Orthogonal transform watermarking techniques for authentication of digital images and documents

Shu, Feng

2005

Shu, F. (2005). Orthogonal transform watermarking techniques for authentication of digital images and documents. Master's thesis, Nanyang Technological University, Singapore.

<https://hdl.handle.net/10356/3253>

<https://doi.org/10.32657/10356/3253>

Nanyang Technological University

Downloaded on 26 Apr 2023 13:40:39 SGT

ORTHOGONAL TRANSFORM WATERMARKING TECHNIQUES FOR AUTHENTICATION OF DIGITAL IMAGES AND DOCUMENTS

SHU FENG

SCHOOL OF ELECTRICAL & ELECTRONIC ENGINEERING
NANYANG TECHNOLOGICAL UNIVERSITY

2005

**Orthogonal Transform Watermarking
Techniques for Authentication of Digital
Image and Documents**

SHU FENG

SCHOOL OF ELECTRICAL & ELECTRONIC ENGINEERING

A thesis submitted to the Nanyang Technological University

in fulfillment of the requirement of the degree of

Master of Engineering

2005

Statement of Originality

I hereby certify that the content of this thesis is the result of work done by me and has not been submitted for a higher degree to any other University or Institution.

Author's signature:

Date:

Acknowledgements

I would like to acknowledge the help and support of everyone during my M.Eng study. Special thanks go to my project supervisor, Associate Professor Ho Tung Shuen, Anthony, who had given me a lot of guidance and encouragement in the past two years. I would like to thank Assistant Professors Guan Yong Liang and Pina Marziliano, and Associate Professor Anamitra Makur, who had given me invaluable supports and suggestions on my research.

I also would like to thank Mr. Shen Jun, Mr. Gao Yupeng and other colleagues in our research group, with whom I always had helpful discussions and shared ideas.

I also take pleasure in thanking the technicians in Workstation Resource Lab Mr. Chu Chun Chung and Ms. Teo Boon Ping, who provided me a lot of technical supports whenever I needed.

Contents

1	Introduction	1
1.1	Motivation	1
1.2	Objectives	3
1.3	Contributions	3
1.4	Author's publications	4
1.5	Organization of this thesis	5
2	Watermarking Basics	7
2.1	Introduction to information hiding	7
2.2	A general framework of watermarking	8
2.3	Watermarking requirements	10
2.3.1	Imperceptibility	11
2.3.2	Robustness	11

2.3.3	Watermark security	12
2.3.4	Capacity	12
2.3.5	Watermark recovery with or without the cover data	13
2.3.6	Complexity	13
2.4	Applications of watermarking	13
2.4.1	Copyright protection	14
2.4.2	Content authentication	14
2.4.3	Fingerprinting	15
2.4.4	Copy control	15
3	Algorithm Review	17
3.1	Classification of watermarking techniques	17
3.1.1	Spatial domain and transform domain watermarking	18
3.1.2	Robust, semi-fragile and fragile watermarking	18
3.1.3	Private, semi-private and public watermarking	19
3.1.4	One-bit and multi-bit watermarking	19
3.2	Spatial domain watermarking	20
3.3	Transform domain watermarking	25

3.3.1	Discrete Cosine transform	26
3.3.2	Discrete Fourier transform	29
3.3.3	Discrete Wavelet transform	32
3.3.4	Hadamard transform	37
3.4	Attacks	39
3.4.1	Removal attacks	39
3.4.2	Presentation attacks	40
3.4.3	Interpretation (protocol) attacks	40
3.4.4	Legal attacks	40
3.5	Benchmarking software	41
3.5.1	Stirmark	41
3.5.2	Checkmark	42
4	Proposed Methodologies	43
4.1	Background	45
4.1.1	Two-level quantization	45
4.1.2	Distortion bound	46

4.2	Proposed semi-private watermarking	
	algorithm	48
4.2.1	Watermark embedding	52
4.2.2	Watermark extraction	57
4.3	Proposed public watermarking algorithm	58
4.3.1	A spread-spectrum perspective	58
4.3.2	Proposed spread-spectrum based watermarking approach . . .	61
5	Performance Evaluation	67
5.1	Experimental results	67
5.1.1	Non-geometric attacks	70
5.1.2	Geometrical attacks	77
5.1.3	Combined attacks	80
5.2	Performance comparison of spread-spectrum watermarking algorithms	83
6	Applications	89
6.1	Introduction to a card authentication system	89
6.2	Print-and-scan process	93
6.3	Experimental results and discussions	95

7 Conclusion and Recommendations	97
7.1 Conclusion	97
7.2 Recommendations	99
Bibliography	101

Summary

We propose two robust watermarking schemes: the semi-private watermarking and the public watermarking schemes. In our transform-domain-based watermarking algorithms, two-level quantization is employed for both watermark embedding and extraction processes. Our public watermarking method is essentially extended from the semi-private approach and a spread-spectrum (SS) technique, which has been widely used in digital communication, is employed.

Instead of modifying the magnitude of the frequency components, in our methods, the watermark bits are inserted by establishing a relationship with the polarities of the selected transform coefficients. The advantage of embedding the watermark into polarities is the considerable improvement on robustness under a variety of accidental and intentional manipulations. This is because that the polarities of the frequency components are found to be more reliable than magnitude, according to our investigation.

Both Stirmark and Checkmark are used for benchmarking, and the experimental results under different categories of attacks (e.g., non-geometrical, geometrical and

combined attacks) are reported. It is found that both the semi-private and public watermarking methods are able to survive up to 90% of the non-geometric attacks from Checkmark. They are also robust against a number of geometrical attacks, (e.g., line removal, scaling and affine transformations), as well as some combined attacks, such as JPEG compression combined with additive noise, and print-and-scan process.

A card authentication system is developed by using our semi-private watermarking algorithm in the Hadamard transform domain. Our system has been proved to be resistant to the print-and-scan (PS) process and authenticated with a success rate close to 96% of 105 photo-IDs. The proposed system could be applied to prevent photo-substitution frauds from many security documents and cards, such as student cards, driving licenses and passports.

List of Figures

2.1	Generic digital watermarking embedding process	9
2.2	Generic digital watermarking decoding process	10
3.1	The circularly symmetric watermark	31
3.2	Three-level wavelet decomposition of LENA	33
3.3	Two-channel wavelet decomposition and reconstruction structures. . .	34
4.1	Test images in our experiments. All images are grayscale, of size 512x512 pixels.	49
4.2	DCT coefficient difference of LENNA after JPEG 30 compression. . .	50
4.3	DCT coefficient difference of BABOON after JPEG 30 compression. .	50
4.4	DCT coefficient difference of PEPPERS after JPEG 30 compression. .	51
4.5	A framework of the semi-private watermarking technique	51
4.6	Amplitude modulation for transform coefficients	53

4.7	Watermark embedding of the semi-private algorithm	54
4.8	The watermark and the selected coefficients in the example.	55
4.9	The modulated coefficients and the generated linkage sequence in the example.	56
4.10	Watermark extraction process of the semi-private algorithm	57
4.11	A general traditional spread-spectrum (TSS) watermarking frame- work	59
4.12	A framework of the TQSS watermarking technique	62
4.13	An example of minimum distance detection.	65
5.1	Watermarked LENA image by the semi-private and public water- marking algorithms.	69
5.2	Sample resultant images by non-geometrical attacks.	71
5.3	More samples of resultant images by non-geometrical attacks.	72
5.4	Sample resultant images by geometrical attacks.	78
5.5	Sample resultant images by combined attacks.	81
5.6	Performance comparison under JPEG compression	84
5.7	Performance comparison under wavelet compression	85
5.8	Performance comparison under AWGN	86

5.9	Performance comparison under scaling down	87
6.1	An example of photo-substitution on student cards.	91
6.2	Watermark embedding process	93
6.3	Watermark retrieval and verification process	93
6.4	The Print-and-scan process	94

List of Tables

3.1	An example of cipher key table.	21
5.1	Results with Stirmark. Common signal processing attacks.	74
5.2	Results with Checkmark 1.2. Non-geometric attacks.	75
5.3	More results with Checkmark 1.2. Non-geometric attacks. Note: (1)DPR: Denoising and remodulation with Wiener filtering; (2)DPRCORR: Denoising and remodulation assuming a correlated watermark.	76
5.4	Results with Stirmark. Geometric attacks.	79
5.5	Results of combined attacks.	82
5.6	Bit error rates under more attacks.	88
6.1	Sample experimental results of the card authentication system. Note: (1)The image's unit is pixels; (2) The IDs used here are not the real identification sequences.	96

Chapter 1

Introduction

1.1 Motivation

The explosive development of digital multimedia products, especially with the popularization of the Internet, has brought us into a digital era. Because it is faster and more economical, people are prone to transmit multimedia files through computer networks rather than use hard copies. As a result, a huge amount of images, video sequences and audio files are stored in databases and transmitted through the Internet in digital formats. On the other hand, with some well-developed editing tools, for example, Photoshop 7, electronic media can be easily copied and redistributed in their exact original forms.

The ease of copying and editing raises the problems of illegally using, pirating and counterfeiting digital images, video, audio and electronic documents. Public key

CHAPTER 1. INTRODUCTION

encryption systems such as RSA algorithm do not completely solve the problem of unauthorized copying because of the ease with which images may be reproduced from previously published documents [1]. All encrypted documents and images need to be decrypted before they can be inspected or used. Once encryption is removed, the document can be passed on in an electronic form. If there is more than one recipient of an image, there is no direct proof that any particular authorized recipient is responsible for passing it on to unauthorized users.

The idea of using an indelible watermark to identify uniquely both the source of an image and an intended recipient has therefore stimulated much interest in the electronic publishing and printing industries. In 1993, Tirkel et al.[2] first introduced the ‘electronic watermark’ by proposing to hide some identification information into the LSB’s (least significant bits) of the cover image. This is considered as the beginning of digital watermarking. Thereafter, digital watermarking has drawn a lot of research attentions and a significant number of watermarking algorithms have been presented in the literature. Although some important issues remain unsolved so far [3], it is clear that digital watermarking has emerged as a promising candidate to solve the problem of protecting digital multimedia products.

In the literature, the main problem of most previously proposed algorithms is that those methods are not robust enough against different kinds of attacks or specifically designed to be only robust against some particular attacks, such as RST (rotation, scaling and translation) attacks [4]. Furthermore, most of the algorithms were only benchmarked with single attacks. However, simple combined attacks, which are sup-

CHAPTER 1. INTRODUCTION

posed to be quite common in reality [5], such as JPEG compression plus sharpening or JPEG compression plus resizing, should be overcome by a practical watermarking system.

In this thesis, we propose two robust watermarking algorithms which were thoroughly tested with two well-known benchmarking tools, Stirmark 4.0 and Checkmark 1.2. Our algorithms are also proved to be immune to some combined attacks, including the print-and-scan process.

1.2 Objectives

The main objective of this thesis is to research, develop and analyze novel robust watermarks for owner verification or copyright protection.

1.3 Contributions

The main contributions in this thesis are as follows:

1. We devised two novel robust watermarking algorithms. One is a semi-private watermarking method and the other is a public watermarking algorithm which employs the spread-spectrum technique. Both of them can survive a significant number of attacks.

CHAPTER 1. INTRODUCTION

2. We designed a practical watermarking system for card authentication, which can successfully survive the print-and-scan process. The card authentication system could be applied to prevent photo-substitution in some security documents such as student cards, driving licenses and passports.

1.4 Author's publications

1. F. Shu, A. T. S. Ho and J. Shen, "Digital Watermarking Techniques Using Polarity of Transform Coefficients," Accepted for *IASTED International Conference on Signal and Image Processing (SIP 2003)*, Honolulu, USA.
2. A. T. S. Ho and F. Shu, "Print-and-scan Resilient Digital Watermark for Card Authentication," in Proc. of *Fourth International Conference on Information, Communication & Signal Processing / Fourth IEEE Pacific-Rim Conference on Multimedia*, Singapore, Dec. 2003.
3. A.T.S.Ho, F. Shu, "A Robust Spread-Spectrum Watermarking Method Using Two-Level Quantization," Accepted for *IEEE Int. Conf. on Image Processing 2004*, Oct. Singapore 2004.
4. A.T.S.Ho, F.Shu, "A Two-Level Quantization Based Spread-Spectrum Technique for Robust Watermarking," Submitted to *IEEE Transaction on Circuit and Systems II*

1.5 Organization of this thesis

This thesis is organized as follows.

Chapter 1 Introduction

We introduce the motivation, objectives, main contributions, publications and organization of this thesis.

Chapter 2 Watermarking Basics

Some basic principles, such as general frameworks of watermark embedding and decoding processes, and watermarking requirements are discussed in Chapter 2. Applications of digital watermarking are also introduced.

Chapter 3 Algorithm Review

In Chapter 3, we categorize the existing watermarking algorithms and briefly review the proposed approaches in the literature. Attacks and benchmarking tools are also discussed.

Chapter 4 Proposed Methodologies

We present our proposed two watermarking methods in Chapter 4. The first algorithm is a semi-private watermarking method and the second one is a public watermarking approach.

Chapter 5 Performance Evaluation

Performance evaluation of the two proposed algorithms are described in Chapter

CHAPTER 1. INTRODUCTION

5. The robustness testing is based on different categories of attacks, including non-geometrical attacks, geometrical attacks and combined attacks. The performance comparison between our public watermarking method and other common spread-spectrum based watermarking algorithms is presented.

Chapter 6 Applications

A card authentication system which uses our proposed algorithm is described in Chapter 6 and experimental results are also presented and analyzed.

Chapter 7 Conclusion and Recommendations

Finally, the conclusion is drawn and some research recommendations are suggested in Chapter 7.

Chapter 2

Watermarking Basics

In this chapter, we introduce some basic knowledge about digital watermarking, including a generic framework of watermark embedding and decoding, watermarking requirements and its applications.

2.1 Introduction to information hiding

There has seen a growing interest in the fields of information hiding in the recent years, which mainly includes steganography and digital watermarking. In a general model of hiding information, the embedded data are the message that one wishes to send secretly. It is usually hidden in an innocuous message referred to as a cover text, cover image, or cover audio, producing the stego-text or other stego-object. A stego-key is used to control the hiding process so as to restrict detection and recovery of the embedded information to parties who know it.

CHAPTER 2. WATERMARKING BASICS

Steganography, which literally means “covered writing” in Greek, represents for techniques that allow secret communication, usually by hiding the secret information in another unsuspected data. Steganographic methods generally do rely on the assumption that the existence of the covert communication is unknown to third parties and are mainly used in secret point-to-point communication between trusting parties. Cryptography is about protecting the content of messages, whereas steganography is concealing their very existence. As a result, steganographic methods are in general not robust, i.e., the hidden information cannot be recovered after attacks.

Another main subdiscipline of information hiding is digital watermarking, which has additional requirement of robustness against possible attacks, as opposed to steganography. Even if the existence of the hidden information is known, it is difficult — ideally impossible — for an attacker to destroy the embedded watermark.

2.2 A general framework of watermarking

Watermarking is a process of adding some identification information into the cover data, such as images, audio, video and documents. The embedded information, known as watermarks, can be extracted from the watermarked multimedia contents to provide proofs for various purposes, such as copyright protection or content authentication.

All the digital watermarking algorithms consist of two parts: watermark embedding

CHAPTER 2. WATERMARKING BASICS

Figure 2.1: Generic digital watermarking embedding process

and watermark retrieval. The generic watermark embedding process is depicted in Figure 2.1. Mathematically, the watermark embedding process can be expressed by Equation 2.1.

$$I' = I + f(W, K) \quad (2.1)$$

where I' and I , are the watermarked and cover signals, respectively. K denotes a secret key, which records some useful information for watermark detection, for example, the embedding positions, the length of the watermark signal and the size of the cover signal. If the watermark embedding algorithm is dependent on the cover data, then the embedding equation can be expressed by Equation 2.2.

$$I' = I + f(I, W, K) \quad (2.2)$$

The watermark extraction process can be expressed as Equation 2.3.

$$W^* = g(I, I^*, K) \quad (2.3)$$

CHAPTER 2. WATERMARKING BASICS

Figure 2.2: Generic digital watermarking decoding process

where I and I^* denote the cover data and the test data, respectively. W^* is the retrieved watermark from the test signal I^* . If the extraction process does not depend on the cover signal,

$$W^* = g(I^*, K) \quad (2.4)$$

The generic process of watermark retrieval is illustrated in Figure 2.2.

2.3 Watermarking requirements

Different requirements should be taken into consideration when a digital watermark is designed for a certain application. In this section, we shall review some of these requirements and indicate when they are important.

CHAPTER 2. WATERMARKING BASICS

2.3.1 Imperceptibility

One of the main requirements for watermarking is the perceptual transparency. The watermark embedding process should not introduce any perceptible artifacts into the cover data. Distortions introduced through a watermarking process are not only annoying and undesirable, but may also reduce or destroy the commercial value of the watermarked data. On the other hand, for high robustness, it is desirable that the watermark strength is as high as possible. Thus the design of a watermarking method always involves a trade-off between imperceptibility and robustness [6]. It is therefore important to design a watermark which exploits effects of the human visual or auditory system in order to maximize the watermark energy under the constraint of the perceptible threshold.

2.3.2 Robustness

Robustness of the watermarked data against modifications and malicious attacks is one of the key requirements in watermarking. The robustness requirement varies depending on the purpose of the designed watermarking application [7]. A robust watermark is usually used to verify the ownership of the protected work, therefore an ideal robust watermark should resist any kind of distortions. A fragile watermark is designed for content authentication or tamper proofing, and thus it must be sensitive to any accidental or intentional manipulations. A semi-fragile watermark is also sensitive to most attacks but can survive some pre-defined attacks

CHAPTER 2. WATERMARKING BASICS

(e.g., lossy compression). There are two main benchmarking tools, Stirmark and Checkmark, available on the Internet which could be used to test the robustness of a watermarking method.

2.3.3 Watermark security

In most applications, such as copyright protection, the secrecy of embedded information needs to be assured. If secrecy is a requirement, a secret key has to be used for the embedding and extraction processes. The security of watermarking is interpreted in the same way as the security of encryption techniques. A secure watermarking procedure cannot be broken unless the unauthorized user has access to a secret key that controls the insertion of the data in the cover signal [8].

2.3.4 Capacity

Embedding capacity measures the amount of information that can be hidden relative to the size of the cover data [9]. Some applications of watermarking, e.g., insertion of a serial number, require that relatively small amounts of information be embedded in the cover signal. On the other hand, many other applications, e.g., embedding a smaller image into a larger image or embedding multiple speech signals into a video, require a significant amount of bandwidth. In these cases, the algorithms must be able to embed a large amount of data, while in the meantime maintaining the robustness level and the integrity of the cover signal.

CHAPTER 2. WATERMARKING BASICS

2.3.5 Watermark recovery with or without the cover data

Watermark recovery is usually more robust if the cover signal is available. This is because that the cover signal allows the detection and inversion of distortions which change the data geometry [4]. However, for some applications, like video watermarking, it may be impractical to use the cover data because a large data volume must be stored in the database. In other applications, such as data monitoring or tracking, access to the cover data is not possible. It is, however, possible to design watermarking techniques that do not require the cover data for watermark extraction. Therefore, it is more desirable that a watermarking method is able to extract the watermark without the cover signals. Such methods are referred to as “public” or “blind” watermarking algorithms, while the methods which require the cover data for extraction are called “private” or “non-blind” watermarking approaches [7].

2.3.6 Complexity

Computational complexity of the watermark encoding and decoding processes is another consideration. In some real-time applications, e.g., copy control of a DVD disc, it is important that the algorithms must operate efficiently.

2.4 Applications of watermarking

In this section, we introduce some typical applications of digital watermarking.

CHAPTER 2. WATERMARKING BASICS

2.4.1 Copyright protection

Copyright protection, which is probably the most prominent application of digital watermarking, is to embed some identification information so that the copyright owner can prevent other parties from reclaiming the copyright on the work. The product might have been manipulated by an adversary in an effort to remove or destroy the watermark or replace it with his own. Therefore, a watermark for copyright protection requires a high level of robustness. The driving force for this application is that the rightful owners want to protect the copyrights of a tremendous amount of digital images, music files (e.g. MP3 songs) and movies, which are constantly subject to piracy and illegally copying on the Internet [3].

Additional issues besides robustness have to be considered. For example, the watermark must be unambiguous and still resolve right ownership if other parties embed additional watermarks. Therefore, a TTP (trusted third party), which is responsible for performing embedding and detection on behalf of the content owners, would be necessary [10].

2.4.2 Content authentication

In some circumstances (e.g., the authentication for images distributed by news agencies), the copyright owner wants to ensure that the content is authentic. If the content has been modified, the copyright owner may hope to know how and to what extent, the work has been impaired. This could be achieved with fragile or

CHAPTER 2. WATERMARKING BASICS

semi-fragile watermarks, which are able to reflect the modifications the work has undergone. Among all the possible watermarking applications, watermarks for authentication require the lowest level of robustness. It is more challenging to design an accurate semi-fragile watermark, which is resistant to certain modifications (e.g., lossy compression), while in the meantime sensitive to all the other manipulations or malicious attacks [8].

2.4.3 Fingerprinting

In the scenario of fingerprinting, the distributor usually embeds different watermarks into each distributed copy of the product, in order to monitor or trace illegal copies. Multiple-bit watermarks should be employed to enable user identification. In fingerprinting, collusion attacks must be considered, because many individually watermarked copies of the same product are distributed [7].

2.4.4 Copy control

The content provider is sometimes interested in constraining the level of control that the users may have on the purchased product. It is possible to embed a copy information into a product and use this information to indicate the copy status of the data. Depending on the information of copy status, the devices can allow or prohibit certain operations on the content. An example is that the DVD or CD player with devices to embed or detect copy control information, could prohibit

CHAPTER 2. WATERMARKING BASICS

illegal copying of the discs [11]. In the next chapter, we will briefly review various watermarking algorithms in the literature.

Chapter 3

Algorithm Review

In this chapter, we will briefly review the existent watermarking algorithms in terms of the spatial domain and transform domain. The classification of the existing algorithms will be introduced. Various attacks, which aim to destroy the watermarks will also be discussed. Two benchmarking tools for watermarking will also be introduced.

3.1 Classification of watermarking techniques

The existing digital watermarking techniques can be classified into several categories as follows [7]:

CHAPTER 3. ALGORITHM REVIEW

3.1.1 Spatial domain and transform domain watermarking

Watermarking methods can be classified into two categories: spatial domain and transform domain. In the spatial domain, least significant bit (LSB) coding is usually used for watermark embedding. The advantages of the spatial-domain-based methods include low computational load and negligible degradation to the watermarked signals. These methods, however, are usually vulnerable to common signal processing manipulations (e.g., compression), and thus cannot meet the requirement for robustness in many cases like copyright protection. On the other hand, transform domain watermarking, which inserts the watermark into the frequency components of the cover signal, has been widely accepted for robust watermarking [6]. In this chapter, we will review the watermarking algorithms based on this classification.

3.1.2 Robust, semi-fragile and fragile watermarking

According to the requirement for robustness, the watermarking methods can be divided into robust, fragile and semi-fragile watermarking. A robust watermark is designed to survive various unintentional and malicious attacks. Such watermarking schemes can be used for copyright protection and owner verification. A fragile watermark is sensitive to all kinds of attacks, which can be employed for tamper proofing and content authentication. While a semi-fragile watermark is expected to be able to tolerate some specific manipulations such as inherent system processing or attacks, such as compression, but it can detect malicious attacks. Semi-fragile

CHAPTER 3. ALGORITHM REVIEW

watermarks are mostly used for content authentication.

3.1.3 Private, semi-private and public watermarking

In terms of the required data during detection, the watermarking techniques can be categorized into private, semi-private and public watermarking [8]. Private watermarking, which is also referred to as non-blind watermarking, requires at least the original data to detect the embedded watermark. Semi-private watermarking, or semi-blind watermarking does not require the original data, but the original watermark is necessary for detection. While public watermarking, which is also called blind or oblivious watermarking, requires neither the original data nor the original watermark for detection. Semi-private and public watermarking approaches are more desirable in practice, as private watermarking needs huge storage for the original data. Furthermore, in some applications (e.g., copy control), the original is not even available or accessible.

3.1.4 One-bit and multi-bit watermarking

A digital watermark can be represented in various forms, depending on the applications. For some applications such as a copy control system, a “Yes” or “No” answer, i.e., 0 or 1, is sufficient to complete the purpose of protection. We call such watermarks as one-bit watermarks. However, in other watermarking systems such as copyright protection and fingerprinting systems, a logo or a sequence of characters,

CHAPTER 3. ALGORITHM REVIEW

would be required. These watermarks are called multi-bit watermarks.

There are many other watermarking classifications [12]. For example, in terms of the type of the cover data, there are image, video, audio and document watermarking. According to the perceptual quality of the watermark, there are also visible and invisible watermarking.

3.2 Spatial domain watermarking

In a watermarking system, the watermark is embedded by modifying the cover data in either spatial domain or transform domain. In early spatial domain watermarking schemes, LSB coding was the most common method used to insert a watermark [13].

In [2], Tirkel et al. proposed two methods based on LSB coding for digital watermarking. In the first approach, an m-sequence-derived PN code was embedded in the LSB plane of the grayscale image. The watermark decoding was straightforward because the LSB plane carried the watermark directly. In the second method, the difference from the first approach was in the detection process, which exploited the unique and optimal autocorrelation function of the m-sequence. In [14], the idea of using m-sequence and LSB coding was extended by using two-dimensional (2-D) m-sequence which resulted in more robust watermarks.

Matsui and Tanaka [15] introduced three watermarking techniques for grayscale, facsimile, color still images and video frames. In the first method, the digitized 8-bit image data $\{x_i, i \in N\}$ which consisted of a set of values between 0 and 255,

CHAPTER 3. ALGORITHM REVIEW

was converted to a sequence in which the first element was x_1 and the subsequent elements were the difference between successive points, i.e., $\Delta_i = x_i - x_{i-1}$. For watermark embedding, a cipher key table, which assigned a corresponding bit c_i to all possible differences Δ_i , was generated. Table 3.1 illustrates an example of a cipher key table. To embed a bit b , select a pixel x_i with its corresponding difference Δ_i . Then it checked in the cipher table to see whether the bit value c_i corresponding to Δ_i had the same value as the embedded bit b , as described in the following equation.

$$\Delta_i = \begin{cases} \Delta_i & \text{if } c_i = b \\ \Delta_j & \text{otherwise.} \end{cases} \quad (3.1)$$

where Δ_j denotes the nearest difference which satisfies $c_j = b$. The distortion introduced in the embedding process is $error_i = \Delta_j - \Delta_i$, which is usually on the same order as noise, i.e., negligible. The watermark was retrieved by looking up the cipher key table.

Δ_i	$\dots$	-3	-2	-1	0	1	2	3	$\dots$
c_i	$\dots$	0	1	1	0	1	1	0	$\dots$

Table 3.1: An example of cipher key table.

The second method modified the ordered dithering scheme for binary images. An image was first divided into 4x4 pixel blocks and brightness thresholds $\{x_i, i = 0, 1, \dots, 15\}$ for each of the 16 pixels in the block were assigned from top to bottom, left to right. Next, define sets

$$S_k = \{(x_i, x_j)_k : x_j - x_i = k\}, i, j = 0, 1, \dots, 15; i \neq j. \quad (3.2)$$

CHAPTER 3. ALGORITHM REVIEW

Let $(y_i, y_j)_k$ be a pair of output signals that pass through $(x_i, x_j)_k$. Then $(y_i, y_j)_k$ is either $(0, 0)$, $(1, 0)$, $(0, 1)$, or $(1, 1)$, where 0 indicates that the pixel will be turned “off” and 1 indicates “on”. Only the pairs $(1, 0)$ or $(0, 1)$ will be used to embed a sequence of bits $B = \{b_n : b_n = 0 \text{ or } 1, n \in N\}$. To embed $b_n = 0$, set $(y_i, y_j)_k = (0, 1)$. To embed $b_n = 1$, set $(y_i, y_j)_k = (1, 0)$. To decode, disregard the $(0, 0)$ and $(1, 1)$ outputs and simply reverse the procedure described above.

The third scheme was proposed to watermark facsimile documents. The embedding algorithm was based on the fact that the data would be compressed using run-length coding (RLC) and Huffman coding. A watermark bit was embedded by shortening or lengthening runs. Decoding an embedded bit was achieved by deriving the decoded run length.

Bender et al. [16] proposed two methods for data hiding. The first method was a statistical approach, which was called “Patchwork”. A subset of the pixels in an image was first divided into two distinct sets, $\{a_i, i = 0, 1, \dots, N\}$, $\{b_i, i = 0, 1, \dots, N\}$. The brightness of one set of pixels was increased by one, and those from the other set were decreased by one. Provided that the image satisfied some statistical properties, the expected value of the sum of the differences between the two sets was given by $2N$,

$$\sum_N a_i - b_i = \begin{cases} 2N, & \text{for watermarked sets} \\ 0, & \text{for non-watermarked sets.} \end{cases} \quad (3.3)$$

The second method was based on texture block coding. To embed a watermark bit, a block from a region with a random texture was copied and placed in a region

CHAPTER 3. ALGORITHM REVIEW

with similar texture. To recover the watermark, the autocorrelation function had to be computed. This method was sensitive to cropping. Furthermore, texture block coding was easy to detect since anyone could make slides of the cover image and its opposite and extract the embedded block or region.

Pitas [17] suggested a signature casting approach on digital images, which was similar to Patchwork. A binary watermark $S = \{s_{m,n}\}$, with the same size as the cover image, consisted of the same amount of “ones” and “zeros”. The cover image I , with luminance values $x_{m,n}$, was partitioned into two parts A and B of equal size as follows,

$$\begin{aligned} A &= \{x_{mn} \in I, s_{mn} = 1\} \\ B &= \{x_{mn} \in I, s_{mn} = 0\} \end{aligned} \tag{3.4}$$

To embed a watermark, the elements of A were increased by a positive integer k , i.e., $A' = \{x_{mn} + k, x_{mn} \in A\}$, while B remained unchanged. To detect the watermark, the test statistic q , which was defined as the normalized difference between the mean $\bar{a}'$ of set A' and the mean $\bar{b}$ of set B , was computed as follows:

$$q = \frac{\bar{b} - \bar{a}'}{\sigma_{A'}^2 + \sigma_B^2} \tag{3.5}$$

where $\sigma_{A'}^2$ and σ_B^2 denote the sample variance of set A' and B , respectively. By comparing q with a certain threshold, it could be decided whether there was a watermark in an inspected image.

Bruyndonckx et al. also employed a block-partitioning-based method but the change in the mean values of the luminance in blocks was used for embedding [18]. The

CHAPTER 3. ALGORITHM REVIEW

blocks were classified into three types of luminance: hard, progressive and noise contrast. The pixels were then assigned to zones. Each zone was further subdivided into two categories A and B based on a grid defined by the coder. Each pixel was thus assigned to one of four zone/category combinations, i.e., $1/A, 1/B, 2/A, 2/B$. Embedding a bit b was carried out by modifying the zone/category mean values to satisfy the following constraints.

$$\begin{aligned}
 \text{if } b = 0: \quad m_{1B}^* - m_{1A}^* &= L \\
 m_{2B}^* - m_{2A}^* &= L \\
 \text{if } b = 1: \quad m_{1A}^* - m_{1B}^* &= L \\
 m_{2A}^* - m_{2B}^* &= L
 \end{aligned} \tag{3.6}$$

where m_{1A}^* , m_{1B}^* , m_{2A}^* and m_{2B}^* are the zone/category mean values after embedding and L the watermark embedding level. Robustness against JPEG compression depended strongly on the compression ratio, embedding level, and grid sizes. Redundant embedding with error-correcting codes (ECC) was effective in reducing errors in the extracted messages.

In the approach proposed by Voyatzis and Pitas [19], a small image watermark was first applied torus automorphism prior to insertion, in order to enhance the robustness of the watermark. A 2-D torus automorphism could be considered as a spatial transformation of planar regions which belonged to a square 2-D area. An example for a well-known automorphism in dynamics was the “cat map”, defined

CHAPTER 3. ALGORITHM REVIEW

by

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} 2 & 1 \\ 1 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} \pmod{1} \quad (3.7)$$

After automorphism, the mixed watermark was then superimposed to the cover image by using an embedding function such as LSB coding.

More recently, Chen and Wornell [20] proposed a new embedding method called quantization index modulation (QIM), in which the watermark message was viewed as an index to select a particular quantizer from an set of possible quantizers and then the watermark was embedded by applying the selected quantizer. In the decoding process, a distance identified the embedded information. In fact, this method could also embed a watermark in frequency domain by quantizing transform coefficients. The authors showed that the performance of QIM was superior to standard spread-spectrum modulation algorithms.

3.3 Transform domain watermarking

One drawback of spatial-domain-based watermarking approaches is their high sensitiveness to common signal processing manipulations. A very common JPEG compression for instance, could destroy most of the spatial domain watermarks proposed in the literature.

It was found that a watermark inserted into a frequency domain was much more robust than that in spatial domain [1]. In the frequency domain, it becomes possible

CHAPTER 3. ALGORITHM REVIEW

to insert the watermark bits into the locations where they are least perceptible, e.g., within the texture of an image. A significant number of transform-domain-based watermarking methods have been proposed [10]. A variety of transforms, such as discrete Cosine transform (DCT), discrete Fourier transform (DFT), discrete Wavelet transform (DWT), Hadamard transform (HT), Fourier-Mellin transform (FMT), and Radon transform have been investigated for watermark embedding. In this section, we will introduce some typical watermarking algorithms based on several most commonly used transforms.

3.3.1 Discrete Cosine transform

Given a 2-D signal $f(x, y)$, the discrete Cosine transform is defined by

$$F(u, v) = \frac{2}{N} C(u) C(v) \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} f(x, y) \cos \frac{\pi u(2x+1)}{2N} \cos \frac{\pi v(2y+1)}{2N} \quad (3.8)$$

where

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

The inverse of DCT (IDCT) is given by

$$f(x, y) = \frac{2}{N} C(u) C(v) \sum_{u=0}^{N-1} \sum_{v=0}^{N-1} F(u, v) \cos \frac{\pi u(2x+1)}{2N} \cos \frac{\pi v(2y+1)}{2N} \quad (3.9)$$

The two-dimensional DCT has been widely used in the area of data compression. DCT based compression algorithms have been adopted in the most popular compression standards JPEG and MPEG for digital images and video, respectively.

There are a number of advantages to employ DCT for watermarking. First, the

CHAPTER 3. ALGORITHM REVIEW

human visual system (HVS) models in the DCT domain (e.g. Watson model [21]) developed for data compression could be reused for watermarking, because the goal of watermarking is to maximize the watermark strength factor while maintaining the integrity of the cover signal at the same time. Furthermore, the DCT-based watermarks are often more robust to JPEG/MPEG compression [7], which are very common signal processing manipulations for a watermarked image or video.

Frequency domain watermarking was first introduced by Boland et al. [1] and Cox et al. [6]. Boland et al. proposed a watermarking approach which combined amplitude modulation and frequency shift keying. They also suggested that different transform domains, e.g., DCT, wavelet transform, Walsh-Hadamard transform, and the fast Fourier transform (FFT) could be employed for watermark embedding.

In [6], Cox et al. proposed a watermarking method based on spread-spectrum communication, in which a noise-like watermark sequence was considered as a narrow-banded signal and spread over a wider frequency bandwidth before embedding. Moreover, they suggested that the watermark should be inserted into the perceptually important frequency components, in order to achieve higher robustness. In their experiments, 1000 largest DCT coefficients were used to embed the watermark. Three embedding schemes were suggested for watermark insertion,

$$v'_i = v_i + \alpha x_i \quad (3.10)$$

$$v'_i = v_i(1 + \alpha x_i) \quad (3.11)$$

$$v'_i = v_i e^{\alpha x_i} \quad (3.12)$$

CHAPTER 3. ALGORITHM REVIEW

where v'_i , v_i and x_i denote the modified frequency components, the original frequency components and the watermark, respectively. α determines the watermark strength. In most cases, Equation 3.11 is used, because the embedding strength is adaptive to the different spectral components and thus the embedding-induced distortion could be minimized. To verify the presence of the watermark, the normalized correlation between the original watermark and the retrieved watermark was computed, which was given by

$$\text{sim}(X, X^*) = \frac{X^* \cdot X}{\sqrt{X^* \cdot X^*}} \quad (3.13)$$

where X and X^* denote the original and recovered watermarks, respectively.

Tao and Dickinson [22] suggested an adaptive watermarking technique in the DCT domain. The technique was based on a regional perceptual classifier with assigned sensitivity indices. Some AC DCT coefficients were selected as to have the smallest quantization step sizes according to the default JPEG compression table. To determine the noise sensitivity, the proposed classification algorithm which exploited luminance masking, edge masking and texture masking effects of the HVS, classified the blocks into six perceptual classes: edge, uniform, low sensitivity, moderately busy, busy and very busy, in descending order of noise sensitivity. Each perceptual class had a noise-sensitivity index assigned to it. The watermark recovery in their method required both the original image and the watermark.

In [23], Podilchuk and Zeng presented a perceptual watermark using just noticeable difference (JND), which was derived from the Watson model [21] in the DCT domain, to determine an image-dependent watermark modulation mask. The watermark

CHAPTER 3. ALGORITHM REVIEW

modulation for the selected coefficients can be described as

$$I_{u,v}^* = \begin{cases} I_{u,v} + JND_{u,v} \times w_{u,v}, & \text{if } I_{u,v} > JND_{u,v} \\ I_{u,v}, & \text{otherwise.} \end{cases} \quad (3.14)$$

where $I_{u,v}$ and $I_{u,v}^*$ denote the original and watermarked transform coefficients, respectively. $w_{u,v}$ are the watermark values, and $JND_{u,v}$ is computed based on the visual model. The authors also proposed a wavelet image-adaptive watermarking method which was based on a very similar mechanism. For detection, the maximum correlation between the original watermark and the retrieved watermark was compared to a threshold to determine whether an image contained the watermark in question.

3.3.2 Discrete Fourier transform

Given a 2-D signal $f(x, y)$, the discrete Fourier transform is defined by

$$F(k_1, k_2) = \sqrt{\frac{1}{N_1 N_2}} \sum_{n_1=0}^{N_1-1} \sum_{n_2=0}^{N_2-1} f(n_1, n_2) \exp\left(\frac{-i2\pi n_1 k_1}{N_1} - \frac{-i2\pi n_2 k_2}{N_2}\right) \quad (3.15)$$

where $i = \sqrt{-1}$.

The inverse DFT (IDFT) is given by,

$$f(n_1, n_2) = \sqrt{\frac{1}{N_1 N_2}} \sum_{k_1=0}^{N_1-1} \sum_{k_2=0}^{N_2-1} F(k_1, k_2) \exp\left(\frac{-i2\pi k_1 n_1}{N_1} - \frac{-i2\pi k_2 n_2}{N_2}\right) \quad (3.16)$$

The DFT domain was considered as a workspace for watermarking as it has been widely studied in the field of signal processing. Ruanaidh et al. [24] proposed a phase-modulation-based watermarking approach. To embed a watermark bit, the

CHAPTER 3. ALGORITHM REVIEW

phase of a selected coefficient $F(k_1, k_2)$ in a DFT transformed image of size (N_1, N_2) was modified by

$$\angle F(k_1, k_2) \leftarrow \angle F(k_1, k_2) + \delta \quad (3.17)$$

The phase should satisfy negative symmetry because the watermarked image must be real. Thus, additional modification was necessary:

$$\angle F(N_1 - k_1, N_2 - k_2) \leftarrow \angle F(N_1 - k_1, N_2 - k_2) + \delta. \quad (3.18)$$

If the cover image was available, it was easy to retrieve the watermark by comparing the phase. If the decoder could not access the cover image, the original phase must be pre-quantized prior to embedding. Then the deviations between the quantized states could be used to convey the data.

Solachidis and Pitas [25] proposed to embed a circularly symmetric watermark, as depicted in Figure 3.1, into the DFT frequency domain. The binary watermark was a zero mean 2-D signal, which had the same amount of 1's and -1 's. The size of the watermark was the same as the cover image. The watermark was embedded into the magnitude of the DFT transformed image as follows:

$$M'(k_1, k_2) = M(k_1, k_2) + f(M(k_1, k_2), W(k_1, k_2), \alpha) \quad (3.19)$$

where $M'(k_1, k_2)$ and $M(k_1, k_2)$ denote the modified and original magnitude, and α is the embedding strength.

The embedding function $f(M(k_1, k_2), W(k_1, k_2), \alpha)$ could be additive or image dependent. In the case of additive watermarking, the embedding function had the

CHAPTER 3. ALGORITHM REVIEW

Figure 3.1: The circularly symmetric watermark

form $f(M, W, \alpha) = \alpha W(k_1, k_2)$. The simplest form of multiplicative embedding was $f(M, W, \alpha) = \alpha M(k_1, k_2)W(k_1, k_2)$. Correlation was used for watermark detection and the cover image was not required. The algorithm was reported to survive some geometrical attacks, such as scaling, translation, cropping and rotation.

Tang and Hang [5] presented a DFT-based robust watermark scheme for copyright protection, which aimed to resist both geometrical distortions and signal processing attacks. First, a feature extraction method was used to generate some salient points, i.e., reference centers of disks, for watermark embedding and detection. The location of blocks in the cover image for watermark embedding was determined from image normalization, which was rotation invariant. As a result, the watermark synchronization during detection was achieved. Next, a 2-D DFT was applied to these 32x32 blocks on each qualified disk in a cover image. Pairs of middle frequency coefficients were selected for embedding. The selected pairs were (x_i, y_i) and $(-y_i, x_i)$,

CHAPTER 3. ALGORITHM REVIEW

which were 90 degrees apart. The watermark bits were embedded by modifying the pairs to satisfy

$$\text{if } wm_i = 1: F'(x_i, y_i) - F'(-y_i, x_i) \geq \alpha \quad (3.20)$$

$$\text{if } wm_i = 0: F'(x_i, y_i) - F'(-y_i, x_i) \leq -\alpha$$

where $F'(x_i, y_i)$ and $F'(-y_i, x_i)$ are magnitudes of the altered coefficients at locations (x_i, y_i) and $(-y_i, x_i)$ in the DFT domain, α denotes the watermark strength, and wm_i is the binary watermark bit.

The watermark extraction was relatively straightforward by examining the difference between the two selected coefficients, which was simply the inverse of the embedding process. The simulation results showed that the proposed scheme could survive a number of geometrical attacks, such as rotation, cropping, shearing and line removal.

3.3.3 Discrete Wavelet transform

The study of the wavelet transform has thrived in the past two decades. Now, the wavelet transform is considered to be a fairly simple mathematical tool, and it has many applications in various fields. Compared with the traditional transforms, for example Fourier transform, the wavelet transform has an advantage of achieving both spatial and frequency localization. In digital signal and image processing, the discrete wavelet is closely related to a series of filter banks. Figure 3.2 shows an example of a three level wavelet decomposition of LENA. A typical two-channel decomposition and reconstruction structure is given in Figure 3.3.

CHAPTER 3. ALGORITHM REVIEW

Figure 3.2: Three-level wavelet decomposition of LENA

The two-channel filter banks must be orthogonal and are defined by

$$\begin{aligned} H(\omega) &= \sum_k h_k e^{-jk\omega} \\ G(\omega) &= \sum_k g_k e^{-jk\omega} \end{aligned} \quad (3.21)$$

The iterative process of decomposition is given by

$$\begin{aligned} c_{j-1,k} &= \sum_n h_{n-2k} c_{j,n} \\ d_{j-1,k} &= \sum_n g_{n-2k} c_{j,n} \end{aligned} \quad (3.22)$$

While the iterative reconstruction process is defined as

$$c_{j,n} = \sum_k h_{n-2k} c_{j-1,k} + \sum_k g_{n-2k} d_{j-1,k} \quad (3.23)$$

Built on Cox et al.'s work [26], Xia et al. [27] proposed to embed a Gaussian random noise watermark into a DWT transformed cover image, by modifying the largest

CHAPTER 3. ALGORITHM REVIEW

(a) Wavelet decomposition.

(b) Wavelet reconstruction.

Figure 3.3: Two-channel wavelet decomposition and reconstruction structures.

CHAPTER 3. ALGORITHM REVIEW

wavelet coefficients in the middle and high frequency bands. The decoding was hierarchical. If distortion of a watermarked was not serious, only a few bands worth of information were needed to detect the signature and therefore computational load could be saved. The results showed that the DWT-based algorithm achieved better robustness against a number of attacks, such as compression, additive noise and image halftoning, compared to Cox's DCT-based approach [26].

Kundur [28] embedded a binary watermark by modifying the amplitude relationship of three transform domain coefficients from distinct detail subbands of the same resolution level of the cover image. Each selected coefficient tuple was sorted and the middle coefficient was quantized to encode either a zero or a one bit. To strengthen the blind watermark extraction process, Kundur resorted to repetition and a reference mask.

Hsu and Wu [29] proposed a multiresolution watermark embedding algorithm, in which both the watermark image and the cover image were consposed of multiresolution representations with different structures. The decomposed watermarks of different resolution were embedded into the corresponding resolution of the decomposed images. The low-resolution rendition of the watermark was preserved within the corresponding low-resolution components of the image. The experimental results showed that the proposed technique was robust to common image processing operations such as JPEG lossy compression.

Pereira [30] described a method based on one-level decomposition of non-overlapping 16x16 image blocks using Haar wavelet filters. The proposed watermarking algo-

CHAPTER 3. ALGORITHM REVIEW

rithm used linear programming to optimize watermark robustness within a visual distortion constraint given by JND threshold maps. For each bit to be embedded, a 2x2 block of neighboring coefficient was selected from an LL subband of size 8x8. The information was embedded by using differential coding. It was important to select neighboring coefficients since it was assumed that their difference was zero on average.

In [31], Wang et al. suggested the use of a binary watermark pattern whose size was the same as the selected wavelet band. If the size of the watermark was smaller than the selected band, the watermark size was increased by padding minus ones. The orthogonal filter banks used for transformation was randomly selected as a part of the secret key to enhance security. The watermark was first scrambled and modulated prior to embedding, and then inserted to the selected wavelet band by replacing.

Hsieh et al. [32] proposed an image accreditation technique by embedding digital watermarks in images. A watermark with visual recognizable patterns, such as binary, gray or color images was embedded by modifying the DWT frequency coefficients. A multi-energy watermarking scheme based on the qualified significant wavelet tree (QSWT) was used to achieve the robustness of the watermarking. Unlike other watermarking techniques that used a single casting energy, QSWT adopted adaptive casting energy in different resolutions. The performance of the method was resistant to a variety of signal distortions, such as JPEG compression, image cropping, sharpening, median filtering and incorporating attacks.

CHAPTER 3. ALGORITHM REVIEW

3.3.4 Hadamard transform

The 2D-Hadamard transform has been used extensively in image processing and image compression. In this section we first briefly introduce some fundamentals of the Hadamard transform.

The core Hadamard matrix is defined by

$$H_1 = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \quad (3.24)$$

The Hadamard matrix of the order n is generated in terms of a Hadamard matrix of order $n - 1$ using Kronecker product:

$$H_n = H_{n-1} \otimes H_1 \quad (3.25)$$

or

$$H_n = \begin{bmatrix} H_{n-1} & H_{n-1} \\ H_{n-1} & -H_{n-1} \end{bmatrix} \quad (3.26)$$

Ho et al. [33] proposed a Hadamard-transform-based watermarking approach for image-in-image watermarking, in which an 8x8 Hadamard transform was applied to both the cover and watermark images. The Hadamard matrix of an 8x8 transform is H_3 , given by

CHAPTER 3. ALGORITHM REVIEW

$$H_3 = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 \\ 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 & -1 & 1 & -1 & 1 \\ 1 & 1 & -1 & -1 & -1 & -1 & 1 & 1 \\ 1 & -1 & -1 & 1 & -1 & 1 & 1 & -1 \end{bmatrix} \quad (3.27)$$

The watermark was embedded by replacing the Hadamard coefficients as follows:

$$x' = \alpha w \quad (3.28)$$

where x' and w are modified cover image's Hadamard coefficients and Hadamard coefficients of transformed watermark, respectively. α is the watermark strength.

The strength factor α was determined by two HVS-based masking models.

$$\alpha = \beta \times \text{mask}_1(j, k) \times \text{mask}_2(j, k) \quad (3.29)$$

where $\text{mask}_1(j, k)$ is used to determine the texture by examining AC energy, $\text{mask}_2(j, k)$ reflects the edge density by employing Canny edge detection [34], and β denotes a certain scaling factor.

The experimental results showed that the proposed Hadamard-based watermarking algorithm achieved a better performance of robustness against a variety of at-

CHAPTER 3. ALGORITHM REVIEW

tacks, such as additive noise and median filtering, compared to a similar approach in the DCT domain.

Our proposed card authentication, which will be discussed in Chapter 6, also employed a Hadamard transform. The reason that we chose Hadamard transform is mainly because of its simplicity for hardware implementation and less processing time.

3.4 Attacks

In this section, we classify all possible types of attacks in terms of their effect on the watermark and the way the watermark is interpreted by the detector. The attacks are divided into four categories [12, 35, 36].

3.4.1 Removal attacks

In this category, the attacks aim to remove the watermark without cracking the security of the watermarking algorithm (e.g., without the key used for watermark embedding). This category includes denoising, lossy compression, quantization, remodulation, collusion and averaging attacks. Sophisticated removal attacks try to optimize operations like denoising or quantization to impair the embedded watermark as much as possible while keeping the quality of the attacked document high enough. Collusion attacks [37] are applicable when many copies of a given data set

CHAPTER 3. ALGORITHM REVIEW

can be obtained by an adversary.

3.4.2 Presentation attacks

Instead of removing the embedded watermark itself, these attacks aim at manipulating the watermarked signal in such a way that the detector cannot find the watermark, i.e., the synchronization is lost. The intention of presentation attacks is essentially the same as in removal, but the techniques employed to achieve it are different. The attacks in this category include mosaic attacks, rotation, scaling and translation (RST), change of aspect ratio, shearing and affine transformations.

3.4.3 Interpretation (protocol) attacks

The interpretation attacks, also known as protocol attacks, aim to attack the concept of the watermarking application. For example, Craver et al. [38] proposed a framework of invertible watermark and showed that the watermarks required to be non-invertible for copyright protection applications. The requirement of non-invertibility on the watermarking technology implies that it should not be possible to extract a watermark from a non-watermarked image.

3.4.4 Legal attacks

This category is quite different from the ones presented above, since it implies all the actions that can be taken in a law court in order to damage the credibility

CHAPTER 3. ALGORITHM REVIEW

of watermarks as proofs of ownership or authenticity in case of disputes. In other words, it does not include manipulations of people, but attempts to take advantage of the lack of legal foundation on watermarking as a proof of ownership (i.e., gaps in the legislation on copyright laws), and challenging the credibility of the owner [12].

3.5 Benchmarking software

In this section, two well-known tools for watermark benchmarking are introduced. Our experimental results, which will be reported in Chapter 5, are based on the testing with both Stirmark and Checkmark.

3.5.1 Stirmark

The first version of Stirmark [39, 40] was published in November, 1999. Stirmark is a generic tool for simple robustness testing of image watermarking algorithms and other steganographic techniques. It can be applied to digital images and it will distort the watermark such that the embedded watermark or steganographic message cannot be detected and decoded from the resultant image. It consists of around 90 different types of image manipulations or attacks. It introduces random bilinear geometric distortions to de-synchronize watermarking algorithms. The attacks in Stirmark can be divided into nine categories: signal enhancement, compression, scaling, cropping, shearing, rotation, linear transformations, other geometric transformations, and random geometric distortions.

3.5.2 Checkmark

Checkmark [41] was developed by researchers at University of Geneva, Switzerland in 2001. Besides the attacks adopted from the StirMark, Checkmark adds several new classes of attacks, which include wavelet compression, projection transformations and warping, copy attack, template removal attack, denoising and denoising followed by perceptual remodulation, non-linear line removal, collage attack, down/up sampling, dithering and thresholding. Most of the added attacks are non-geometric attacks that aim to remove the watermark directly or crack the entire watermarking system. In this thesis, we tested our watermarking algorithms with the non-geometric attacks in Checkmark and the experimental results and analysis will be presented in Chapter 5.

Chapter 4

Proposed Methodologies

A significant number of embedding methods have been proposed in the literature, for example, additive embedding [26], substitutive embedding [33], amplitude modulation [42, 43] and phase modulation [24]. More recently, Chen and Wornell [20] proposed a new embedding method called quantization index modulation (QIM), in which the watermark message was viewed as an index to select a particular quantizer from an set of possible quantizers and then the watermark was embedded by applying the selected quantizer. However, QIM is very sensitive to the scaling attack [44].

Detection techniques corresponding to the embedding methods were also developed by researchers. Some early detection methods [26, 23] had to employ the cover data. Such private watermarking schemes required the test data to be subtracted from the cover data, and then the watermark could be retrieved. However, these

CHAPTER 4. PROPOSED METHODOLOGIES

watermarking systems were proved to be undesirable for some real-world watermarking applications. This was because a large volume of original data had to be stored for watermark detection, which would increase the computational overhead and the cost of system implementation. Furthermore, such watermarks were also exposed to protocol attacks [45]. Semi-private or public watermarking schemes, which do not require the original data for detection, are more desirable. Most of the recent watermarking research [3, 4] belongs to these two categories.

The proposed watermarking methodologies are discussed in this chapter. The first method is a semi-private watermarking approach, which requires a linkage sequence for watermark decoding. In the second method, a spread-spectrum technique is employed and it requires neither the cover image nor the original watermark for detection. Our proposed methods are based on the two-level quantization and distortion bound. In both of the two methods, the selected transform coefficients of the cover signal are two-level quantized, and by using amplitude modulation function their amplitudes are modulated to a pre-defined threshold, which is determined by the estimated distortion bound. The two-level quantizer is again used for watermark retrieval. The watermark bits are essentially embedded into the polarities of the transform coefficients.

4.1 Background

4.1.1 Two-level quantization

The two-level quantizer is defined as follows.

$$Q(x) = \begin{cases} 1 & x \geq 0 \\ -1 & \text{else} \end{cases} \quad (4.1)$$

Many watermarking methods employed the two-level quantizer as the sign detector for watermark detection [44, 46]. The sign detector addresses the problem of detecting signals by using very coarse information about the statistical behavior of the noise.

In a robust watermarking scenario, the perturbation noise is unknown and thus its statistical characteristic may not be simply estimated [47]. Therefore, the two-level quantization could be employed as a useful alternative detection method which achieves a good trade-off of robustness and complexity. In this thesis, we employ the two-level quantizer for watermark retrieval. Furthermore, prior to embedding the watermark signal, we also quantize the selected frequency coefficients of the cover signal into two levels. Thus, instead of embedding the information by modifying the amplitude of the coefficients, we essentially insert the watermark bits into the polarities of the frequency components.

CHAPTER 4. PROPOSED METHODOLOGIES

4.1.2 Distortion bound

In a general digital watermarking scheme, the embedded information is superimposed to the cover signal by introducing small modifications. The watermark embedding process can be described by

$$\mathbf{s} = \mathbf{x} + \mathbf{w} \quad (4.2)$$

where $\mathbf{x}$ denotes a representation (e.g., a sequence of transform coefficients) of the cover, $\mathbf{w}$ denotes the watermark signal and $\mathbf{s}$ the watermarked signal. The watermarked signal $\mathbf{s}$ generally goes through a distortion channel.

$$\mathbf{y} = \mathbf{s} + \mathbf{n} \quad (4.3)$$

where $\mathbf{y}$ and $\mathbf{n}$ denote the distorted watermarked signal and the perturbation noise, respectively. Thus, the distortion caused by the channel is:

$$E[D] = \|\mathbf{n}\| = \|\mathbf{y} - \mathbf{s}\| \quad (4.4)$$

The distortion $\mathbf{n}$ could be from both common signal manipulations (such as format conversion) and intentional attacks by adversaries. Therefore, it is very difficult to completely characterize the distortion channel [48]. However, the objective of an adversary is not only to destroy the embedded watermark but also to reclaim the ownership of the documents. Thus, the attacks are constrained by some specific criteria (e.g., image's visual quality).

The distortion bound can be defined as the range of degradation caused by the possible attacks, while the quality of the attacked signal still satisfies the defined

CHAPTER 4. PROPOSED METHODOLOGIES

criteria. However, the distortion bound is difficult to calculate exactly.

The first reason is that it is difficult to define a general subjective or objective quality measurement method to assess the visual quality of the multimedia contents such as an image or a video frame [48]. Thus, the strength of attacks could be very different because of the different quality measurement methods. Furthermore, at the embedding stage, the distortion channel is unknown. The attack could be a single manipulation, combination of many manipulations (for example, the digital-analogue-digital conversion), or even a totally new attack to the embedder. However, the distortion bound can be estimated and used during watermark embedding after we investigate how the common attacks degrade the watermarked signal.

An 8x8 Discrete Cosine Transform (DCT) is chosen in our experiments. Three test images are shown in Figure 4.1. Figure 4.2, Figure 4.3 and Figure 4.4 illustrate the difference of frequency coefficients between the cover image subblocks B_{ij} and the distorted subblocks $\widehat{B}_{ij}$ for the three testing images, respectively. For simplicity, only the left upper corner 32x32 sub-images of the three test images are used for illustration.

For JPEG compression, a quality factor of 30 was used as an attack for this experiment. From the figures, it can be easily observed that the distortion of most frequency components caused by JPEG 30 is approximately within a range of 20. By trying several other attacks, such as rescaling, filtering and additive noise, we found that although some coefficients changed drastically, most of the coefficients' amplitude change was within a range of approximately 20, especially for the low

CHAPTER 4. PROPOSED METHODOLOGIES

frequency portion. This is reasonable because the frequency distribution of the attacked image still remains intact. The lower frequency coefficients represent the more basic information of the original image and thus they are more stable.

A simple method to estimate the distortion bound T_d is to calculate the mean of the difference between I_{ij} and $\widehat{I}_{ij}$.

$$T_d = \frac{1}{M \times N} \times \sum_{m=1}^M \sum_{n=1}^N |I_{mn} - \widehat{I}_{mn}| \quad (4.5)$$

where I_{ij} and $\widehat{I}_{ij}$ denote the cover image and distorted image, both of size (M, N) . T_d can be accurately determined if the watermark encoder knows what kind of attacks would be imposed to the watermarked media. However, for some applications (e.g., copyright protection), the watermark encoder has no knowledge about the attacks. In our experiments, we found that T_d usually changed within the range of 25, provided that the visual quality of the attacked images maintains acceptable.

4.2 Proposed semi-private watermarking algorithm

In this section, we introduce the watermark embedding and retrieval algorithms of the proposed semi-private watermarking method. A general framework of the proposed watermarking method is depicted in Figure 4.5.

In this framework, before embedding, a two-level quantizer $Q(\cdot)$ is performed to $\mathbf{x}$, which denotes a representation of the cover signal (e.g., a sequence of transform

CHAPTER 4. PROPOSED METHODOLOGIES

(a) LENA

(b) BABOON

(c) PEPPERS

Figure 4.1: Test images in our experiments. All images are grayscale, of size 512x512 pixels.

CHAPTER 4. PROPOSED METHODOLOGIES

Figure 4.2: DCT coefficient difference of LENA after JPEG 30 compression.

Figure 4.3: DCT coefficient difference of BABOON after JPEG 30 compression.

CHAPTER 4. PROPOSED METHODOLOGIES

Figure 4.4: DCT coefficient difference of PEPPERS after JPEG 30 compression.

Figure 4.5: A framework of the semi-private watermarking technique

CHAPTER 4. PROPOSED METHODOLOGIES

coefficients). The encoder $f(\mathbf{x}_q, T_d)$ embeds the binary watermark bits m based on the prior knowledge about the weighted distortion bound T_d , which is introduced in the previous section. A “linkage sequence” (LS) is generated during the embedding process. The watermark decoder $g(LS, \mathbf{y}_q)$ employs the two-level quantizer again to retrieve the watermark m' from the possible signal-bearing vector $\mathbf{y}$. The LS, which is required for watermark decoding, can be stored in the secret key file.

4.2.1 Watermark embedding

For watermarking embedding, we first apply a Discrete Cosine Transform (DCT) to the original image in order to obtain a transform coefficient matrix with the same size of the original image. The transform coefficients are pseudo-randomly selected to embed the watermark. The watermark used here is a binary signal, which consists of only 1's and 0's.

The embedding algorithm can be summarized by two steps:

1. Amplitude modulation for the selected coefficients;
2. Establishes the “link sequence” between the watermark and the selected coefficients.

CHAPTER 4. PROPOSED METHODOLOGIES

Figure 4.6: Amplitude modulation for transform coefficients

Amplitude modulation

For the first step, amplitude modulation is applied to the selected coefficients as follows:

$$s_i = \begin{cases} x_{qi} \cdot T_d & \text{if } |x_i| \leq T_d \\ x_i & \text{else.} \end{cases} \quad (4.6)$$

where s_i denotes the modified transform coefficient, x_i denotes the selected transform coefficients and x_{qi} is the output of the two-level quantized x_i . As described in Equation 4.6, only the frequency coefficients whose amplitudes are smaller than the distortion bound T_d are modified, and those coefficients with higher amplitude remain intact. As depicted in Figure 4.6, after modulation we obtain the modulated coefficients with absolute values greater than the prescribed threshold and the polarities of these coefficients are maintained.

Establishing linkage sequence

For the second step, a link between the current watermark bit and the polarity of current transform coefficient is established during the embedding process. We call

CHAPTER 4. PROPOSED METHODOLOGIES

Figure 4.7: Watermark embedding of the semi-private algorithm

this a “linkage sequence”, which is produced as follows:

$$LS = \begin{cases} 0 & \text{if } m = 1 \text{ and } x_i \geq 0, \text{ or } m = 0 \text{ and } x_i < 0 \\ 1 & \text{if } m = 0 \text{ and } x_i \geq 0, \text{ or } m = 1 \text{ and } x_i < 0 \end{cases} \quad (4.7)$$

The linkage sequence essentially reflects the relationship between the cover image and the embedded watermark. The sequence is stored in the secret key, which is used during the extraction process. Therefore, this watermarking algorithm is not fully “blind” and is referred as a semi-private watermarking method. The overall embedding process is depicted in Figure 4.7.

For example, if we want to embed a character ‘a’ to the cover image, first we must convert ‘a’ into binary stream by using the ASCII table and a sequence ‘1100001’

CHAPTER 4. PROPOSED METHODOLOGIES

(a) Watermark sequence, ASCII code of "a".

(b) Selected DCT coefficients.

Figure 4.8: The watermark and the selected coefficients in the example.

CHAPTER 4. PROPOSED METHODOLOGIES

(a) Modulated DCT coefficients

(b) Linkage sequence

Figure 4.9: The modulated coefficients and the generated linkage sequence in the example.

Figure 4.10: Watermark extraction process of the semi-private algorithm

is obtained. We pseudo-randomly select seven transform coefficients to embed the seven bits, for example [32.4 -5.8 -25.6 12.5 1.3 -15 7.8]. Figure 4.8(a) and Figure 4.8(b) show the watermark signal and selected transform coefficients, respectively. The modulated coefficients are shown in Figure 4.9(a), and Figure 4.9(b) illustrates the generated linkage sequence.

4.2.2 Watermark extraction

The proposed extraction algorithm, which is illustrated in Figure 4.10, is relatively simpler than the embedding algorithm. The seed of the pseudo-random sequence is first obtained from the key. With this seed, we can find all the coefficient positions that are used during the insertion process. The operation of this algorithm is as follows:

$$m' = \begin{cases} 1 & \text{if } LS = 0 \text{ and } y_i \geq 0, \text{ or } LS = 1 \text{ and } y_i < 0 \\ 0 & \text{if } LS = 1 \text{ and } y_i \geq 0, \text{ or } LS = 0 \text{ and } y_i < 0 \end{cases} \quad (4.8)$$

CHAPTER 4. PROPOSED METHODOLOGIES

Based on this approach, all the embedded watermark bits can be retrieved successfully. To further increase the robustness of the watermark, it is possible to protect the watermark bits by employing error control coding (ECC), such as BCH and Reed-Solomon codes [49].

4.3 Proposed public watermarking algorithm

In this section, we first introduce some common spread-spectrum based watermarking methods in the literature, and then present our novel public watermarking method using the spread-spectrum technique.

4.3.1 A spread-spectrum perspective

Spread-spectrum (SS) is a technique which has been widely used in digital communications because of its high anti-jamming capability [49]. In spread-spectrum communications, a narrow-band signal is transmitted over a much larger bandwidth and thus an attack which is aiming to destroy the transmitted signal must have very high energy. Likewise, in digital watermarking, the watermark can be spread over a wide frequency band prior to embedding in order to achieve higher robustness.

Cox et al. [26] suggested an additive spread-spectrum (SS) based embedding method, in which a noise-like watermark sequence was added to the frequency domain of the cover signal. In [46], Hartung and Girod proposed to spread the watermark bits

CHAPTER 4. PROPOSED METHODOLOGIES

Figure 4.11: A general traditional spread-spectrum (TSS) watermarking framework by using a chip-rate, and then the spread sequence was modulated by a binary pseudo-noise sequence. For detection, the watermarked signal was first filtered by the pseudo-noise sequence, and the sign detector was used to extract the watermark bits. Other similar watermarking techniques can be found in [50, 51]. We call the above-mentioned methods traditional spread-spectrum (TSS) watermarking approaches.

A general TSS watermarking framework is shown in Figure 4.11. A pseudo-random number (PRN) generator is used to produce a sequence $\mathbf{u}$ with zero mean and whose elements are equal to $+\sigma_u$ or $-\sigma_u$. Prior to embedding, the watermark bits m are expanded by using the spreading sequence $\mathbf{u}$ and then the expanded sequence is added to the cover data $\mathbf{x}$. The following equation describes the embedding process.

$$\mathbf{s} = \mathbf{x} + m\mathbf{u} \quad (4.9)$$

CHAPTER 4. PROPOSED METHODOLOGIES

The watermarked sequence $\mathbf{s}$ goes through a distortion channel. If we only consider additive noise:

$$\mathbf{y} = \mathbf{s} + \mathbf{n} \quad (4.10)$$

A correlation detector can be employed to decode the embedded watermark as follows:

$$\begin{aligned} r &= \frac{\langle \mathbf{y}, \mathbf{u} \rangle}{\langle \mathbf{u}, \mathbf{u} \rangle} \\ &= \frac{\langle m\mathbf{u} + \mathbf{x} + \mathbf{n}, \mathbf{u} \rangle}{\sigma_u^2} \\ &= m + x + n \end{aligned} \quad (4.11)$$

where $x = \frac{\langle \mathbf{x}, \mathbf{u} \rangle}{\|\mathbf{u}\|}$ and $n = \frac{\langle \mathbf{n}, \mathbf{u} \rangle}{\|\mathbf{u}\|}$. $\langle \cdot \rangle$ denotes the inner product, which is defined as follows.

$$\langle \mathbf{a}, \mathbf{b} \rangle = \frac{1}{N} \sum_{i=0}^{N-1} a_i b_i \quad (4.12)$$

where $\mathbf{a}, \mathbf{b}$ are vectors with length of N , and a_i, b_i are elements of $\mathbf{a}, \mathbf{b}$, respectively.

The embedded bit can be estimated by judging the sign of r :

$$m' = \text{sign}(r) \quad (4.13)$$

From Equation 4.11 and Equation 4.13, we note that the detection performance of these traditional SS based methods is affected by x , i.e., the interference of the cover signal, which possibly results in detection errors even without any distortion at all.

Malvar and Florencio [44] suggested a modified embedding formula,

$$\mathbf{s} = \mathbf{x} + \mu(x, b)\mathbf{u}. \quad (4.14)$$

CHAPTER 4. PROPOSED METHODOLOGIES

where $x \triangleq \frac{\langle \mathbf{x}, \mathbf{u} \rangle}{\|\mathbf{u}\|}$, b denotes the embedded watermark bits, and $\mu(x, b)$ is a function used to vary the amplitude of the inserted chip sequence $\mathbf{u}$.

If we restrict $\mu(x, b)$ to be a linear function, we have

$$\mathbf{s} = \mathbf{x} + (\alpha b - \lambda x)\mathbf{u}. \quad (4.15)$$

where α and λ control the distortion level and the removal of the interference of the cover signal, respectively.

With the same channel noise model as before, the receiver sufficient statistic becomes

$$\begin{aligned} r &= \frac{\langle \mathbf{y}, \mathbf{u} \rangle}{\|\mathbf{u}\|} \\ &= \alpha b + (1 - \lambda)x + n \end{aligned} \quad (4.16)$$

Thus, λ is used to control the influence of x . This method is called improved spread-spectrum (ISS) modulation technique, which is used to reduce the interference from the cover signal. In ISS, the prior knowledge x , i.e., the projection of $\mathbf{x}$ on the watermark is employed to modulate the energy of the inserted watermark so that the signal inference can be compensated. According to the authors' report [44], ISS was proved to achieve a significant performance improvement over TSS methods.

4.3.2 Proposed spread-spectrum based watermarking approach

In this section, we introduce our proposed SS based public approach for robust watermarking, which essentially embeds the watermark bits into the polarities of

CHAPTER 4. PROPOSED METHODOLOGIES

Figure 4.12: A framework of the TQSS watermarking technique

the frequency components of the cover signal. The general idea of this method is the same as the previous semi-private watermarking method, however the new method incorporates spread-spectrum in order to make the algorithm public, i.e., neither the cover signal nor linkage sequence or original watermark is required for watermark detection.

The framework of the proposed SS based public watermarking technique is shown in Figure 4.12. Similarly to other SS based methods, the binary watermark bits $m : \{1, -1\}$ are first spread by using a spreading sequence $\mathbf{u} : \{1, -1\}$, which is produced by a pseudo-random number (PRN) generator. After spreading, according to the watermark bit to be 1 or -1, we obtain a corresponding sequence $\mathbf{w}_1$ or $\mathbf{w}_{-1}$

CHAPTER 4. PROPOSED METHODOLOGIES

as follows:

$$\mathbf{w}_m = m\mathbf{u} \quad (4.17)$$

The entire watermark embedding process is divided into two passes: polarity modulation and amplitude modulation.

Polarity Modulation

In pass 1, $\mathbf{x}$ first goes through the two-level quantizer $Q(\cdot)$, and then a binary sequence $\mathbf{x}_q$ is generated.

$$\mathbf{x}_q = Q(\mathbf{x}) \quad (4.18)$$

We can always find a $\mathbf{x}_q^*$, which satisfies,

$$\langle \mathbf{w}_m, \mathbf{x}_q^* \rangle \geq T_s \quad (4.19)$$

where T_s denotes an appropriate positive threshold, whose value is between 0 and 1, and $\mathbf{x}_q^*$ denotes the polarity modulated $\mathbf{x}_q$.

To find $\mathbf{x}_q^*$, we define a polarity modulation function, which is used to toggle values in the binary sequence $\mathbf{x}_q$. If the inner product between $\mathbf{w}$ and $\mathbf{x}_q$ is less than T_s , we toggle one bit in $\mathbf{x}_q$, then compute the inner product again, until the inner product value is greater than or equal to the pre-defined threshold T_s .

For example, if $\mathbf{u} = [1, -1, -1, -1]$ and the watermark bit to embed is $m = 1$, thus $\mathbf{w}_1 = \mathbf{u} = [1, -1, -1, -1]$ based on Equation 4.17. Suppose the coefficients we found for embedding are $\mathbf{x} = [55.2, 72.0, -30.5, 50.0]$ and $T_s = 0.5$. After two-level quantization, easily we get $\mathbf{x}_q = [1, 1, -1, 1]$.

CHAPTER 4. PROPOSED METHODOLOGIES

Calculating the inner product of $\mathbf{w}_1$ and $\mathbf{x}_q$, we obtain $\langle \mathbf{w}_1, \mathbf{x}_q \rangle = 0$. To let the inner product value be greater than or equal to T_s , we have to change $\mathbf{x}_q$, that is, the signs of the selected coefficients. If we toggle the second value in x_q , we obtain $\mathbf{x}_q^* = [1, -1, -1, 1]$. Again, we calculate the inner product of $\mathbf{w}_1$ and $\mathbf{x}_q^*$, we obtain $\langle \mathbf{w}_1, \mathbf{x}_q^* \rangle = 0.5 \geq T_s$. Thus, the polarity modulation is finished.

Amplitude Modulation

In the second pass, the amplitude modulation is performed as follows:

$$s_i = \begin{cases} x_{qi}^* \cdot T_d & \text{if } |x_i^*| \leq T_d \\ x_i^* & \text{else.} \end{cases} \quad (4.20)$$

where s_i denotes the modified transform coefficient, x_i^* denotes the polarity modulated coefficients and x_{qi}^* is the output of the two-level quantized x_i^* .

For watermark retrieval, we first locate the watermarked coefficients and perform a two-level quantizer to find the polarity sequence $\mathbf{y}_q$. The pseudo-random sequence $\mathbf{u}$ is reproduced by using the seed from the secret key.

A minimum distance (MD) detector is then performed to decide the watermark bits.

$$m' = \arg \min_{m=1,-1} (\text{dist}(\mathbf{w}_m, \mathbf{y}_q)) \quad (4.21)$$

where m' denotes the extracted watermark bit and $\text{dist}(\cdot)$ denotes the Hamming distance function. The Hamming distance between two sequences is defined to be the number of elements in which they differ [49].

The mechanism of MD detection is shown in Figure 4.13. Figure 4.13(a) shows an example that the extracted watermark bit is -1 , in which $\mathbf{y}_q$ is closer to $\mathbf{w}_{-1}$. If $\mathbf{y}_q$

CHAPTER 4. PROPOSED METHODOLOGIES

(a) The extracted watermark bit is -1 . (b) The extracted watermark bit is 1 .

Figure 4.13: An example of minimum distance detection.

CHAPTER 4. PROPOSED METHODOLOGIES

is closer to $\mathbf{w}_1$, then a watermark bit 1 is retrieved, as depicted in Figure 4.13(b).

In the next chapter, the experimental results of the two proposed watermarking algorithms will be presented.

Chapter 5

Performance Evaluation

In this chapter, the experimental results of the proposed semi-private and public watermarking algorithms are presented based on different categories of attacks. Furthermore, The performance comparison between some common spread-spectrum based watermarking techniques (such as traditional spread-spectrum (TSS) and improved spread-spectrum (ISS)) and our proposed public watermarking algorithm, which is also known as two-level quantization based spread-spectrum (TQSS), is also discussed.

5.1 Experimental results

In our experiments, grayscale test images with size of 512x512 pixels were used. Some sample test images are listed in Figure 4.1. The Discrete Cosine Transform (DCT) was employed and its low/middle frequency regions were selected for water-

CHAPTER 5. PERFORMANCE EVALUATION

mark embedding, in order to achieve good robustness and to minimize the degradation to the watermarked image. Two kinds of watermarks were used for testing the robustness of our algorithms. One was the binary DMT trademark image, of size 32x32 pixels. The other watermark was a 36-character long text string: '0123456789abcdefghijklmnopqrstuvwxyz'. An image watermark can tolerate more error bits, but it requires more bits to embed and therefore the watermarked image would be more distorted. Text watermark requires relatively less bits than an image watermark. However, if one extracted bit was corrupted, the watermark would be incorrect.

Figure 5.1(a) and Figure 5.1(b) show the watermarked images embedded using the semi-private watermarking algorithm with the 36-character long string and the DMT image, respectively. The watermarked images by the public watermarking approach are illustrated in Figure 5.1(c) and Figure 5.1(d). The Peak Signal-to-Noise ratio (PSNR) is an objective criterion commonly used for measuring the perceptual quality of an image, given as follows:

$$PSNR = 10 \log_{10} \frac{N \max_i X_i^2}{\sum_{i=1}^N (X_i - X'_i)^2} (dB) \quad (5.1)$$

where N is the image size, i is the index of each pixel, and X_i and X'_i are the gray levels of the original and the processed pixels.

As shown in Figure 5.1, the PSNR values of the watermarked LENA images are 44.7dB for the semi-private method and 39.3dB for the public method, when the 36-character text watermark is embedded. When embedding the DMT logo, the PSNR values are 39.0dB and 36.0dB for the semi-private and public methods, respectively.

CHAPTER 5. PERFORMANCE EVALUATION

(a) Watermarked image (36 chars), using the semi-private watermarking algorithm, PSNR = 44.7 dB.

(b) Watermarked image (DMT logo), using the semi-private watermarking algorithm, PSNR = 39.0 dB.

(c) Watermarked image (36 chars), using the public watermarking algorithm, PSNR = 39.3 dB.

(d) Watermarked image (DMT logo), using the public watermarking algorithm, PSNR = 36.0 dB.

Figure 5.1: Watermarked LENA image by the semi-private and public watermarking algorithms.

CHAPTER 5. PERFORMANCE EVALUATION

It can be observed that the images watermarked by the semi-private watermarking method have slightly better image quality than those by the public watermarking approach, if embedded with the same watermark. This is because in the latter method, some transform coefficient signs have to be changed due to the polarity modulation process.

We benchmarked our watermarking algorithms with both Stirmark 4.0 [32] and Checkmark 1.2 [33], which are two of the most well-known benchmarking software in the watermarking research community. In the following sections, the experimental results based on different categories of attacks, including non-geometrical attacks, geometrical attacks and combined attacks, will be shown and discussed. For simplicity, we only show the results based on the LENA image, since very similar results were obtained by the algorithms for other images, such as BABOON and PEPPERS.

5.1.1 Non-geometric attacks

Non-geometrical attacks refer to the manipulations which do not change the geometrical characteristics (e.g., size, boundaries and angles.) of an image. Common non-geometrical attacks include lossy compression, additive noise, low-pass filtering, sharpening, thresholding and dithering. Some typical attacked images are illustrated in Figure 5.2 and Figure 5.3. All the above-mentioned attacks can be found in Stirmark and Checkmark.

CHAPTER 5. PERFORMANCE EVALUATION

(a) JPEG compression (QF=20).

(b) Wavelet compression (QF=20).

(c) Additive noise 8%.

(d) Thresholding.

Figure 5.2: Sample resultant images by non-geometrical attacks.

CHAPTER 5. PERFORMANCE EVALUATION

(a) Wiener filtering (5x5).

(b) Denoising and remodulation with Wiener filtering.

(c) Midpiont filtering.

(d) Median filtering.

Figure 5.3: More samples of resultant images by non-geometrical attacks.

CHAPTER 5. PERFORMANCE EVALUATION

Let us consider some of the non-geometrical attacks given as follows.

The Gaussian filter matrix is

$$\begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}$$

The 3x3 spatial sharpening filter matrix is

$$\begin{bmatrix} 0 & -1 & 0 \\ -1 & 5 & -1 \\ 0 & -1 & 0 \end{bmatrix}$$

For additive noise, the attacked image is

$$I'(x, y) = I(x, y) \cdot (1 + \beta \cdot n(x, y))$$

where $I(x, y)$ is the luminance pixel value of an input image at (x, y) , β is a parameter that controls the strength of the additive noise $n(x, y)$, with uniform distribution, zero mean, and unit variance. $I'(x, y)$ denotes the luminance pixel value of the attacked image at (x, y) .

Different objective evaluation methods were used to measure the extracted watermarks. For the trademark “DMT”, a normalized correlation (NC) was used, which is given by

$$NC = \frac{\sum_{x,y} p_{x,y} \tilde{p}_{x,y}}{\sum_{x,y} p_{x,y}^2} \quad (5.2)$$

where $p_{x,y}$ and $\tilde{p}_{x,y}$ denote the cover and watermarked image pixel values at (x, y) , respectively.

CHAPTER 5. PERFORMANCE EVALUATION

Common signal processing attack		Semi-private watermarking			Public watermarking		
		Trademark		Text	Trademark		Text
		Retrieved	NC	Survived Characters	Retrieved	NC	Survived Characters
Median filt.	3x3		1	36		1	36
	5x5		0.74	25		0.93	28
Gaussian filt.			1	36		1	33
Sharpening (3x3)			1	36		1	36
JPEG 70			1	36		1	36
JPEG 50			1	36		1	36
JPEG 40			1	36		1	36
JPEG 30			1	36		1	36
JPEG 15			0.96	34		0.91	11
Add. noise 2%			1	36		0.99	36
Add. noise 5%			0.96	34		0.78	13
Add. noise 8%			0.63	16		0.60	5

Table 5.1: Results with Stirmark. Common signal processing attacks.

CHAPTER 5. PERFORMANCE EVALUATION

Non-geometric attack		Semi-private watermarking			Public watermarking		
		Trademark		Text	Trademark		Text
		Retrieved	NC	Survived Characters	Retrieved	NC	Survived Characters
Wavelet cmp 350			1	36		1	36
Wavelet cmp. 80			1	36		1	36
Wavelet cmp. 50			1	36		0.99	31
Wavelet cmp. 20			0.98	33		0.68	9
Dithering			0.97	33		0.97	30
Hard/Soft thresholding			1	36		1	36
Thresholding			0.93	36		0.61	11
Wiener filt.	1		1	36		1	36
	2		0.99	34		0.96	27

Table 5.2: Results with Checkmark 1.2. Non-geometric attacks.

CHAPTER 5. PERFORMANCE EVALUATION

Non-geometric attack		Semi-private watermarking			Public watermarking		
		Trademark		Text	Trademark		Text
		Retrieved	NC	Survived Characters	Retrieved	NC	Survived Characters
DPR ⁽¹⁾	1		0.79	28		0.96	33
	2		0.51	15		0.92	29
DPRCORR ⁽²⁾	1		0.73	24		0.93	31
	2	Fail	N/A	15		0.83	25
Midpoint filt.	1		0.96	33		1	36
	2	Fail	N/A	1	Fail	N/A	2
Sampling up	1		1	36		1	36
	2		1	36		0.96	35
Trimmed mean filt.	1		1	36		1	36
	2	Fail	N/A	9		1	36

Table 5.3: More results with Checkmark 1.2. Non-geometric attacks. Note: (1)DPR: Denoising and remodulation with Wiener filtering; (2)DPRCORR: Denoising and remodulation assuming a correlated watermark.

CHAPTER 5. PERFORMANCE EVALUATION

For text string watermarks, the number of survived characters out of the 36-character long string is counted.

Table 5.1 shows the benchmarking results of some common signal processing manipulations using Stirmark. In most cases, such signal processing manipulations are not malicious attacks and considered to be part of the system processes. The experimental results of more non-geometrical attacks from Checkmark are depicted in Table 5.2 and Table 5.3.

From the results, it is observed that our algorithms are very robust against non-geometrical attacks. This is because in our algorithms, the watermarks were embedded to low/middle frequency polarities instead of the conventional way of modifying the frequency amplitudes. Although the watermarked image has been slightly distorted, the low and middle frequency distribution, which represents the most basic elements of an image, remained intact. Both the semi-private and public watermarking methods could survive up to 90% of the non-geometric attacks from Checkmark. Our semi-private algorithm was relatively more robust than the public algorithm.

5.1.2 Geometrical attacks

Our scheme was able to survive some geometrical attacks, such as line removal, rescaling and affine transformation. Some sample attacked images are shown in Figure 5.4. In this thesis, we assume that the watermark detector can detect the change of the image size, and the image size can be restored prior to watermark

CHAPTER 5. PERFORMANCE EVALUATION

(a) Scaling down 35%.

(b) Line removal 10 (remove 1 line in each 10 lines).

(c) Affine transformation 2 in Stir-mark4.0

(d) Affine transformation 5 in Stir-mark4.0

Figure 5.4: Sample resultant images by geometrical attacks.

CHAPTER 5. PERFORMANCE EVALUATION

Geometric attack		Semi-private watermarking			Public watermarking		
		Trademark		Text	Trademark		Text
		Retrieved	NC	Survived Characters	Retrieved	NC	Survived Characters
Line removal 10			1	36		1	36
Scaling down 75%			1	36		1	36
Scaling down 50%			1	36		1	33
Scaling down 35%			0.60	21		0.98	12
Affine Transform	1		1	36		0.96	35
	2	Fail	N/A	6	Fail	N/A	N/A
	3		0.49	20	Fail	N/A	N/A
	4	Fail	N/A	5	Fail	N/A	N/A
	5		0.67	29	Fail	N/A	N/A
	6		0.44	9	Fail	N/A	N/A
	7		0.62	25	Fail	N/A	N/A
	8		0.71	26	Fail	N/A	N/A

Table 5.4: Results with StirMark. Geometric attacks.

CHAPTER 5. PERFORMANCE EVALUATION

decoding. Although some information will be lost during the interpolation process, the successful extraction of the watermark was due to the fact that the frequency distribution of the whole image still remained intact.

The semi-private watermarking method could survive the attack of the line removal 10 (removes one line in each ten lines) and the scaling-down attack with a ratio as low as 35%. It could also survive the six of the eight affine transformation attacks from Stirmark 4.0. Like the semi-private method, the public watermarking algorithm was able to survive the line removal and scaling-down attacks. However, it was more sensitive to affine transformation attacks. The experimental results of geometrical attacks are shown in Table 5.4.

Both schemes were vulnerable to rotation and translation attacks. The immunity to such attacks, which destroy the frequency distribution of the image, might be achieved by incorporating advanced restoration techniques which are capable of synchronizing the embedded watermark under these attacks. For example, we could detect some salient points in the original image prior to embedding the watermark, and deploy them as reference points during the detection process.

5.1.3 Combined attacks

In practice, several attacks could be combined to destroy the watermark. We also tested our algorithms against some combined attacks. These attacks included low quality JPEG compression combined with median filtering, sharpening, additive

CHAPTER 5. PERFORMANCE EVALUATION

(a) JPEG compression with quality factor of 30 plus scaling down by 35%.

(b) JPEG compression with quality factor of 30 plus median filtering (3x3).

(c) JPEG compression with quality factor of 30 plus additive noise of 5%.

(d) Print-and-scan process.

Figure 5.5: Sample resultant images by combined attacks.

CHAPTER 5. PERFORMANCE EVALUATION

Combined attack	Semi-private watermarking			Public watermarking		
	Trademark		Text	Trademark		Text
	Retrieved	NC	Survived Characters	Retrieved	NC	Survived Characters
JPEG30 + Median filt. (3x3)		1	36		1	34
JPEG30 + Median filt. (5x5)		0.72	22		0.88	15
JPEG30+Sharpening		1	36		1	36
JPEG30+ Add. noise 5%		0.49	13		0.68	4
JPEG30 + Scaling down 50%		0.99	35		1	36
JPEG30 + Scaling down 40%		0.75	22		0.95	29
JPEG30 + Line removal 10		0.98	34		0.99	35
Print-and-scan		0.70	20		0.72	25

Table 5.5: Results of combined attacks.

noise, scaling-down, line removal and print-and-scan. Figure 5.5 shows the LENA images after the combined attacks. The results of our algorithms' performance under these combined attacks are illustrated in Table 5.5. Experimental results show that our algorithms were also immune to these combined attacks.

5.2 Performance comparison of spread-spectrum watermarking algorithms

We compared our proposed public watermarking TQSS algorithm with traditional spread-spectrum (TSS) and improved spread spectrum (ISS) methods.

We fixed the amount of embedded watermark bits and modified the embedding strength to obtain the same perceptual quality of the watermarked images. Figure 5.6, Figure 5.7, Figure 5.8 and Figure 5.9 illustrate the performance comparisons under JPEG compression, Wavelet compression, additive Gaussian white noise (AWGN) and scaling, respectively. Table 5.6 shows the simulation results under more attacks. It can be observed that when the attack-induced noise becomes dominant in the watermarked signal, ISS achieves similar robustness to TSS. TQSS showed a significant improvement over TSS and ISS. This is because in TQSS, the watermark can be always correctly decoded, as long as most of the polarities of the transform coefficients remain unchanged.

In the next chapter, a practical card authentication system based on our semi-private watermarking algorithm along with the experimental results, will be presented.

CHAPTER 5. PERFORMANCE EVALUATION

Figure 5.6: Performance comparison under JPEG compression

CHAPTER 5. PERFORMANCE EVALUATION

Figure 5.7: Performance comparison under wavelet compression

CHAPTER 5. PERFORMANCE EVALUATION

Figure 5.8: Performance comparison under AWGN

CHAPTER 5. PERFORMANCE EVALUATION

Figure 5.9: Performance comparison under scaling down

CHAPTER 5. PERFORMANCE EVALUATION

Attack	TSS	ISS	TQSS
Gaussian filtering	0.03	0	0
Sharpening	0	0.01	0
Median filtering	0.09	0.08	0
Wiener filtering	0.11	0.03	0
Scaling(40%)	0.21	0.16	0
Scaling(35%)	0.25	0.20	0.06
JPEG70+AWGN(10dB)	0.08	0.05	0.03
JPEG70+Median filtering	0.15	0.10	0
JPEG70+Scaling(60%)	0.35	0.30	0.12
Print-and-scan ¹	0.30	0.21	0.15

Table 5.6: Bit error rates under more attacks.

¹The printer and scanner used in our experiments are HP LaserJet 5000GN and CanoScan1220U respectively.

Chapter 6

Applications

Apart from copyright protection, which attracted much research attention, another application of digital watermarking which has recently also received a considerable interest is authentication. In a watermarking authentication system, a sequence of identification or a visual pattern, i.e., a watermark, is imperceptibly embedded into the cover data. The watermarked data generally goes through some distortion channel. The recovered watermark from an inspected data is used for authentication for different purposes, such as localizing the distortions [52, 53], determining the type of attacks performed [54] or verifying the genuineness of the data.

6.1 Introduction to a card authentication system

In this section, we introduce a card authentication system using our proposed semi-private robust watermarking algorithm. These cards could include student cards,

CHAPTER 6. APPLICATIONS

identity cards, passports or driving licenses. Nowadays, with some sophisticated editing tools (e.g., Photoshop 7), and cheap but high quality printing and scanning facilities, it is extremely easy to modify or replace the original photograph, in order to make fake security documents or cards. It is reported that two-thirds of passport frauds are due to photograph substitution [55].

Consider a simple example. Figure 6.1(a) shows a student's matriculation card, on which there are the owner's name and matriculation number. However, it is extremely easy to fake such a card by replacing the owner's photograph, as depicted in Figure 6.1(b). The fake card owner may use the card to access some exclusive resources (for example, library materials which are accessible for members only), while the library assistant is not able to detect it because the photograph has been changed. Therefore, a secure approach to prevent photo-substitution is to provide a digital linkage between the bearer's photograph and his/her personal information [56].

To prevent such photograph substitution frauds, in the proposed card authentication system, we directly establish a linkage between the owner's photograph and his/her personal identification information, which is considered as the watermark, into the original photograph. Thus, the system can verify the card holder's identity by examining whether the owner's ID matches the information retrieved from the photograph on the inspected card.

The cover data in our card authentication system are face photographs. The size of the standard photographs on the security cards, such as student cards, are around

CHAPTER 6. APPLICATIONS

(a) An original matriculation card.

(b) The faked matriculation card.

Figure 6.1: An example of photo-substitution on student cards.

CHAPTER 6. APPLICATIONS

1x1 inch (we use 180x130 pixels photographs), which is much smaller than the cover images used in most image watermarking algorithms proposed in the literature. Face photographs are typically very smooth, which are much easier to introduce degradation during watermark embedding. Furthermore, a multi-bit watermark is hidden into the smooth face photographs of small size, while the photograph integrity must be maintained at the same time.

The general embedding process is shown in Figure 6.2. The Owner's photograph is first scanned into the computer. The scanned photograph will be the cover image for the embedding algorithm. The watermark is a sequence of personal information (a sequence consisting of combination of characters and numbers is used here). The embedding algorithm is based on the polarity of Hadamard transform coefficients, that determines the insertion of the watermark sequence into the cover photograph. The watermarked photograph and a secret key, are then generated. The watermarked photograph is printed on the owner's security document or card. The retrieval and verification process, as depicted in Figure 6.3, is performed on the scanned photograph of an inspected card for authentication.

The main reason that we choose Hadamard transform in our card authentication system is that it is fast for processing and easy for hardware implementation, compared to other commonly used orthogonal transforms like DCT and DFT. An efficient Hadamard transform hardware structure has been reported [57]. Based on the radix-2 algorithm, the structure can be implemented with a shift register and controllable adder/subtractors for high processing throughput. Therefore, it becomes

CHAPTER 6. APPLICATIONS

Figure 6.2: Watermark embedding process

Figure 6.3: Watermark retrieval and verification process

possible for a real-time implementation of the proposed authentication system.

6.2 Print-and-scan process

In the last section, the proposed card authentication system is introduced. One of the main challenges of such systems is to survive the print-and-scan process or digital-to-analogue-to-digital (DAD) conversion. There has been a very limited research performed on the print-and-scan conversion. After a digital image is printed-and-scanned, various distortions could be introduced. Distortion could occur in

CHAPTER 6. APPLICATIONS

Figure 6.4: The Print-and-scan process

both the pixel values and the geometric boundary of the rescanned image [58]. The distortion of pixel values is typically caused by the luminance, contrast, gamma correction, chrominance variations and blurring of adjacent pixels. Distortion of the geometric boundary in a DAD process is commonly caused by rotation, scaling and cropping. These are some of the common distortion effects of commercial printers and scanners. The print-and-scan process is shown in Figure 6.4.

The print-and-scan (PS) process could be modeled and explained by considering two different scenarios. For the first scenario, rotation and cropping are included in the PS process. In this scenario, PS is usually performed by an adversary, thus it is considered as a malicious combined attack. For scenario two, rotation and cropping are not taken into account. To remedy the first scenario, techniques capable of surviving rotation, scaling and translation (RST) must be included. However, the RST attacks still remain generally an unsolved problem in the watermarking research community, although some heuristic techniques had been proposed [4]. For our consideration, the PS is regarded as an inherent system distortion rather than a malicious attack, which basically relates to scenario two.

6.3 Experimental results and discussions

In our experiments, we used a common printer (HP LaserJet 4MV) and scanner (Cano Scan N1220U), and resolutions were set to 170 dpi (dots per inch) and 200 dpi, respectively. Our project was chosen for the College of Engineering, Technology Week 2003 Exhibition at Nanyang Technological University. We tested our authentication system for 105 different face photographs and the success rate was better than 96%. The failures were mainly due to manual misalignment, because our algorithm was not so resistant to rotation and translation attacks. Some examples of the experimental results are listed in Table 6.1.

CHAPTER 6. APPLICATIONS

Original Image 180x134	Watermarked Image 180x134	Print-and-scan Image 214x159	Watermark Embedded	Verified?
AAAAA1111	Yes
BBBBB2222	Yes
CCCCC3333	Yes
DDDDD4444	Yes
EEEEEE5555	Yes

Table 6.1: Sample experimental results of the card authentication system. Note: (1)The image's unit is pixels; (2) The IDs used here are not the real identification sequences.

Chapter 7

Conclusion and Recommendations

7.1 Conclusion

In this thesis, we proposed two robust watermarking schemes: the semi-private watermarking and the public watermarking schemes. Our algorithms were transform-domain-based, and employed two-level quantization for both watermark embedding and extraction. In our methods, the watermark bits were inserted by establishing a relationship with the polarities of the selected transform coefficients, instead of modifying the magnitude of the frequency components. The advantage of embedding the watermark into polarities was the considerable improvement on performance of robustness under a variety of accidental and intentional manipulations. This is because that the polarities of the frequency components are found to be more reliable than magnitude, according to our investigation.

CHAPTER 7. CONCLUSION AND RECOMMENDATIONS

Our first approach was a semi-private watermarking scheme, in which the watermark bits were embedded by establishing a “linkage sequence” (LS) with the polarities of the selected coefficients. At the same time, the amplitudes of the selected coefficients were modulated to a certain threshold to enhance the reliability. The LS was stored in the secret key and required during the watermark extraction process.

The second method was a public watermarking algorithm, also referred to as two-level quantization spread-spectrum watermarking. This method was adapted from the semi-private approach combined with spread-spectrum. In this method, the selected coefficient sequence was modulated by a polarity modulation function to satisfy that the inner product between the spread-spectrum coded watermark sequence and coefficient sequence must be greater than a prescribed threshold. Furthermore, all the transform coefficients were also amplitude modulated as in the semi-private approach. A minimum distance (MD) detector was employed to retrieve the watermark bits.

We tested the two watermarking schemes with the two well-known benchmarking tools, Stirmark and Checkmark. The experimental results under different categories of attacks (e.g., non-geometrical, geometrical and combined attacks) were reported and the corresponding analyses were given. We found that both the semi-private and public watermarking methods were able to survive up to 90% of the non-geometric attacks from Checkmark. They were also robust against a number of geometric attacks from Stirmark, (e.g., line removal, scaling and affine transformation), as well as some combined attacks, such as JPEG compression combined with additive

CHAPTER 7. CONCLUSION AND RECOMMENDATIONS

noise, and print-and-scan process.

A card authentication system was developed using our semi-private watermarking algorithm in the Hadamard transform domain. The system was successful in surviving the print-and-scan process and authenticated with a success rate close to 96% of 105 photo-IDs. The proposed system could be applied to prevent photo-substitution frauds of many security documents and cards, such as student cards, driving licenses and passports.

7.2 Recommendations

In our algorithms, the distortion bound used for amplitude modulation threshold was based on an empirical estimation. A further research direction could be a more accurate calculation of the distortion bound and combined with the requirement of a particular application.

In our spread-spectrum-based public watermarking approach, some further improvements could be achieved. First, the embedding-induced-distortion of this method was slightly more than the semi-private algorithm, as some polarities of the transform coefficients had to be inverted. A better polarity modulation function could be devised to minimize the degradation. Moreover, the length of the spreading vector could be optimized based on the trade-off between robustness, degradation and imperceptibility.

Finally, our card authentication system treated the print-and-scan process as a

CHAPTER 7. CONCLUSION AND RECOMMENDATIONS

“black box” in the overall process. However, the print-and-scan process could be mathematically modeled and the corresponding restoration could be performed prior to the detection process, and thus a better reliability could be achieved.

Bibliography

- [1] F. M. Boland, J.J.K. O Ruanaidh, and W. J. Dowling, “Watermarking digital images for copyright protection,” in *Proc. Int. Conf. Image Processing and Its Applications*, Edinburgh, U.K., 1995, vol. 410.
- [2] A. Z. Tirkel, G. A. Rankin, R. M. van Schyndel, W. J. Ho, N. R. A. Mee, and C. F. Osborne, “Electronic watermark,” in *Digital Image Computing, Technology and Applications (DICTA ’93)*, Macquarie University, Sydney, 1993, pp. 666–673.
- [3] I. J. Cox and M. L. Miller, “The first 50 years of electronic watermarking,” *Journal of Applied Signal Processing*, pp. 126–132, 2002.
- [4] C.-Y. Lin, M. Wu, J. A. Bloom, I. J. Cox, M. L. Miller, and Y. M. Lui, “Rotation, scale, and translation resilient watermarking for images,” *IEEE Trans. Image Processing*, vol. 10, no. 5, pp. 767–782, 2001.
- [5] C.-W. Tang and H.-M. Hang, “A feature-based robust digital image watermark scheme,” *IEEE Trans. Image Processing*, vol. 11, no. 4, pp. 950–959, 2002.

BIBLIOGRAPHY

- [6] I. Cox, J. Kilian, T. Leighton, and T. Shamoon, “Secure spread spectrum watermarking for images, audio and video,” in *Proc. IEEE Int. Conf. Image Processing (ICIP 96)*, Lausanne, Switzerland, September 1996, pp. 666–673.
- [7] S. Katzenbeisser and A. P. F. Petitcolas, *Information Hiding Techniques for Steganography and Digital Watermarking*, Artech House, 2002.
- [8] I. J. Cox, M. L. Miller, and J. A. Bloom, *Digital watermarking*, Morgan Kaufmann Publishers, 2002.
- [9] E. T. Lin and E. J. Delp, “A review of data hiding in digital images,” in *Proceedings of the Image Processing, Image Quality, Image Capture Systems Conference (PICS '99)*, Savannah, Georgia, April 1999, pp. 274–278.
- [10] F. Hartung and M. Kutter, “Multimedia watermarking techniques,” in *Proceedings of the IEEE*, 1999, vol. 87, pp. 1079–1107.
- [11] M. Yeung, B.-L. Yeo, and M. Holliman, “Digital watermarks: Shedding light on the invisible,” *Micro, IEEE*, vol. 18, no. 6, pp. 32–41, 1998.
- [12] S.-J. Lee and S.-H. Jung, “A survey of watermarking techniques applied to multimedia,” in *Proc. IEEE International Symposium on Industrial Electronics*, Pusan, Korea, June 2001, vol. 1, pp. 272–277.
- [13] M. D. Swanson, M. Kobayashi, and A. H. Tewfik, “Multimedia data-embedding and watermarking technologies,” *Proceedings of the IEEE*, vol. 86, no. 6, pp. 1064–1087, 1998.

BIBLIOGRAPHY

- [14] A. Z. Tirkel, R. M. van Schyndel, and C. F. Osborne, "A two-dimensional watermark," in *Digital Image Computing, Technology and Applications (DICTA '93)*, 1993.
- [15] K. Matsui and K. Tanaka, "Video-steganography," in *Proc. IMA Intellectual Property Project*, Jan, 1994, vol. 1, pp. 187–206.
- [16] W. Bender, D. Gruhl, and N. Morimoto, "Techniques for data hiding," in *Proc. SPIE*, 1995, vol. 2420.
- [17] I. Pitas, "A method for signature casting on digital images," in *IEEE International Conference on Image Processing (ICIP'96)*, Lausanne, Switzerland, September 1996, vol. III, pp. 215–218.
- [18] O. Bruyndonckx, J. J Quisquater, and B. Macq, "Spatial method for copyright labeling of digital images," in *Proc. IEEE Workshop Nonlinear Signal and Image Processing*, Halkidiki, Greece, June 1995.
- [19] G. Voyatzis and I. Pitas, "Applications of toral automorphisms in image watermarking," in *IEEE International Conference on Image Processing (ICIP'96)*, Lausanne, Switzerland, September 1996, vol. 2, pp. 237–240.
- [20] B. Chen and G. W. Wornell, "Quantization index modulation: A class of provably good methods for digital watermarking and information embedding," *IEEE Trans. Information Theory*, vol. 47, no. 4, pp. 1423–1443, 2001.

BIBLIOGRAPHY

- [21] A. B. Watson, “DCT quantization matrices visually optimized for individual images,” in *Proc. SPIE Conf. Human Vision, Visual Processing, and Digital Display IV*, February 1993, vol. 1913, pp. 202–216.
- [22] B. Tao Bo and B. Dickinson, “Adaptive watermarking in the DCT domain,” in *International Conf.on Accoustics, Speech, and Signal Processing, ICASSP '97*, April 1997.
- [23] C. I. Podilchuk and W. Zeng, “Image-adaptive watermarking using visual models,” *IEEE Journal on Selected Areas in Communications*, vol. 16, no. 4, pp. 525–539, 1998.
- [24] J. J. K. O Ruanaidh, W. J. Dowling, and F. M. Boland, “Phase watermarking of digital images,” in *Proc. IEEE Int. Conf. Image Processing*, 1996, vol. 3, pp. 239–242.
- [25] V. Solachidis and L. Pitas, “Circularly symmetric watermark embedding in 2-D DFT domain,” *IEEE Trans. Image Processing*, vol. 10, no. 11, pp. 1741–1753, 2001.
- [26] I. Cox, J. Kilian, T. Leighton, and T. Shamoan, “Secure spread spectrum watermarking for multimedia,” *IEEE Transactions on Image Processing*, vol. 6, no. 12, pp. 1673–1687, 1997.
- [27] X.G. Xia, C. G. Boncelet, and G. R. Arce, “A multiresolution watermark for digital images,” in *IEEE International Conference on Image Processing (ICIP'97)*, October 1997, vol. 1, pp. 548–551.

BIBLIOGRAPHY

- [28] D. Kundur and D. Hatzinakos, “A robust digital image watermarking method using wavelet-based fusion,” in *Proc. IEEE Int. Conf. Image Processing*, Santa Barbara, CA, 1997, vol. 1, pp. 544–547.
- [29] C.-T. Hsu and J.-L. Wu, “Multiresolution watermarking for digital images,” *IEEE Trans. Circuits and Systems-II: Analog and Digital Signal Processing*, vol. 45, no. 8, pp. 1097–1101, 1998.
- [30] S. Pereira, S. Voloshynovskiy, and T. Pun, “Optimized wavelet domain watermark embedding strategy using linear programming,” in *SPIE Aerosense: Wavelet applications VII*, Orlando, FL, USA, 2000.
- [31] Y. W. Wang, J. F. Doherty, and R. E. van Dyck, “A wavelet-based watermarking algorithm for ownership verification of digital images,” *IEEE Trans. Image Processing*, vol. 11, pp. 77–88, 2002.
- [32] M.-S. Hsieh, D.-C. Tseng, and Y.-H. Huang, “Hiding digital watermarks using multiresolution wavelet transform,” *IEEE Trans. Industrial Electronics*, vol. 48, no. 5, pp. 875–882, 2001.
- [33] A. T. S. Ho, J. Shen, and S. H. Tan, “Robust digital image-in-image watermarking algorithm using the fast hadamard transform,” in *SPIE’s 47th Annual Meeting on Optical Science and Technology*, Seattle, USA, 2002.
- [34] R. C. Gonzalez and R. E. Woods, *Digital image processing*, Prentice Hall, 2002.
- [35] S. Craver, B.-L. Yeo, and M. Yeung, “Technical trials and legal tribulations,” *Communications of the ACM*, vol. 41, no. 7, 1998.

BIBLIOGRAPHY

- [36] S. Voloshynovskiy, “Attack modeling: Towards a second generation watermarking benchmark,” *Signal Processing Special Issue on Information Theoretic Issues in Digital Watermarking*, vol. 81, no. 6, pp. 1171–1214, 2001.
- [37] F. Deguillaume, G. Csurka, and T. Pun, “Countermeasures for unintentional and intentional video watermarking attacks,” *SPIE’s 12th Annual Symposium, Electronic Imaging 2000: Security and Watermarking of Multimedia Content II of SPIE Proceedings*, vol. 3971, January 2000.
- [38] S. Craver, B.L. Yeo N. Memon, and M. M. Yeung, “Can invisible watermark resolve rightful ownership?,” in *Fifth Conference on Storage and Retrieval for Images and Video Database*, San Jose, CA, USA, February 1997, vol. 3022, pp. 310–321.
- [39] F. A. P. Petitcolas and R. J. Anderson, “Evaluation of copyright marking systems,” in *IEEE Multimedia Systems*, Florence, Italy, June 1999.
- [40] “Sitmark. [online]. available: <http://www.petitcolas.net/fabien/watermarking/stirmark/>,” .
- [41] “Checkmark. [online]. available: <http://watermarking.unige.ch/checkmark/>,” .
- [42] M. Kutter, F. Jordan, and F. Bossen, “Digital signature of color images using amplitude modulation,” in *Proc. Electronic Imaging*, San Jose, CA, 1997.
- [43] Y. W. Wang and J. F. Doherty, “A POCS-based algorithm to attack image watermarks embedding in DFT amplitude,” in *Proc. 43rd IEEE Midwest Symp. On Circuits and Systems*, Lansing, MI, 2000.

BIBLIOGRAPHY

- [44] H. S. Malvar and D. A. F. Florencio, “Improved spread spectrum: A new modulation technique for robust watermarking,” *IEEE Trans. Signal Processing*, vol. 51, no. 4, pp. 898–905, 2003.
- [45] S. Craver, N. Memon, B. L. Yeo, and M. M. Yeung, “Resolving rightful ownerships with invisible watermarking techniques: Limitations, attacks and implications,” *IEEE Journal on Selected Areas in Communications*, vol. 16, no. 4, pp. 573–586, 1998.
- [46] F. Hartung and B. Girod, “Fast public-key watermarking of compressed video,” in *International Conference on Image Processing (ICIP’97)*, Santa Barbara, California, U.S.A., 1997, vol. I, pp. 528–531.
- [47] H.V. Poor, *An Introduction to Signal Detection and Estimation*, New York: Springer-Verlag, 1994.
- [48] J.J. Eggers and B.Girod, “Quantization effects on digital watermarks,” *Signal Processing*, vol. 81, no. 2, pp. 239–263, 2001.
- [49] B. Sklar, *Digital Communications: Fundamentals and Applications*, Prentice-Hall International, Inc., 2002.
- [50] M. George, J.-V. Chouinard, and N. Georganas, “Digital watermarking of images and video using direct sequence spread spectrum techniques,” in *Proceedings of 1999 IEEE Canadian Conference on Electrical and Computer Engineering*, 1999, vol. 1, pp. 116–121.

BIBLIOGRAPHY

- [51] S. Cheng, H. Yu, and Z. Xiong, “Enhanced spread spectrum watermarking of MPEG-2 AAC,” in *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing*, 2002, vol. 4, pp. IV–3728–IV–3731.
- [52] R. B. Wolfgang and E. J. Delp, “Fragile watermarking using the VW2D watermark,” in *Proc. SPIE/IS&T Inter. Conf. Security and Watermarking of multimedia Contents*, 1999, pp. 40–51.
- [53] J. Hu, J. Huang, D. Huang, and Y. Q. Shi, “Image fragile watermarking based on fusion of multi-resolution tamper detection,” *IEE Electronics Letters*, vol. 38, no. 24, pp. 1512–1513, 2002.
- [54] D. Kundur and D. Hatzinakos, “Digital watermarking for telltale tamper proofing and authentication,” *Proceedings of the IEEE (USA)*, vol. 87, no. 7, pp. 1167–1180, 1999.
- [55] J. Mercer, *Authentication News*, 5(9/10), 2001.
- [56] A. T. S. Ho, J. Shen, H. P. Tan, and J. Woon, “Security-printing authentication using digital watermarking,” *Electronic Imaging*, vol. 13, no. 1, 2003.
- [57] M. Ramkumar, *Data Hiding in Multimedia - Theory and Applications*, PhD Thesis, 2000.
- [58] C.-Y. Lin and S.-F. Chang, “Distortion modeling and invariant extraction for digital image print-and-scan process,” in *Intl. Symp. on Multimedia Information Processing*, Taipei, Taiwan, 1999.