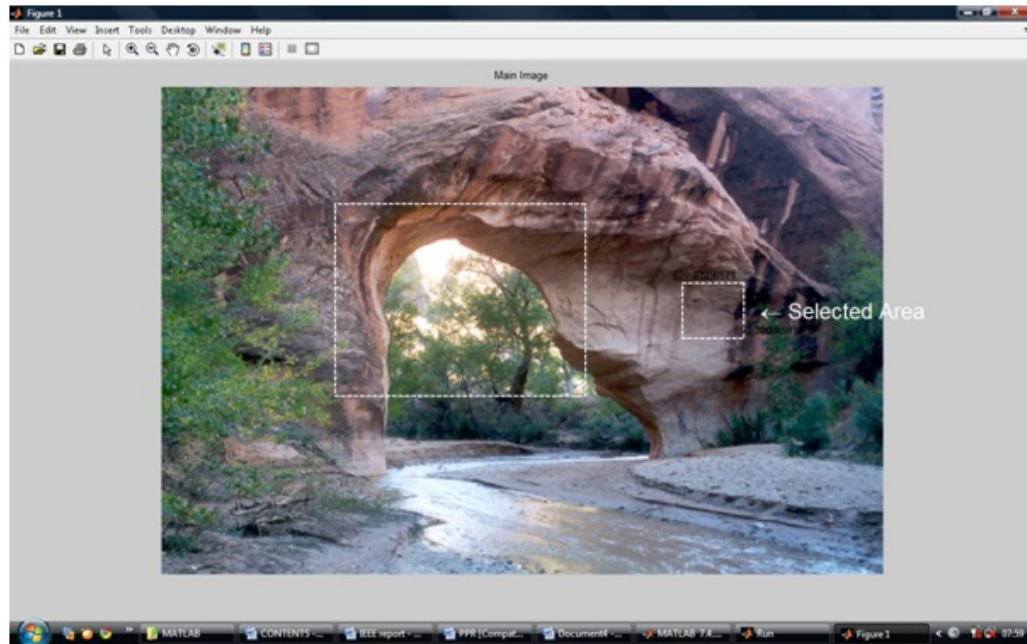


Figure 6.39: Double steganography within the same image.

6.8 StegAlyzer AS 3.2 Evidence Report

Steganalysis software like StegAlyzer AS 3.2 is a popular tool used for forensic investigations across the world. This tool is prepared by SARC and is available as a trial version from [11]. A case information to check that steganalysis software is able to detect the presence of hidden data or not, is performed and a report for the same is shown below. The report provided here, shows different scans and these scans, are the various signatures, which are available under the software to be tested individually. As the StegAlyzerAS 3.2 does not provide the option to test all signature, at just one go, individual signatures needs to be checked. All the signatures available under StegAlyzerAS were tested and a few test results, show that no signature were detected in the watermarked images.

StegAlyzerAS 3.2 Evidence Report: Case# 1

Case Information			
Case ID:	1	Organization Country:	INDIA
Investigator Name:	Samir B. Patel	Organization Name:	Insitute of Technology, Nirma University
Organization Address:	SG Highway, Charrodi	Organization State:	
Organization City:	Ahmedabad	Organization Zip:	382481
Organization Phone:	9427401616		

Case File: D:\Samir Everything\Samir\My Research papers\IJITST\blue\ReportStegAlyzer.ascf

Case Description: This is to check that steganalysis software is able to detect the presence of hidden data or not?

Case Notes:

Report Contents:

Click on a link to view the different sections of this report:

1. [Detected File Artifacts Table](#)
2. [Detected Applications Summary](#)
 - [1. D:\Samir Everything\Samir\My Research papers\IJITST](#)
 - [2. D:\Samir Everything\Samir\My Research papers\IJITST](#)
 - [3. D:\Samir Everything\Samir\My Research papers\IJITST](#)
 - [4. D:\Samir Everything\Samir\My Research papers\IJITST](#)
 - [5. D:\Samir Everything\Samir\My Research papers\IJITST](#)
 - [6. D:\Samir Everything\Samir\My Research papers\IJITST](#)
 - [7. D:\Samir Everything\Samir\My Research papers\IJITST](#)
 - [8. D:\Samir Everything\Samir\My Research papers\IJITST](#)
 - [9. D:\Samir Everything\Samir\My Research papers\IJITST](#)
 - [10. D:\Samir Everything\Samir\My Research papers\IJITST](#)
3. [Case Log](#)

File Artifact Table:

Artifact Name	False Positive	Application Name	Location Scanned	Verification Type	Verification Hash	Size	Found Under	Time Found
---------------	----------------	------------------	------------------	-------------------	-------------------	------	-------------	------------

This case does not contain any file artifacts.

[Return to Report Contents](#) [Return to Top of Table](#)

Detected Applications Summary:

- There were no steganography application artifacts detected during the course of this investigation.

[Return to Report Contents](#) [Return to Top of Listing](#)

Case Log [Return to Report Contents](#)

Time Stamp	Event Type	Event Item	Additional Information
13-05-2010 14:29:37	Created Case File	D:\Samir Everything\Samir\My Research papers\IJITST\blue\ReportStegAlyzer.ascf	Samir B. Patel
13-05-2010 14:29:40	Saved Case File	D:\Samir Everything\Samir\My Research papers\IJITST\blue\ReportStegAlyzer.ascf	Samir B. Patel
13-05-2010 14:29:40	Created Case File	D:\Samir Everything\Samir\My Research papers\IJITST\blue\ReportStegAlyzer.ascf	Samir B. Patel
13-05-2010 14:30:39	Scan Started	D:\Samir Everything\Samir\My Research papers\IJITST	Samir B. Patel
13-05-2010 14:30:39	Scan Completed	D:\Samir Everything\Samir\My Research papers\IJITST	Samir B. Patel
13-05-2010 14:30:57	Scan Started	D:\Samir Everything\Samir\My Research papers\IJITST	Samir B. Patel

The list is long which looks similar for all the remaining signatures; the display is truncated here for rest of the cases.

[Return to Report Contents](#) [Return to Top of Case Log](#)

Chapter 7

Conclusions and Future Work

7.1 Conclusions

The techniques have been developed to protect the ownership information of the digital content by watermarking on gray and color digital images integrating the concept of data mining and image processing.

Experimental result shows that data mining technique such as Decision tree induction (ID3) can be useful in identifying the image blocks suitable for watermarking in color images and even in gray scale images. It may be concluded that with the help of decision tree we could take decision, based on the various categories of AC as well as DC coefficients. Starting from the root node to leaf nodes, decision tree gives the class label which identifies image block required for embedding or for rejection.

In this work we have identified an area whereby decision tree can be used for performing Steganography and Digital watermarking applications in color and gray scale images.

As the design makes use of Arnold transform, which is periodic in nature, the water-

mark is scrambled before the embedding is actually performed. So, if the keys are compromised, the intruder is unaware of the number of iterations to execute on the receiver side, resulting into adequate level of security.

The method developed has been analyzed for the following performance metrics:

Capacity: Enough embedding capacity is obtained by this unique design to perform watermarking within images and the experimental results show that almost double the capacity is available for 32×32 size image to be embedded. However, the technique is scalable for the required size of watermark.

Multiple Embedding: Normally, the watermarking technique does not allow multiple watermarking, because by doing it, the previous watermark is not preserved and it gets damaged. It is observed that the design is successful in implementing a multiple watermarking in each plane. The design allows the owner to perform multiple watermarking on multiple image planes like Red, Green and Blue separately by making use of different DC thresholds.

Robustness: It is observed that the design is robust, because, we have identified all such ‘A’category blocks whose PSNR values are high during the training phase after a tentative embedding. So, the watermarking gain is already high in our case. Also, as the embedding is in very low frequency coefficient which are $AC1$ to $AC4$, robustness is preserved. Possibility of removing high frequency coefficients always remains during compression, but the low frequency values are never sacrificed.

Security: This work proves that the design is secure by making use of Arnold transform and encrypted keys. Even though the keys are available the number of iterations on the sender and receiver side is not known making the whole system highly secure.

Imperceptibility: The watermarked image does not show any artifacts as only those blocks are selected whose high value of PSNR is obtained before the actual embedding in ‘A’ category blocks. The Human Visual System (HVS) is not able to perceptually identify the embedded watermark(s).

Attacks: Different type of noise is added in the watermarked image to damage the watermarked image almost up to 60 to 70 %, however, the recovery of the watermark is observed to be of acceptable quality. The design of algorithm withstands most of the attacks like histogram equalization, intensity change like gamma correction, addition of noise like salt and pepper, Gaussian noise, Poisson noise and speckle noise. Various signal processing operations are applied to check against the robustness of the watermark. Signal processing operations applied to test the design includes JPEG lossy compression, average filters, disk average filter, Gaussian filter, motion blur filter, sharpened image filter, Laplacian filter, Prewitt and Sobel filter, Laplacian of Gaussian filter, dithering of pixels, quantization attack and multiple watermarks. Rotation, scaling, shearing and random local distortion is attempted to check various geometric attack which is found to be robust. As part of specialized attack based on the knowledge, addition of constant and contrast stretching is performed, the design is found to withstand these attacks.

The weakness of the algorithm was observed in operations like Prewitt and Sobel filter, dithering operations, uniform quantization and minimum variance color quantization.

Along with the above details separate software, based on Region of Interest i.e. choice based embedding of keys using LSB is designed with convenient GUI based interface. The selected pixel locations or coordinates are stored accordingly into the four different quadrants of the watermarked image. So, the design turns out to be a hybrid approach which combines watermarking and steganography. A test is done for the extraction of keys, after the cropping of the watermarked image and the approach

was successful in retrieving the keys even though some part of the image is cropped from multiple sides. These keys are required by the decoder to extract the watermark to identify the actual ownership. This module is also tested successfully to store multiple keys within the same watermarked image. Finally the watermarked image not only contains multiple watermark but also multiple keys for the retrieval without making the whole thing visually perceptible to the end user.

7.2 Future Work

Based on this research work and experiences few areas of further research work are identified as follows:

- Further enhancement could be possible, wherein the same design techniques could be applied on the video files of different types. Applying this technique on MPEG, H.264 and 3D video files could be a future direction for research.
- As the current trend is now towards multicore computing and the requirements for the real time video transmission is very much in demand, the ownership rights information also needs to be preserved. It could be a good future direction of research to deploy this design technique under multicore platforms like NVIDIA-CUDA, ATI Radeon, etc. To implement such algorithms for embedding watermarks in real time video, these algorithms will need to be implemented with real time constraints using high performance GPUs or FPGA based accelerators.
- Advanced techniques could also be applied for pruning the decision tree at leaf level to reduce the number of nodes and to scale down to a certain limit. Algorithms such as CART can also be used for continuously variable values.

Appendix A

Publications

A.1 Complete List of Publications

1. Samir B. Patel, “Hiding messages in images with steganography and digital watermarking techniques ”at National Technical Seminar and Symposium on Security and Communication Technology at Ahmedabad Management Association - March-2004.
2. Samir B. Patel, “Image based watermarking and authentication mechanism”at National Conference on Current Trends in Technology (NUCONE 07) during 29th Nov to 1st December, 2007 at Nirma University, Ahmedabad, Gujarat, India.
3. Samir B. Patel, “Digital watermarking for CD distribution containing multimedia data”at the National Conference of ENVISION October-2007, at S V Institute of Computer Studies Kadi, Gujarat, India.
4. Samir B. Patel and S. N. Pradhan “Proposed secure mechanism for Identification of ownership of undressed photographs or movies captured using camera based mobile phones”at the 2nd IEEE-International Conference on Digital Information Management ICDIM -2007 held at Lyon, FRANCE. pp no. 442-447

5. Samir B. Patel, “Lossy and Lossless Compression” at the National Conference of ENVISION October-2007, at S V Institute of Computer Studies Kadi, Gujarat, India.
6. Samir B. Patel and S. N. Pradhan “A proposed secure mechanism using parallelization techniques in steganography and digital watermarking for multiprocessor systems” at NUCONE-2008. A National conference on Current Trends in Technology, during 27th Nov to 29th November 2008 held at Nirma University, Ahmedabad, Gujarat, India.
7. Samir B. Patel, Jeet R. Patanji and Nisarg H. Patel “An Implementation of 3 Dimensional DCT for Compression of Video Sequences” at NUCONE-2009. A National conference on Current Trends In Technology, during 25-27 November, held at Nirma University, Gujarat, India.
8. Samir B. Patel, S. N. Pradhan and Tejas B. Mehta “A Novel approach Using Transformation Techniques and Decision Tree algorithm on images for performing Digital Watermarking” at the 4th IEEE International Conference on Internet Technology and Secured Transactions (ICITST)-2009 during 9th Nov. to 12th November 2009 at London UK.
9. Samir B. Patel, Vivek R. Vekaria and Mitesh K. Pithadiay, “Steganography in mobile phones on Multimedia Data” - at NUiCONE, 9-11 December 2010. A International conference on Current Trends In Technology, held at Nirma University, Ahmedabad, Gujarat, India.
10. Samir B. Patel, Daxa Vasoya and S. N. Pradhan, “Performance Optimization of Transformation Techniques - 2D DCT using NVIDIA CUDA”, at NUiCONE, 9-11 December 2010. A International conference on Current Trends In Technology, held at Nirma University, Ahmedabad, Gujarat, India.

International Journal papers

11. Samir B. Patel and Shrikant N. Pradhan, "Proposed Secure Mechanism for Identification of Ownership of Undressed Photographs or Movies captured Using Camera Based Mobile Phones" at Journal for Information Assurance and Security (JIAS), <http://www.mirlabs.org/jias/vol2-issue4.html> Page No.297 to 302
12. Samir B. Patel, S. N. Pradhan and Saumitra Umbegoankar, "A Novel Approach for Implementing Steganography with Computing Power Obtained by Combining CUDA with MATLAB" in International Journal of computer science and Information security(IJCSIS)-2009, Vol6. No2. pp 155-163, Nov.-09
13. Samir B. Patel and S. N. Pradhan, "The power of LSB: A Novel Approach to Secure Highly Confidential Documents/Files in Corporate or Institutes having Unsecured Workstation on the Network" at the International Journal of Computer Science and Information Security (IJCSIS)- November 2009, Vol. 6 No. 2 PP 133-137
14. Samir B. Patel, S. N. Pradhan and Tejas B. Mehta "A Unified Technique for Robust Digital Watermarking of Colour Images Using Data Mining and DCT" at the International journal of Internet Technology and Secured Transactions (IJITST) Inder-Science Publisher - Vol. 3. No. 1 PP-81-96 ,ISSN 1748-569_ (Print), ISSN: 1748 - 5703 (Online and Print)

A.2 Publications with Abstract

A.2.1 Samir B. Patel and Shrikant N. Pradhan “Proposed secure mechanism for identification of ownership of undressed photographs captured using camera based mobile phones” of the 2nd IEEE International conference on Digital Information Management, pp 442 - 447 , 28-31 October 2007. Followed by Journal Acceptance at the Journal of Information Assurance and Security 2 (2007) 297-302

Abstract: Cameras attached to mobile phones are becoming more and more common, and as we move towards 3G and Next Generation Networks, it has become more a standard feature of mobile phones. Over recent months there have been a few grandiose claims within the media about the potential misuse of phones with camera capability. Unfortunately some of these claims are not proved by available facts resulting into confusion and misunderstanding. It may suffice to say that some digital cameras are smaller, convenient and technology superior in image quality. This makes them easier to use in an unacceptable manner. Camera phones are designed to provide a means of transferring images via your mobile phone to complement voice or text based communication for business or personal reasons. Normally the youth gets attracted towards the sexual photography and watching movies on the mobile devices. Sometimes such movies get broadcast on the network like wild fire and it is available to all the community. It is indeed a difficult task to identify the user who has captured these photographs or movies and made it public. This paper focuses on a technique through which this problem can be solved. This technique, if implemented, on a mobile phone can really help the concerned authority to identify the culprits.

URL: [URL:www.mirlabs.org/jias/vol2-issue4.html](http://www.mirlabs.org/jias/vol2-issue4.html)

A.2.2 Samir B. Patel, Tejas B. Mehta and Shrikant N. Pradhan, “A Novel Approach Using Transformation Techniques and Decision Tree algorithm on Images for Performing Digital Watermarking”, of the 4th International Conference on Internet Technology and Secured Transactions ”, IEEE - Nov - 2009, ISBN: 978-1-4244-5647-5. A modified paper is acceptance at the International Journal of Internet Technology and Secured Transactions 2011 with the title “A unified technique for robust digital watermarking of colour images using data mining and DCT ”Vol3. No.1 pp. 81-96

Abstract: Digital watermarking is an emerging copyright protection technology. The paper presents a new robust watermarking technique based on combining the power of transform domain technique, the Discrete Cosine Transform (DCT) and the data mining technique such as Decision Tree Induction (ID3). The paper focuses on a technique through which the notion of decision tree can be applied on transformed vectors to build the decision tree.

We train the image blocks for deriving the classification tree. The resulting decision tree provides decision making rules to identify good quality image blocks for insertion of watermark. The implementation results have shown that the algorithm has an acceptable robustness against the JPEG compression and addition of noise.

Keywords: Digital Watermarking, DCT, Arnold Transform, Data Mining, ID3, Security, Copyright protection.

URL: <http://www.inderscience.com/browse/index.php?journalID=190&year=2011&vol=3&issue=1>

A.2.3 Samir B. Patel, Shrikant N. Pradhan, Saumitra U. Ambegaokar, “A Novel Approach for Implementing Steganography with Computing Power Obtained by Combining CUDA and MATLAB”In the International journal of Computer Science and Information Security. pp. 133-137, Vol. 6, Nov -09. ISSN 1947-5500.

Abstract:With the current development of multiprocessor systems, strive for computing data on such processor have also increased exponentially. If the multi core processors are not fully utilized, then even though we have the computing power the speed is not available to the end users for their respective applications. In accordance to this, the users or application designers also have to design newer applications taking care of the computing infrastructure available within. Our application is to use the CUDA (Compute Unified Device Architecture) as backend and MATLAB as the front end to design an application for implementing steganography. Steganography is the term used for hiding information in the cover object like image, audio or video data. As the computing required for multimedia data is much more than the text information, we have been successful in implementing image steganography with the help of technology for the next generation.

Keywords: CUDA, STEGANOGRAPHY, LSB.

URL: <http://sites.google.com/site/ijcsis/vol-6-no-2-november-2009>
<http://www.doaj.org/doaj?func=abstract&id=474321>

A.2.4 Samir B. Patel and Shrikant N. Pradhan, “An Approach to Secure Highly Confidential Documents of any Size in the Corporate or Institutes having Unsecured Networks”, In the International journal of Computer Science and Information Security. pp. 155-163, Vol. 6, Nov -09. ISSN 1947-5500.

Abstract: With the tremendous amount of computing and because of the wide usage of the internet it is observed that some user(s) are not able to manage their desktop with antivirus software properly installed. It is happening few times, that we allow our friends, students and colleagues to sit on our networked PC. Sometimes the user is unaware of the situation that there workstations are unsecured and so someone else could also be monitoring your flow of information and your most important data could go haywire, resulting into leakage of most confidential data to unwanted or malicious user(s). Example of some such documents could be question papers designed by the faculty members by various universities. Now a day most of the universities are having the biggest threat about the question papers and many other confidential documents designed by their faculty members. We in this paper present the solution to overcome such a situation using the concept of steganography. Steganography is a technique through which one can hide information into the cover object. This technique, if used, in positive direction could be of great help to solve such a problem and even other.

Keywords: Steganography, DCT, LSB, Digital Watermarking.

URL: <http://arxiv.org/abs/0912.0954>

A.2.5 Samir B. Patel, Mr. Vivek R. Vekaria and Mr. Mitesh K. Pithadiya,
“Steganography in Mobile Phones on Multimedia Data”at Nirma
University First International Conference on Current Trends and
Technology NUiCONE 2010. Held at Nirma University during 9-11
December 2010.

Abstract: Mobile camera was developed so that people do not have to carry any separate gadgets with them all the time, since it is integrated with their mobile phones; it helps in capturing images whenever and wherever wanted. But this mobile camera is also used for some wicked purposes i.e. nude photography followed by harassment to the individual. Hence there is an utmost need to prevent such type of happenings. We have tried to develop a module which will run when the image is saved in the memory or the file system of the mobile phone. It uses steganography to hide the IMEI, model number of the phone and the date on which the photograph is taken. When this image is forwarded via Bluetooth, the same module will get executed and the IMEI, model number and date on which image it is forwarded will be stored in the image. Objective of such an approach is to capture the original culprit or the ultimate source who captured the images or videos by backtracking and also the ones who helped in forwarding of such images and prevent such misuse of camera based mobile phones.

Appendix B

Prerequisite

B.1 DCT Basics

Formally, the discrete cosine transform is an invertible function $F : \Re^N \rightarrow \Re^N$ or equivalently an invertible square $N \times N$ matrix [16]. The formal definition for the DCT of two-dimensional sequence of length N is given by the following formula [16]:

$$C(u, v) = \alpha(u)\alpha(v) \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} f(x, y) \cos \left[\frac{\pi(2x+1)u}{2N} \right] \cos \left[\frac{\pi(2y+1)v}{2N} \right] \quad (\text{B.1})$$

The inverse of two-dimensional DCT for a sample of size $N \times N$ is given by:

$$f(x, y) = \sum_{u=0}^{N-1} \sum_{v=0}^{N-1} \alpha(u)\alpha(v) C(u, v) \cos \left[\frac{\pi(2x+1)u}{2N} \right] \cos \left[\frac{\pi(2y+1)v}{2N} \right] \quad (\text{B.2})$$

We can separate equation B.1 in the following form:

$$C(u, v) = \alpha(u)\alpha(v) \sum_{x=0}^{N-1} \cos \left[\frac{\pi(2x+1)u}{2N} \right] \left\{ \sum_{y=0}^{N-1} f(x, y) \cos \left[\frac{\pi(2y+1)v}{2N} \right] \right\} \quad (\text{B.3})$$

To perform the 2D DCT of length N , the cosine values are usually pre-computed offline. A 2D DCT is implemented on image by applying DCT on rows and columns of the input image by using equation B.3. So the whole 2D DCT process can be

represented in matrix notation using the following formula:

$$C(u, v) = A^T X A \quad (\text{B.4})$$

B.2 DWT : Discrete Wavelet Transform

The DWT inherently provides a multi-resolution image representation while also improving compression efficiency due to good energy compaction and the ability to decorrelate the image across a larger scale[124], [16]. DWT is important and computationally demanding part of JPEG2000 algorithm. JPEG2000 standard specifies use of LeGall (CDF) 5/3 DWT filter-banks for lossless compression process and Daubechies-Feauveau (CDF) 9/7 DWT filter-banks for lossy processing.

The basic idea of the wavelet transform is to represent any arbitrary function f as a weighted sum of functions, referred to as wavelets. Each wavelet is obtained from a mother wavelet function by conveniently scaling and translating it. The result is equivalent to decomposing f into different scale levels (or layers), where each level is then further decomposed with a resolution adapted to that level.

To do the wavelet transform on an image, we consider the n pixels in one row as level 0 approximation of a function. DWT decomposes image into a number of low and high sub bands at different levels of resolution. Two dimensional DWT is performed by applying the one-dimensional DWT row-wise and then column-wise in each component as shown in Figure B.1

In the first level of decomposition, four subbands LL1, HL1, LH1 and HH1 are created. Definition of these subbands is as follows:

- **LL:** low subbands for both row and column filtering.
- **HL:** high subbands for row filtering and low subbands for column filtering.

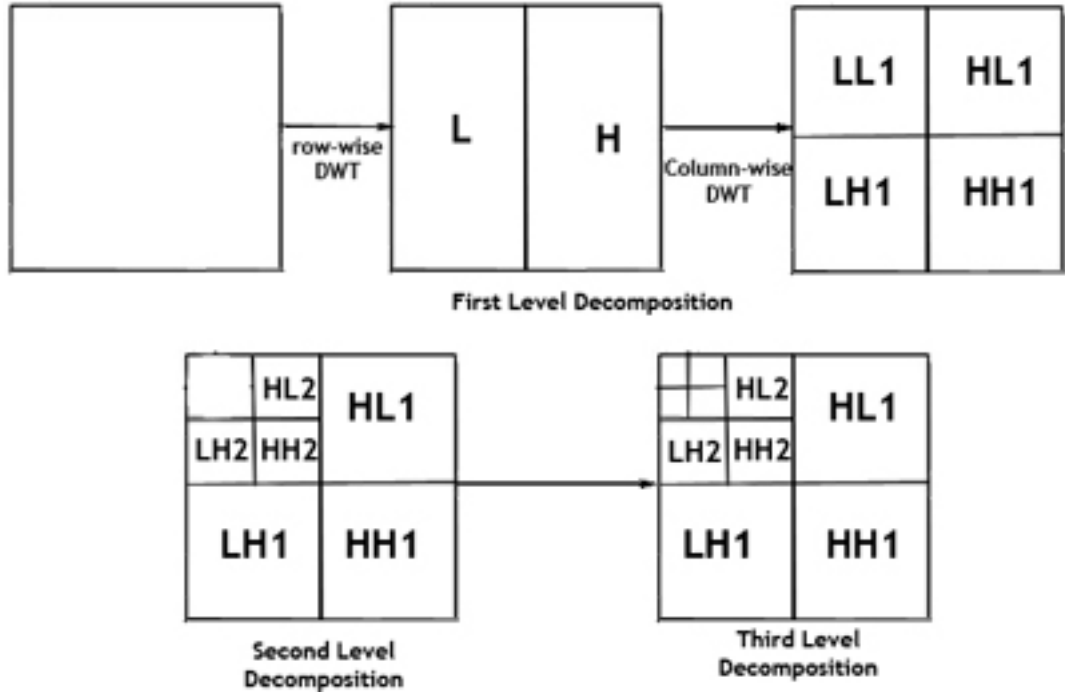


Figure B.1: 2D DWT decomposition

- **LH:** low subbands for row filtering and high subbands for column filtering.
- **HH:** high subbands for both row and column filtering.

The low-pass sub band (LL1) represents the original component 2:1 sub sampled in both horizontal and vertical directions. It is a low-resolution version of the original component. The other subbands HL1, LH1 and HH1 represent down sampled residual versions of the original image necessary for the reconstruction of the original image. DWT can be applied on LL1 sub band repeatedly to produce four other subbands LL2, HL2, LH2 and HH2 with the same meaning as corresponding subbands originated in first step. The Figure B.2 shows 2-level 2D wavelet transform. DWT can be applied up to 32 times in the JPEG2000 standard. Nevertheless, more than five levels of decomposition do not bring any benefits.

There are basically 2 approaches to implement DWT.

1. **The convolution:** It performs series of dot products between the two filter

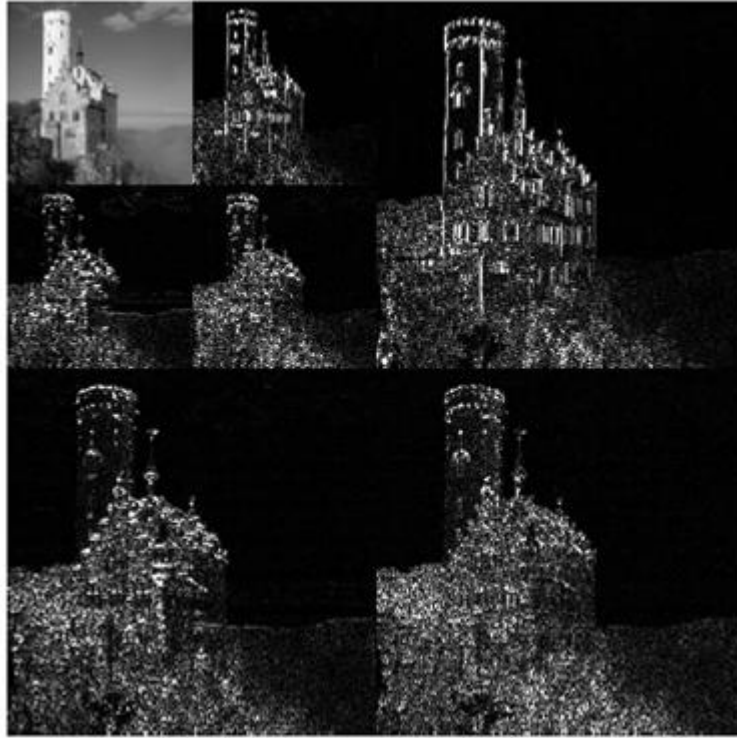


Figure B.2: Decomposition of Image

masks.

2. **The lifting scheme:** In this method odd sample values of the signal are updated with a weighted sum of even sample values, and even sample values are updated with a weighted sum of odd sample values.

B.3 PSNR

PSNR stands for Peak Signal to Noise Ratio [16], The designers of image processing methods require a standard metric to measure the quality of the reconstructed objects compared with the original ones. The better a reconstructed image resembles the original one, the bigger should be the value produced by this metric. As, we are performing transformation using DCT, there is a possible loss of data as bitwise check of results may fail, this is because of possible differences in floating point operations sequences in both implementations or due to differences in floating point units [16]. Therefore, for lossy transformation the consistency checking is performed using the objective image similarity metric PSNR. PSNR is defined for two images I and K of size M x N as:

$$PSNR(I, K) = 20 \log_{10} \frac{MAX_I}{\sqrt{MSE(I, K)}} \quad (B.5)$$

Where I is the original image, K is a reconstructed or noisy approximation, MAX_I is the maximum pixel value in image I and MSE is a mean square error between image I and K :

$$MSE(I, K) = \frac{1}{M} \frac{1}{N} \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} \| I(i, j) - K(i, j) \|^2 \quad (B.6)$$

PSNR is expressed in decibel scale and takes on positive infinity for identical images. In image reconstruction typical values for PSNR vary within the range. PSNR value between 30 to 50 calculated from two images that were processed on diverse devices with the same algorithm says the results are practically identical.

Appendix C

Test Images

All the images shown in this part of text are scaled down to 65% of the original images.



Figure C.1: Lena Gray Scale [2]



Figure C.2: Barbara Gray Scale [3]



Figure C.3: Lena Color [4]



Figure C.4: Sea Shore Color



Figure C.5: Cartoon Color Image [5]



Figure C.6: Sail Ship Color [6]



Figure C.7: Bridge Color Image [6]



Figure C.8: Sail Boat Color Image [7]



Figure C.9: Watermark Image

References

- [1] “Block diagram of steganography.” http://debut.cis.nctu.edu.tw/Demo/HighCapacityImageSteganographicTool/stego/stego_1.jpg.
- [2] “Lena gray scale image.” <http://www.ece.rice.edu/~wakin/images/lena512.bmp>.
- [3] “Barbara gray scale image.” <http://decsai.ugr.es/~javier/denoise/barbara.png>.
- [4] “Lena image.” http://www.petitcolas.net/fabien/watermarking/image_database/.
- [5] “Cartoon image.” <http://dev.boltcreative.com/pocketgod/IdleHands2Icon512x512.jpg>.
- [6] “Sail ship, bridge image.” <http://www.google.co.in/imghp?hl=en&tab=wi>.
- [7] “Sail boat image.” http://t1.gstatic.com/images?q=tbn:ANd9GcQ54EZhKUHdn_BcXj-Ur4bC6pRs1NAX_GmLUqpdo8neh0hY66BEmg.
- [8] N. F. Johnson, Z. Duric, and S. Jajodia, *Information Hiding Steganography and Watermarking - Attacks and Countermeasures*. Kluwer Academic Publishers, 2000.
- [9] S. B. Patel, T. B. Mehta, and S. N. Pradhan, “A novel approach using transformation techniques and decision tree algorithm on images for performing dig-

- ital watermarking,” *International Journal of Internet Technologies and Secure Transactions*, vol. 3, pp. 81 – 96, Apr. 2011.
- [10] S. C, “A hidden bits: A survey of techniques for digital watermarking,” tech. rep., <http://web.vu.union.edu/~shoemakc/watermarking/>, 2002.
- [11] “Steganography analysis.” www.sarc-wv.com.
- [12] M. Kutter and F. Hartung, *Introduction to Watermarking Techniques in chapter 5*. Northwood, MA: Artec House publisher, Dec. 1999.
- [13] I. Cox, M. Miller, and J. Bloom, *Digital Watermarking*. San Fransisco: Morgan Kaufmann, 2002.
- [14] N. D. Memon and P. W. Wong, “Protecting digital media content,” *Communications ACM*, vol. 41, no. 7, pp. 34 – 43, 1998.
- [15] G. W. Braudaway, K. A. Magerlein, and F. Mintzer, “Protecting publicly-available images with a visible image watermark,” *Proceedings of the SPIE International Conference Electronic Imaging*, vol. 2659, pp. 126 – 133, Feb. 1-2 1996.
- [16] D. Salomon, *Data Compression, The Complete Reference*. Springer Press, 2nd ed., 2001.
- [17] S. B. Patel and S. N. Pradhan, “Proposed secure mechanism for identification of ownership of undressed photographs or movies captured using camera based mobile phones,” *Journal for Information Assurance and Security*, vol. 2, pp. 297 – 302, 2007.
- [18] M. Arnold, M. Schmucker, and S. D. Wolthusen, *Techniques and Applications of Digital Watermarking and Content Protection*. Artech House, 2003.

- [19] S. B. Patel and S. N. Pradhan, "An approach to secure highly confidential documents files of any size in the corporate or institutes having unsecured networks," *IJCSIS*, vol. 6, pp. 155 – 163, Nov. 2009.
- [20] S. Burgett, E. Koch, and J. Zhao, "Copyright labeling of digitized image data," *Communications Magazine, IEEE*, vol. 36, pp. 94 – 100, Mar. 1998.
- [21] M. D. Swanson, B. Zhu, A. H. Tewfik, and L. Boney, "Robust audio watermarking using perceptual masking," *Signal Processing*, vol. 66, pp. 337 – 355, May 1998.
- [22] http://en.wikipedia.org/wiki/Intellectual_property/.
- [23] C. P. Pfleeger and S. L. Pfleeger, *Security-in-Computing*. prentice Hall Publisher, 2003.
- [24] http://en.wikipedia.org/wiki/Decision_tree.
- [25] N. F. Johnson, "Steganography," tech. rep., <http://www.jjtc.com/Stegdoc/>, Nov., 1995.
- [26] B. Natarajan, "Robust public key watermarking of digital images," Tech. Rep. 97 - 118, HP Laboratories <http://www.hpl.hp.com/techreports/97/HPL-97-118.pdf>, Oct. 1997.
- [27] I. Cox, M. Miller, J. Bloom, J. Fridrich, and T. Kalker, *Digital Watermarking and Steganography*. Burlington, USA: Morgan Kaufmann Publisher, 2008.
- [28] http://bytescout.com/products/enduser/watermarking/digital_watermark_types.html.
- [29] www.iprlawindia.org.
- [30] http://ec.europa.eu/justice_home/fsj/privacy/docs/intellectual_property_rights/dwvg_en.pdf.

- [31] http://www.visualwatermark.com/watermarking_faq.htm.
- [32] I. J. Cox, M. Miller, and J. Bloom, *Digital Watermarking*. Morgan Kaufmann publisher, 1st edition, 2005.
- [33] V. M. Potdar, S. Han, and E. Chang, "A survey of digital image watermarking techniques," in *3rd IEEE International Conference on Industrial Informatics (INDIN '05)*, pp. 709 – 716, Aug. 2005.
- [34] E. Muharemagic and B. Furht, "Survey of watermarking techniques and applications," tech. rep., Department of Computer Science and Engineering, FAU, USA, 2005.
- [35] A. Nikolaidis, S. Tsekeridou, A. Tefas, and V. Solachidis, "A survey on watermarking application scenarios and related attacks," in *Proceedings of 2001 International conference on Image processing*, vol. 3, pp. 991 – 994, 2001.
- [36] S. Katzenbeisser, *Information Hiding Techniques for Steganography and Digital Watermarking*. Artech House, 2000.
- [37] F. Petitcolas, "Principles of steganography. chapter 2 of information hiding techniques for steganography and digital watermarking," pp. 2 – 40, Dec. 1999.
- [38] J. R. Hernandez, F. Perez-Gonzalez, and M. Amado, "Dct-domain watermarking techniques for still images: Detector performance analysis and a new structure," *IEEE Transactions on Image Processing*, vol. 9, pp. 55 – 68, 2000.
- [39] N. F. Johnson and S. C. Katzenbeisser, *A Survey of Steganographic Techniques*. Northwood, MA: Artec House publisher, Dec. 2000.
- [40] G. C. Langelaar, I. Setyawan, and R. L. Lagendijk, "Watermarking digital image and video data. a state-of-the-art overview," *Signal Processing Magazine, IEEE*, vol. 17, pp. 20 – 46, Sep. 2000.

- [41] P. Meerwald and A. Uhl, "A survey of wavelet-domain watermarking algorithms," in *Proceedings of SPIE, Electronic Imaging, Security and Watermarking of Multimedia Contents III*, pp. 505 – 516, SPIE, Jan. 2001.
- [42] F. A. P. Petitcolas, "Watermarking schemes evaluation," *Signal Processing Magazine, IEEE*, vol. 17, pp. 58 – 64, Sep. 2000.
- [43] D. J. Fleet and D. J. Heeger, "Embedding invisible information in color images," in *Proceedings of International Conference Image Processing*, vol. 1, pp. 532 – 535, Oct. 1997.
- [44] A. H. Tewfik, "Digital watermarking," *IEEE Signal Processing Magazine*, vol. 17, pp. 17 – 18, Sep. 2000.
- [45] I. Pitas, "A method for signature casting on digital images," in *Proceedings of 1996 IEEE International conference on Image Processing*, vol. 3, pp. 215 – 218, Sep. 1996.
- [46] W. Bender, D. Gruhl, N. Morimoto, and A. Lu, "Techniques for data hiding," *IBM Systems Journal*, vol. 35, no. 3.4, pp. 313 – 336, 1996.
- [47] G. W. Braudaway, "Protecting publicly-available images with an invisible image watermark," in *Proceedings of International Conference Image Processing*, vol. 1, pp. 524 – 527, Oct. 1997.
- [48] M. M. Yeung and F. Mintzer, "An invisible watermarking technique for image verification," in *Proceedings of International Conference on Image Processing*, vol. 2, pp. 680 – 683, Oct. 1997.
- [49] H. Kinoshita, "An image digital signature system with zkip for the graph isomorphism," in *International Conference on Image Processing*, vol. 3, (Lausanne, Switzerland), pp. 247 – 250, Sept. 16 - 19 1996.

- [50] P. Kuosmanen, J. Astolat, K. Davibssont, and K. Halonenf, "Digital watermarking through filtering," in *Proceedings of the IEEE Non-linear Signal and Image Processing Workshop*, Available Online at <http://www.iwaenc.org/proceedings/1997/nsip97/pdf/scan/ns970233.pdf>, 1997.
- [51] G. C. Langelaar, J. C. A. V. der Lubbe, and R. L. Lagendijk, "Robust labeling methods for copy protection of images," in *Storage and Retrieval for Image and Video Databases (SPIE)'97*, vol. 3022, (V, San Jose, CA), pp. 298 – 309, 1997.
- [52] F. Hartung, J. K. Su, and B. Girod, "Spread spectrum watermarking: Malicious attacks and counterattacks," pp. 147–158, 1999.
- [53] M. George, J. Y. Chouinard, and N. Georganas, "Spread spectrum spatial and spectral watermarking for images and video using direct sequence techniques," in *Proc. 1999 IEEE Canadian Workshop in Information Theory, Kingston, Canada.*, pp. 119 – 122, June 15-18 1999.
- [54] R. G. Van Schyndel, A. Z. Tirkel, and C. F. Osborne, "A digital watermark," in *Proceedings of the IEEE International Conference on Image Processing*, vol. 2, pp. 86 – 90, Nov. 1994.
- [55] R. B. Wolfgang and E. J. Delp, "A watermark for digital images," in *International Conference on Image Processing*, vol. 3, pp. 219 – 222, Sep. 1996.
- [56] R. B. Wolfgang and E. J. Delp, "A watermarking technique for digital imagery: Further studies," in *International Conference on Imaging, Systems, and Technology*, vol. 1, pp. 279 – 287, IEEE, 1997.
- [57] G. B. Rhoads, "U. s. patent 5636292 : Stenography methods employing embedded calibration data," June 3, 1997.

- [58] I. J. Cox, J. Kilian, F. T. Leighton, and T. Shamoon, "Secure spread spectrum watermarking for multimedia," *IEEE Transactions on Image Processing*, vol. 6, no. 12, pp. 1673 – 1687, 1997.
- [59] A. Piva, M. Barni, F. Bartolini, and V. Cappellini, "Dct-based watermark recovering without resorting to the uncorrupted original image," in *Proceedings of International Conference on Image Processing*, vol. 1, pp. 520 – 523, Oct. 1997.
- [60] E. Koch and J. Zhao, "Towards robust and hidden image copyright labeling," in *Proceedings of the IEEE Workshop on Nonlinear Signal and Image Processing*, (Neos Marmaras, Halkidiki, Greece), pp. 452 – 455, Jun. 20 - 22 1995.
- [61] J. Zhao and E. Koch, "Embedding robust labels into images for copyright protection," in *Proceedings of the Conference on Intellectual property rights and new Technologies*, (Munich, Germany, Germany), pp. 242 – 251, R. Oldenbourg Verlag GmbH, 1995.
- [62] A. G. Borys and I. Pitas, "Image watermarking using dct domain constraints," in *Proceedings of International Conference on Image Processing*, vol. 3, pp. 231 – 234, Sep. 1996.
- [63] M. J. Holliman, N. D. Memon, B.-L. Yeo, and M. M. Yeung, "Adaptive public watermarking of dct-based compressed images," in *Proceedings of Storage and Retrieval for Image and Video Databases (SPIE)'98*, vol. 3312, pp. 284 – 295, Jan. 28 - 30 1998.
- [64] B. Tao and B. Dickinson, "Adaptive watermarking in the dct domain," in *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing*, vol. 4 of *ICASSP '97*, (Washington, DC, USA), pp. 2985 – 2988, IEEE Computer Society, 1997.

- [65] Hsu, C. T., Wu, and J. L., “Hidden digital watermarks in images,” in *IEEE Transactions on Image Processing*, vol. 8, pp. 56 – 68, 1999.
- [66] C.-T. Hsu and J.-L. Wu, “Hidden signatures in images,” in *Proceedings of International Conference Image Processing*, vol. 3, pp. 223 – 226, Sep. 1996.
- [67] K. K. Wong, C. H. Tse, K. S. Ng, T. H. Lee, and L. M. Cheng, “Adaptive watermarking,” in *IEEE Transactions on Consumer Electronics*, vol. 43, pp. 1003 – 1009, Nov. 1997.
- [68] C. I. Podilchuk and W. Zeng, “Digital image watermarking using visual models,” in *Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series* (B. E. R. . T. N. Pappas, ed.), vol. 3016, pp. 100 – 111, Jun. 1997.
- [69] M. Barni, F. Bartolini, V. Cappellini, and A. Piva, “A dct-domain system for robust image watermarking,” *Signal Processing*, vol. 66, pp. 357 – 372, May 1998.
- [70] M. D. Swanson, B. Zhu, and A. H. Tewfik, “Transparent robust image watermarking,” in *Proceedings of 1996 International Conference on Image Processing*, vol. 3, pp. 211 – 214, Sep. 1996.
- [71] X.-G. Xia, C. G. Boncellet, and G. R. Arce, “A multiresolution watermark for digital images,” in *Proceedings of the 1997 International Conference on Image Processing (ICIP '97) 3 - Volume Set*, ICIP '97, (Washington, DC, USA), pp. 548–551, IEEE Computer Society, 1997.
- [72] C. I. Podilchuk and W. Zeng, “Image-adaptive watermarking using visual models,” *IEEE Journal on Selected Areas in Communications*, vol. 16, pp. 525 – 539, May 1998.
- [73] J. F. Delaigle, C. D. Vleeschouwer, and B. Macq, “Digital watermarking,” in *Optical Security and Counterfeit Deterrence Techniques, SPIE Electronic Imag-*

- ing : Science and Technology*, vol. in Conference 2659, (San Jose, CA), pp. 99 – 110, Feb. 1996.
- [74] D. Kundur and D. Hatzinakos, “A robust digital image watermarking method using wavelet-based fusion,” in *Proceedings of the IEEE International Conference on Image Processing*, vol. 1, (Santa Barbara, CA), pp. 544 – 547, Oct. 26-29 1997.
- [75] C. I. Podilchuk and W. Zeng, “Watermarking of the jpeg bitstream,” in *Proceedings of the International Conference on Imaging Science, Systems and Technology*, (Las Vegas, Nevada), pp. 253 – 260, June 30 - July 3 1997.
- [76] W. Zeng and B. Liu, “On resolving rightful ownership’s of digital images by invisible watermarks,” in *Proceedings of the 1997 International Conference on Image Processing*, vol. 3 of *ICIP ’97*, 1997.
- [77] J. J. K. O. Ruanaidh, W. J. Dowling, and F. M. Bol, “Phase watermarking of digital images,” in *International Conference on Image Processing*, vol. 3, (Lausanne, Switzerland), pp. 239 – 242, Sept. 16-19 1996.
- [78] J. J. K. O’Ruanaidh and T. Pun, “Rotation, scale and translation invariant digital image watermarking,” in *Proceedings of the IEEE International Conference on Image Processing*, vol. 1, pp. 536 – 539, Oct 1997.
- [79] M. L. Miller, I. J. Cox, J. paul, and M. G. Linnartz, “A review of watermarking principles and practices,” in *K. Parhi and T. Nishitani, eds, ‘Digital Signal Processing in Multimedia Systems’, Marcell Dekker Inc*, pp. 461–485, 1999.
- [80] S. Czerwinski, R. Fromm, and T. Hodes, “Digital music distribution and audio watermarking,” 1999.
- [81] M. Kutter and F. A. P. Petitcolas, “A fair benchmark for image watermarking systems,” *Proceedings of Electronic Imaging 99, Security and Watermarking of*

- Multimedia Contents, The Society for Imaging Science and Technology and the International Society for Optical Engineering*, vol. 3657, pp. 226 – 239, Jan., 25 - 27 1999.
- [82] J. Zhao, E. Koch, and C. Luo, “In business today and tomorrow,” *Communication ACM*, 41(7), pp. 66 – 72, July 1998.
- [83] M. Arnold, “Audio watermarking: features, applications and algorithms,” in *Proceedings of the IEEE International Conference on Multimedia and Expo. (ICME 2000)*, vol. 2, pp. 1013 – 1016, July 30- August 2 2000.
- [84] L. Boney, A. H. Tewfik, and K. N. Hamdy, “Digital watermarks for audio signals,” in *Proceedings of the Third IEEE International Conference on Multimedia Computing and Systems*, pp. 473 – 480, June 1996.
- [85] S. L. E. Rafael C. Gonzalez, Richard E. Woods, *Digital Image Processing using MATLAB*. Pearson Education publisher, 2005.
- [86] M. H. Dunham and S. Sridhar, *Data Mining, Introductory and Advanced Topics*. Pearson Education, 2008.
- [87] “Weka.” <http://perun.pmf.uns.ac.rs/radovanovic/dmsem/cd/install/Weka/doc/html/Weka%203.4.5.htm>.
- [88] A. Noore, “An improved digital watermarking technique for protecting jpeg images,” in *Proceedings of the IEEE International Conference on Consumer Electronics*, (West Virginia University, Morgantown, WV, USA), pp. 222 – 223, Jun. 2003.
- [89] V. Fotopoulos and A. N. Skodras, “A subband dct approach to image watermarking,” in *Proceedings of Tenth European Signal Processing Conference*, (Tampere, Finland), Sep. 4 - 8 2000.

- [90] Y. Choi and K. Aizawa, "Digital watermarking using inter-block correlation: Extension to jpeg coded domain," in *Proceedings of the The International Conference on Information Technology: Coding and Computing (ITCC'00)*, ITCC '2000, (Washington, DC, USA), IEEE Computer Society, 2000.
- [91] M. A. Suhail and M. S. Obaidat, "Digital watermarking-based dct and jpeg model," *IEEE Transactions on Instrumentation and Measurement*, vol. 52, pp. 1640 – 1647, Oct. 2003.
- [92] A. Golikeri and P. Nasiopoulos, "A robust dct energy based watermarking scheme for images," (The University of British Columbia, Vancouver, BC, Canada), Jun. 1 2005.
- [93] J. Huang, Y. Q. Shi, and Y. Shi, "Embedding image watermarks in dc components," *IEEE Transactions on Circuits Systems Video Technology*, vol. 10, no. 6, pp. 974 – 979, 2000.
- [94] P. H. W. Wong, O. C. Au, and J. W. C. Wong, "Data hiding and watermarking in jpeg compressed domain by dc coefficient modification," in *Security and Watermarking of Multimedia Contents II, Proceedings of SPIE*, vol. 3971, June 2001.
- [95] E. Ganic, S. D. Dexter, and A. M. Eskicioglu, "Embedding multiple watermarks in the dft domain using low- and high-frequency bands," in *Security, Steganography, and Watermarking of Multimedia Contents' 05*, 2005.
- [96] P. Tao and A. M. Eskicioglu, "A robust multiple watermarking scheme in the discrete wavelet transform domain," in *Proceedings of The Society of Photo-Optical Instrumentation Engineers (SPIE)- Conference Series* (. S. P. J. R. Smith, T. Zhang, ed.), vol. 5601, pp. 133 – 144, Oct. 2004.

- [97] D. Kundur and D. Hatzinakos, "Toward robust logo watermarking using multiresolution image fusion principles," *IEEE Transactions on Multimedia*, vol. 6, pp. 185 – 198, Feb. 2004.
- [98] M. S. Raval and P. P. Rege, "Discrete wavelet transform based multiple watermarking scheme," in *Conference on Convergent Technologies for Asia-Pacific Region (TENCON '03.)*, vol. 3, pp. 935 – 938, Oct. 2003.
- [99] X. Kang, J. Huang, Y. Q. Shi, and Y. Lin, "A dwt-dft composite watermarking scheme robust to both affine transform and jpeg compression," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 13, pp. 776 – 786, Aug. 2003.
- [100] C. tang Hsieh and Y. kuang Wu, "Digital image multiresolution watermark based on human visual system using error correcting code," 2001.
- [101] X. Niu, S. Sun, and W. Xiang, "Multiresolution watermarking for video based on gray-level digital watermark," *IEEE Transactions on Consumer Electronics*, vol. 46, pp. 375 – 384, May 2000.
- [102] C. T. Hsu and J. L. Wu, "A multiresolution watermark for digital images," in *IEEE Transactions on Circuits Systems*, pp. 1097 – 1101, Aug. 1998.
- [103] J. J. Chae and B. S. Manjunath, "A robust embedded data from wavelet coefficients," in *SPIE: Storage and Retrieval for Image and Video Databases VI*, vol. 3312, pp. 308 – 317, Jan. 1998.
- [104] X. G. Xia, C. Boncelet, and G. Arce, "Wavelet transform based watermarking for digital images," in *Optics Express*, vol. 3, pp. 497 – 511, 1998.
- [105] W. Xiao, Z. Ji, X. Zhang, and W. Wu, "A watermarking algorithm based on chaotic encryption," in *Proceedings of IEEE Region 10 Conference on Computers, Communications, Control and Power Engineering (TENCON '02)*, vol. 1, pp. 545 – 548, Oct. 2002.

- [106] N. Kaewkamnerd and K. R. Rao, "Multiresolution based image adaptive watermarking scheme," in *EUSIPCO*, (Tampere, Finland), 2000.
- [107] N. Kaewamnerd and K. R. Rao, "Wavelet based image adaptive watermarking scheme," *Electronics Letters*, vol. 36, pp. 312 – 313, Feb. 2000.
- [108] C.-S. Lu, Liao, H.-Y. Mark, Huang, Shih-Kun, and C.-J. Sze, "Cocktail watermarking on images," in *Proceedings of the Third International Workshop on Information Hiding*, IH '99, (London, UK), pp. 333 – 347, Springer-Verlag, 2000.
- [109] W. Zhu, Z. Xiong, and Y. Zhang, "Multiresolution watermarking for images and video: A unified approach," vol. 9, pp. 545 – 550, June 1999.
- [110] D. Kundur and D. Hatzinakos, "Digital watermarking using multiresolution wavelet decomposition," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, vol. 5, pp. 2969 – 2972, May 1998.
- [111] G. Zeng and Z. Qiu, "Image watermarking based on dc component in dct," in *Proceedings of the 2008 International Symposium on Intelligent Information Technology Application Workshops*, (Washington, DC, USA), pp. 573–576, IEEE Computer Society, 2008.
- [112] J. Huang and Y. Q. Shi, "Embedding strategy for image watermarking in dct domain," in *Fifth Asia-Pacific Conference on Communications and Fourth Optoelectronics and Communications Conference (APCC/OECC '99)*, vol. 2, pp. 981 – 984, 1999.
- [113] M. Eyadat, "Factors that affect the performance of the dct-block based image watermarking algorithms," in *Proceedings of International Conference on Information Technology: Coding and Computing (ITCC '04.)*, vol. 1, pp. 650 – 654, Apr. 2004.

- [114] K. Sayood, *Introduction to Data Compression*. Morgan Kaufmann Publishers, 2006.
- [115] “Video processing.” http://disp.ee.ntu.edu.tw/Digital_Video_Compression_Fundamentals_and_Standards/.
- [116] T. Wiegand, G. Sullivan, G. Bjontegaard, and A. Luthra, “Overview of the h.264/avc video coding standard,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 13, pp. 560 – 576, Jul. 2003.
- [117] C. Song, S. Sudirman, M. Merabti, and D. Llewellyn-Jones, “Analysis of digital image watermark attacks,” in *Proceedings of the 7th IEEE Consumer Communications and Networking Conference (CCNC 2010)*, pp. 1 – 5, Jan. 2010.
- [118] R. Ridzon and D. Levicky, “Robust digital watermarking in color images,” in *15th International Conference on Systems, Signals and Image Processing (IWSSIP 2008.)*, pp. 425 – 428, June 2008.
- [119] I. J. Cox, J. Kilian, T. Leighton, and T. Shamoon, “A secure, imperceptible yet perceptually salient, spread spectrum watermark for multimedia,” in *Southcon/96. Conference Record*, pp. 192 – 197, June 1996.
- [120] L. Cai and S. Du, “Robust digital image watermarking method against rst attacks,” in *Proceedings of the International Conference on Signal Processing and Communications (SPCOM '04.)*, pp. 491 – 495, Dec. 2004.
- [121] C. S. Lu, C. Y. Hsu, S. W. Sun, and P. C. Chang, “Robust mesh-based hashing for copy detection and tracing of images,” in *Proceedings of the IEEE International Conference on Multimedia and Expo. (ICME '04.)*, vol. 1, pp. 731 – 734, June 2004.
- [122] S. K. Yip, O. C. Au, C. W. Ho, and H. M. Wong, “Content adaptive watermarking using a 2-stage predictor,” in *Proceedings of the IEEE International Conference on Image Processing (ICIP 2005.)*, vol. 1, pp. 953 – 956, 2005.

- [123] S. P. Mohanty, R. Sheth, A. Pinto, and M. Chandy, "Cryptmark: A novel secure invisible watermarking technique for color images," in *Proceedings of the IEEE International Symposium on Consumer Electronics (ISCE '07.)*, pp. 1 – 6, June 2007.
- [124] M. Antonini and M. Barlaud, "Image coding using wavelet transform," in *IEEE Transactions on Image Processing*, vol. 1, pp. 205 – 220, Apr. 1992.
- [125] D. Benham, N. Memon, B. L. Yeo, and M. Yeung, "Fast watermarking of dct-based compressed images," in *Proceedings of the International Conference on Imaging Science, Systems and Technology*, vol. 5, (Las Vegas, Nevada), pp. 243–252, June 30 - July 3 1997.
- [126] J. J. K. A. Ruanaidh, W. J. Dowling, and F. M. Boland, "Watermarking digital images for copyright protection," in *Proceedings of the International Conference on Image Processing and its Applications*, pp. 250 – 256, 1996.
- [127] S. Katzenbeisser and F. A. P. Petitcolas, *Information hiding techniques for steganography and digital watermarking*. Eds. Northwood, MA: Artec House, 1999.
- [128] M. Kutter, F. Jordan, and F. Bossen, "Digital signature of color images using amplitude modulation," in *Journal of Electronic Imaging*, pp. 326–332, 1997.
- [129] H. L. V. Trees, *Detection, Estimation, and Modulation Theory*. John Wiley & Sons, 1968.
- [130] J. Dugelay and S. Roche, *A Survey of Current Watermarking Techniques*. 1999.
- [131] J. Han and M. Kamber, *Data mining concepts and techniques*. Morgan Kaufmann publishers, 2005.
- [132] Kipper and Gregory, *Investigator's Guide to Steganography*. Boca Raton, FL, USA: CRC Press, Inc., 2003.

- [133] S. B. Patel, S. N. Pradhan, and S. U. Ambegaokar, “A novel approach for implementing steganography with computing power obtained by combining cuda and matlab,” *IJCSIS*, vol. abs/0912.0947, 2009.
- [134] S. B. Patel, “Proposed secure mechanism for identification of ownership of undressed photographs or movies captured using camera based mobile phones,” in *Proceedings of 2nd International Conference on Digital Information Management*, vol. 1, pp. 442 – 447, Oct. 2007.
- [135] Rajmohan, “Watermarking of digital images,” me thesis report, Dept. Electrical Engineering, Indian Institute of Science, Bangalore, India, 1998.
- [136] M. Wu, *Multimedia Data Hiding*. Springer Press, 2002.
- [137] <http://www.jjtc.com/Steganography/tools.html>.
- [138] <http://hypatia.math.uri.edu/~kulenm/diffeqaturi/victor442/index.html>.
- [139] www.nvidia.in.
- [140] “Ipr.” www.iprindia.com.
- [141] “Ipr.” <http://www.nipo.in/>.

Index

- 2D DCT, 148
- , 48
- Acknowledgement, xi
- Applications, 14
- Arnold transformation, 75
- Block based DCT algorithms, 43
- Candidate's Statement, v
- Certificate, iii
- Classification of Watermarking Techniques,
27
- Comparison and Contribution, 80
- Compression, 13
- Conclusions, 136
- Contrast stretching Results, 110
- Convolution, 150
- Copyright, 19
- Cryptography, 10
- Data Payload, 32
- DCT, 148
- Decision tree, 65
- Digital Rights Management, 21
- Discrete Cosine Transform, 148
- Discrete Wavelet Transform, 149
- DWT based Embedding Technique, 51
- Encryption, 21
- Evidence Report, 133
- Experimental Results, 85
- Fundamental Properties, 30
- Future work, 139
- General Block diagram of Steganography
and Watermarking, 5
- General Results and Discussion, 87
- Generalization of Coefficients into classes,
60
- Geometric Attack, 104
- Geometric Random Local Distortion At-
tack, 105
- Geometric Rotational Attack, 104
- Geometric Scaling Attack, 104
- Geometric Shearing Attack, 105
- ID3, 38
- Imperceptibility, 30
- Intellectual Property Rights, 18
- Introduction to Data mining, 36

- Key usage, 129
- LSB Technique, 6
- Mathematical model, 58
- MSE, 152
- Objective, 3
- Partial decision tree, 66
- Patents, 20
- Perceptibility, 30
- PSNR, 152
- Publication details, 140
- Quantization Table used in JPEG, 46
- Result after Adding Constant to an Image Plane, 110
- Results for Averaging Filter Attack, 106
- Results for Disk Circular Averaging Filter Attack, 106
- Results for Dithering of Pixels Attack, 109
- Results for Gaussian Filter Attack, 107
- Results for JPEG Lossy Compression Attack, 106
- Results for Laplacian Filter Attack, 108
- Results for Laplacian of Gaussian Filter Attack, 109
- Results for Motion Blur Filter Attack, 107
- Results for Prewitt and Sobel Filter Attack, 108
- Results for Sharpen Image Filter Attack, 108
- Results from Datamining approach, 85
- Results of Minimum Variance Quantization Attack, 109
- Results of Uniform Quantization Attack, 109
- Robustness, 31
- ROI, 64
- Signal Processing Attacks, 105
- Specialized Attack based on Knowledge of method, 110
- StegAlyzerAS, 133
- Steganalysis, 11
- Steganography, 5
- Test for Robustness against attacks, 98
- Test Images, 153
- Trade Secrets, 20
- Training before embedding, 58
- Transform domain techniques, 28
- Watermarking, 22
- Watermarking using Decision tree and DCT, 57