Robust image watermarking against geometric attacks

Dong, Qi

2013

Dong, Q. (2013). Robust image watermarking against geometric attacks. Doctoral thesis, Nanyang Technological University, Singapore.

<http://hdl.handle.net/10356/54840>

<https://doi.org/10.32657/10356/54840>

ROBUST IMAGE WATERMARKING AGAINST GEOMETRIC ATTACKS

DONG QI

School of Electrical & Electronic Engineering

A thesis submitted to the Nanyang Technological University
in partial fulfillment of the requirement for the degree of
Doctor of Philosophy

2013

Acknowledgments

I would like to express my most sincere gratitude to my supervisor Associate Professor Foo Say Wei for his conscientious supervision, sound advices, invaluable guidance and constant support to improve my research work and thesis. His thoughtful comments and encouragement are highly appreciated. I would also like to acknowledge Associate Professor Soon Ing Yann who has given me an excellent opportunity to work with him and provided continual support throughout my research. In addition, I would like to extend my appreciation to Nanyang Technological University for the award of research scholarship during my candidature.

I am deeply grateful to my family for their support, encouragement and unfailing blessings all these years. Their endless love, support, encouragement and care are a strong motivation for me.

Last but not least, I would like to thank all my friends and those who have helped me a lot in the past.

Contents

Acknowledgments	i
Summary	v
List of Figures	vii
List of Tables	xi
List of Abbreviations	xiii
List of Symbols	xv
1 Introduction	1
1.1 Motivation	1
1.2 Objectives	4
1.3 Contributions	5
1.4 Organization of the Thesis.....	6
2 Overview of Digital Watermarking	8
2.1 Introduction to Digital Watermarking Technology	8
2.2 Robust Image Watermarking.....	11
2.3 A Typical Digital Watermarking Scheme	13
2.3.1 Watermark Embedding Process.....	13
2.3.2 Watermark Extracting Process	15
2.4 Chapter Summary	16
3 Related Works	18
3.1 Normalization	18
3.2 Mesh Embedding.....	19
3.3 Exhaustive Search	20
3.4 Template Matching.....	21

3.5 Feature Point Extraction	22
3.6 Invariant Domain.....	24
3.7 Chapter Summary	25
4 Normalization-based Image Watermarking Using SVD	27
4.1 Image Normalization	28
4.1.1 Effects of Geometric Attacks	28
4.1.2 Decomposition of Geometric Attacks	31
4.1.3 Procedures of Image Normalization	32
4.2 Brief Introduction	36
4.2.1 Singular Value Decomposition.....	36
4.2.2 Robust Image Watermarking Schemes Using SVD	37
4.3 Watermark Embedding Process.....	39
4.4 Frequency Masking	44
4.5 Watermark Extracting Process	46
4.6 Security Considerations.....	47
4.7 Experimental Results and Discussions	48
4.7.1 Assessment of Visual Quality.....	49
4.7.2 Assessment of Robustness.....	53
4.7.3 Comparison with Other Watermarking Schemes	67
4.7.4 Effects of Changing Watermark Size	70
4.8 Chapter Summary	72
5 Feature-based Robust Watermarking Using Zernike Moments	73
5.1 Feature Point Extraction	73
5.2 Construction of Local Watermarking Regions	77
5.2.1 Invariance of Circular Patches.....	79
5.3 Zernike Moments and Invariant Properties	82
5.3.1 Zernike Moments.....	82
5.3.2 Testing of Invariance properties	84
5.4 Watermark Embedding Process.....	93
5.5 Watermark Extracting Process	96
5.6 False Alarm Analysis	98

5.7 Experimental Results and Discussions	102
5.7.1 Assessment of Visual Quality Degradation	103
5.7.2 Robustness against Image Processing Operations	106
5.7.3 Robustness against Geometric Attacks.....	113
5.7.4 Robustness against Combination Attacks.....	117
5.7.5 Comparison with Other Watermarking Schemes..	120
5.8 Chapter Summary	122
6 A DFT-based Watermark Synchronization Technique Using LPM	123
6.1 Background Theories.....	123
6.1.1 General Properties of DFT	124
6.1.2 Log-polar Mapping.....	125
6.2 Watermark Synchronization Procedures	127
6.2.1 Watermark Embedding Process.....	127
6.2.2 Watermark Extracting Process	131
6.3 Performance Evaluation	134
6.3.1 Assessment of Robustness.....	136
6.3.2 Comparison with Other Watermarking Schemes	140
6.4 Chapter Summary	141
7 Conclusion and Recommendations	142
7.1 Conclusion.....	142
7.2 Recommendations for Further Research	145
Appendices	148
Appendix A	148
Appendix B.....	150
Author's Publications	151
Bibliography	153

Summary

The recent progress in digital multimedia technologies has offered many facilities in transmission, reproduction and manipulation of digital contents. However, this advance in technology has also brought the problems such as copyright protection and ownership verification. One of the proposed solutions to such problems is digital watermarking. In the design of a robust watermarking scheme, the most important requirement is robustness. For image watermarking, improving the robustness against geometric attacks is one of the challenges in the field of digital watermarking. Thus, the work presented in this thesis is concerned with the design of robust image watermarking schemes against geometric attacks as well as image processing operations.

A normalization-based robust image watermarking is proposed and implemented in the thesis. The proposed scheme utilizes image normalization and singular value decomposition (SVD) to embed robust watermarks. The watermark embedding process is performed in a composite transform domain achieved by SVD and discrete cosine transform (DCT). A watermark bit is embedded in the mid-frequency band by imposing a particular relationship between two pseudo-randomly selected DCT coefficients. High robustness and image fidelity can be achieved. Another robust watermarking scheme is also presented in the thesis. The procedures of feature point extraction are first demonstrated. The invariance properties of image feature points and Zernike moments are then investigated and tested. Several circular patches centered on feature points are generated as stable carriers of watermark information. The watermark bits are embedded by quantizing the magnitudes of local Zernike moments. Experimental results show that

the number of correctly extracted watermark bits is still large enough to prove the ownership even when the watermarked images are severely distorted. All effects of geometric attacks can be converted to translations in a log-polar mapping (LPM) representation. Based on this, a watermark synchronization technique is described. The discrete Fourier transform and LPM are applied to original host image to achieve an invariant domain. The watermark is embedded by adaptively modifying LPM magnitude coefficients. This technique is robust against geometric attacks and common image processing operations. The recommendations for future research are also given based on the limitations of proposed watermarking schemes.

List of Figures

2.1: Three common conflicting parameters of a watermarking scheme: watermark capacity, imperceptibility and robustness	10
2.2: Watermark embedding process, as a part of digital watermarking scheme.....	14
2.3: Watermark extracting process, as a part of digital watermarking scheme	16
3.1: (a) Meshes on an image (b) A typical mesh (c) A distorted mesh after geometric distortion (d) Another distorted mesh after geometric distortion	20
3.2: Demonstration of LPM transform in the spatial domain. (a) Original image and its LPM image (b) Rotated image and its LPM image (c) Resized image and its LPM image (d) Sheared image and its LPM image.....	25
4.1: Different types of geometric attacks and distortions. (a) Original image (b) Scaling with 110% (c) Rotating with the degrees of 15° (d) Shearing 10% in x direction (e) Shearing 15% in y direction (f) Shearing 5% and 10% in x and y directions, respectively (g) Scaling with 105% and rotating with 10° (h) Shearing 10% and 15% in x and y directions, scaling with 80% and rotating with 20°	30
4.2: Effects of StirMark attack: the original image is on the left and the attacked image is on the right. By focusing observation on the grid, it is easy to distinguish the geometric distortions. Without the grid, the distortions are not noticeable, but this attack can defeat many watermarking schemes	30
4.3: Demonstration of image normalization procedures (a) Original host images (b) and (c) Various attacked and distorted images after geometric attacks (d) Normalized images with standard size and orientation.....	35
4.4: SVD of Lena image (a)-(h) are the first to the eighth SV eigenimages, respectively	37
4.5: Illustration of a SVD-DCT block.....	40
4.6: Block diagram of watermark embedding process.....	45
4.7: Attacked watermarked image by averaging operations (PSNR=15.6 dB)	48
4.8: Original host images (a) Lena (b) Girls (c) Fruits (d) Goldhill (e) Plane (f) Lake (g) Boat (h) Peppers.....	49

4.9: Watermarked images (a) Lena (b) Girls (c) Fruits (d) Goldhill (e) Plane (f) Lake (g) Boat (h) Peppers.....	50
4.10: Difference images (a) Lena (b) Girls (c) Fruits (d) Goldhill (e) Plane (f) Lake (g) Boat (h) Peppers.....	50
4.11: Histograms of quantization errors in difference images (a) Lena (b) Girls.....	51
4.12: Extracting BER values under JPEG compression (a) Lena (b) Girls (c) Fruits (d) Goldhill (e) Plane (f) Lake (g) Boat (h) Peppers	54
4.13: Extracting BER values under Gaussian noise addition (a) Lena (b) Girls (c) Fruits (d) Goldhill (e) Plane (f) Lake (g) Boat (h) Peppers.....	55
4.14: Extracting BER values under uniform noise addition (a) Lena (b) Girls (c) Fruits (d) Goldhill (e) Plane (f) Lake (g) Boat (h) Peppers	56
4.15: Extracting BER values under StirMark random bending attack (a) Lena (b) Girls (c) Fruits (d) Goldhill (e) Plane (f) Lake (g) Boat (h) Peppers	57
4.16: (a) Average BER vs. JPEG compression (b) Average BER vs. Gaussian noise addition (c) Average BER vs. uniform noise addition (d) Average BER vs. StirMark random bending attack.....	71
5.1: Extracted feature points on original host images (a) Lena (b) Girls (c) Fruits (d) Goldhill (e) Plane (f) Lake (g) Boat (h) Peppers	76
5.2: Circular patches centered on the feature points (a) Lena (b) Girls (c) Fruits (d) Goldhill (e) Plane (f) Lake (g) Boat (h) Peppers	79
5.3: Circular patches on rotated images (a) 3° (b) 10° (c) 20° (d) 30°.....	80
5.4: Circular patches on resized images (a) 80% (b) 90% (c) 110% (d) 120%	81
5.5: Circular patches on sheared images (a) (5%, 1%) (b) (6%, 1%) (c) (1%, 3%) (d) (1%, 6%).....	81
5.6: Reconstructed images using Zernike moments from orders 10 to 30.....	84
5.7: (a) Original Lena image (b) Rotated Lena image through 30°	85
5.8: Variations of magnitudes under JPEG compression and sharpening	88
5.9: Variations of magnitudes under filtering	89

5.10: Variations of magnitudes under noise addition	90
5.11: Variations of magnitudes under scaling and shearing attacks	90
5.12: Variations of magnitudes under StirMark random bending attacks	91
5.13: Illustration of watermark embedding process.....	96
5.14: Illustration of watermark extracting process	98
5.15: False alarm probability curve	100
5.16: The number of matched bits in un-watermarked images (a) JPEG compression (b) Median filtering (c) Rotation (d) Scaling (e) Shearing (f) Original image without any attack.....	101
5.17: Original host images (a) Lena (b) Girls (c) Fruits (d) Goldhill (e) Plane (f) Lake (g) Boat (h) Peppers.....	102
5.18: (a) Watermarked image (b) Quantization errors.....	104
5.19: Histograms for quantization errors (a) Patch 1 (b) Patch 3.....	105
5.20: Circular patches on the watermarked Lena image before operations	106
5.21: Regenerated circular patch on compressed images with different quality factors (a) 5 (b) 10 (c) 15 (d) 20 (e) 25 (f) 30.....	107
5.22: Numbers of correctly extracted bits from each patch on the compressed images with different quality factors (a) 5 (b) 10 (c) 15 (d) 20 (e) 25 (f) 30	109
6.1: (a) Sampling points in the spatial domain (b) Corresponding sampling points in the LPM representation.....	126
6.2: (a) Original images in the spatial domain (b) LPM representations (c) Reconstructed images	127
6.3: (a) Watermark embedding region in LPM magnitude spectrum (b) Watermark embedding region in Cartesian magnitude spectrum.....	129
6.4: LPM magnitude spectrum for Lena image	129
6.5: (a) Original LPM magnitude spectrum (b) Modified LPM magnitude with a local peak.....	131

6.6: A triangular window for local peak detection	132
6.7: (a) Detected local peak in the rotated image (b) Detected local peak in the resized image.....	133
6.8: (a) Detected local peak in the compressed image (JPEG quality factor =15) (b) Dctected local peak in the filtered image (median filter size =6×6).....	133
6.9: Watermarked images (a) $\alpha=1.5$, PSNR=43.2dB SSIM index=0.993 (b) $\alpha=2$, PSNR=36.4dB SSIM index=0.981 (c) $\alpha=2.5$, PSNR=31.7dB SSIM index=0.875 (d) $\alpha=3$, PSNR=23.8dB SSIM index=0.801.....	135
6.10: (a) Original LPM magnitude spectrum (b) Modified LPM magnitudes with two local peaks.....	135
6.11: Extracting BER values under rotation attacks	136
6.12: Extracting BER values under scaling attacks	137
6.13: Extracting BER values under shearing attacks	137
6.14: Extracting BER values under JPEG lossy compression	138

List of Tables

4.1: PSNR (dB) values of watermarked images	52
4.2: SSIM index of watermarked images.....	53
4.3: Extracting BER values under rotation attacks	58
4.4: Extracting BER values under scaling attacks (x direction).....	59
4.5: Extracting BER values under scaling attacks (x and y directions)	60
4.6: Extracting BER values under shearing attacks	61
4.7: Extracting BER values under median filtering	62
4.8: Extracting BER values under common image operations	63
4.9: Extracting BER values under general affine transformations	64
4.10: Extracting BER values under combination attacks.....	65
4.11: NC values under watermarking attacks	66
4.12: Comparison of PSNR (dB) values	68
4.13: Comparison of SSIM index	68
4.14: Comparison of BER values against image processing operations.....	69
4.15: Comparison of BER values against geometric attacks	70
5.1: Zernike moments and magnitudes of two images	86
5.2: Suitable Zernike moments for robust watermarking	93
5.3: PSNR (dB) values under different step sizes.....	103
5.4: SSIM index of watermarked images.....	104
5.5: Numbers of watermarked patches under JPEG compression	110
5.6: Numbers of watermarked patches under median filtering.....	111
5.7: Numbers of watermarked patches under noise addition	112

5.8: Numbers of watermarked patches under common image processing operations	113
5.9: Numbers of watermarked patches under rotation attacks	114
5.10: Numbers of watermarked patches under scaling attacks	115
5.11: Numbers of watermarked patches under shearing attacks	116
5.12: Numbers of watermarked patches under StriMark bending attacks	117
5.13: Numbers of watermarked patches under ombination attacks	118
5.14: Numbers of watermarked patches under combination attacks	119
5.15: Comparison of PSNR (dB) and SSIM index	120
5.16: Comparison of watermarked patches against attacks	121
6.1: Extracting BER values under noise addition	139
6.2: Extracting BER values under median filtering	139
6.3: Comparison of BER values against attacks	140

List of Abbreviations

Abbreviation	Full expression
ANN	Artificial Neural Network
BCH	Bose-Chaudhuri-Hocquenghem
BER	Bit Error Rate
CWT	Complex Wavelet Transform
DCT	Discrete Cosine Transform
DFT	Discrete Fourier Transform
DoG	Difference-of-Gaussian
DVD	Digital Video Disc
DWA	Digital Watermarking Alliance
DWT	Discrete Wavelet Transform
FMT	Fourier-Mellin Transform
GA	Genetic Algorithm
LPM	Log-polar Mapping
LSB	Least Significant Bit
NC	Normalized Correlation
PSNR	Peak Signal-to-Noise Ratio
SIFT	Scale Invariant Feature Transform
SSIM	Structural SIMilarity

SVD	Singular Value Decomposition
SVM	Support Vector Machine

List of Symbols

Symbol	Meaning
α	Scaling parameter
γ	Ratio between two eigenvalues
λ	Eigenvalue of covariance matrix
θ	Angular coordinate
μ_{pq}	Central moments
ρ	Radial coordinate
σ	Feature scale of an image
Φ_i	Set of selected coefficient pairs
φ	Rotation angle
A	Transformation matrix
C_m	Points in Cartesian magnitude spectrum
D	Watermark decoding function
E	Watermark embedding function
$e(x, y)$	Quantization errors in the spatial domain
F	SVD-DCT block matrix
$F(\mu, \theta)$	DFT coefficient for watermark embedding
$F_w(\mu, \theta)$	Modified DFT coefficient
$G(x, y, \sigma)$	Variable-scale Gaussian function
H_{ss}	Hessian matrix

I	Original input image
I_W	Watermarked image
K	Secret key
L	Length of watermark bit stream
$L(x, y, \sigma)$	Scale-space function of an image
L_m	Points in LPM magnitude spectrum
m_{pq}	Geometric moments
M	Magnitude of Zernike moments
N	Number of generated circular patches
P	LPM computation operator
P_{image}	False alarm probability of an un-watermarked image
P_{patch}	False alarm probability of an un-watermarked patch
P_w	Probability of extracting W watermark bits
R	Radius of circular patches
R_{nm}	Zernike radial polynomials
S	Diagonal matrix
t	Translation vector
U	Unitary orthogonal matrix
V	Unitary orthogonal matrix
$V_{nm}(x, y)$	Complex-valued function
W	Watermark signal
W'_i	Extracted watermark bits

Z	Zernike moment vector
Z'	Quantized Zernike moment vector
Z_{nm}	Zernike moments of order n with repetition m

Chapter 1

Introduction

1.1 Motivation

In recent years, digital multimedia technologies and computer networks have shown significant progress. The technologies offer many new advantages when compared with the old analog counterpart. The advantages such as transmission of data, capability to copy or modify digital contents and many other advantages in communication and signal processing applications have made the digital technology superior to the analog technology. The growth of digital multimedia technology has been tremendous on Internet and the distribution of multimedia data is much easier and faster with the great success of Internet.

The advancements in these technologies have also brought about some problems despite those advantages. The great facility in copying and modifying digital contents rapidly and perfectly has resulted in the problems of copyright protection and multimedia security. It has been recognized for quite some time that current copyright laws are inadequate to deal with digital contents. This has led to an interest towards developing a new copyright protection mechanism. Motivated by this, digital watermarking is proposed as an effective solution to these problems in networked and distributed environment [1-5]. It has become an active and important area of research, and

development of digital watermarking schemes is being considered as an essential way to address the challenges faced by the copyright protection, multimedia security and rapid proliferation of digital contents.

Digital watermarking is a process of embedding hidden information called watermark into original host image, audio or video signals [6-10]. The embedded watermark is usually coded in binary format and it is expected to be a secret imperceptible signal. It should not significantly degrade the quality of original host signals. When the watermarked signal is copied, the embedded watermark will also be copied. The watermark remains present as long as the perceptual quality of the watermarked signal remains at an acceptable level. The owner of the original signal can prove the ownership by extracting the embedded watermark from the watermarked signal.

For robust image watermarking, the embedded watermarks must be robust enough to survive various image processing operations and geometric attacks. The watermarks must be readily extracted from watermarked images even if the images are subject to a variety of possible operations or attacks. In the design of robust image watermarking schemes, two important considerations must be taken into account [10, 12]. First, there must be an appropriate trade-off between robustness and imperceptivity. In other words, the highest-possible robustness must be achieved without visible quality degradation [11]. Second, the watermarking schemes must be able to detect the watermark embedding locations and extract the embedded watermark accurately.

Many robust image watermarking schemes have been proposed in the literature. Most of these schemes are able to guarantee high robustness against common image processing

operations such as compression, filtering and noise addition, but they lack robustness against geometric attacks such as rotation, scaling and translation [10-17]. The main damage for geometric attacks is the loss of synchronization of embedded watermark. Thus, watermark extraction fails even if the watermark still exists in the watermarked image [15-17]. Those attacks are very damaging for digital watermarking schemes because they do not introduce the visible quality degradation but make the extracting process very complex and difficult.

Recently, some well-known techniques in computer vision and pattern recognition such as image normalization, feature point extraction and image moments, have been gradually employed for robust image watermarking to combat geometric attacks [18-26]. Image normalization is able to generate a normalized image that is robust against general geometric attacks such as rotation, scaling and translation [18, 19]. Hence the embedded watermark in the normalized image can still be accurately synchronized under a variety of possible geometric attacks or distortions. Feature point extraction is an approach in pattern recognition to extract features of the contents of digital images [20]. In general, regions around the feature points contain higher energy than other regions of the image so these regions are more robust against image distortions and they can be used as stable carriers of watermark information. Hence feature points can be used to determine the image regions for robust watermark embedding. Some image moments have been proved to be invariant against image distortions [21, 23]. Thus the robust watermark can be embedded by modifying the image moments of host image. Based on these facts, two robust image watermarking schemes and one watermark synchronization technique are proposed and implemented in this thesis. The experimental results show that the proposed

watermarking schemes are able to achieve high robustness against geometric attacks as well as image processing operations.

1.2 Objectives

In this thesis, robust image watermarking schemes against geometric attacks are studied and developed. A large number of papers in the literature show that this issue is theoretically challenging and practically important. As mentioned, geometric attacks and distortions are very damaging to robust watermarking schemes. The visual quality of watermarked image is not degraded by geometric attacks but there will be a lot of extracting errors due to the loss of watermark synchronization even if the watermark still exists in the image. Therefore the effects of geometric attacks should be eliminated in order to synchronize and extract the embedded watermark. Some techniques in computer vision such as image normalization or feature point extraction are adopted in robust watermarking schemes and they are effective in combating geometric attacks and synchronizing embedded watermarks.

There are two main objectives for robust image watermarking. The first one is the high robustness against possible attacks and operations. It is also the main focus of this thesis. The second one is the preservation of visual quality of original host image. Hence highest-possible robustness should be achieved without significant visual quality degradation. However, traditional watermarking schemes do not fully utilize the features of some techniques in computer vision [23] and geometric attacks like StirMark random bending attacks or combination attacks can still defeat many robust watermarking

schemes so it is still an open question for discussion. Hence the main objectives of this thesis are:

- to investigate invariance properties and suitability of the techniques like image normalization, feature point extraction and Zernike moments
- to propose novel algorithms for robust image watermarking against geometric attacks as well as image processing operations
- to develop and implement effective watermarking schemes that utilize the features of image normalization, feature point extraction and Zernike moments in order to achieve high robustness, security and image fidelity
- to design a watermark synchronization technique that adaptively embeds watermark bits in a geometric invariant domain

1.3 Contributions

As mentioned, the objective of this thesis is to develop novel robust image watermarking schemes providing high performance in the presence of watermarking attacks, and the major contributions of this thesis are summarized as follows:

- A normalization-based robust image watermarking scheme is proposed and implemented. The proposed scheme adopts image normalization and singular value decomposition (SVD) techniques to embed robust watermarks in a composite transform domain. Normalization procedures are demonstrated and high robustness can be achieved without significant quality degradation.

- The procedures of extracting image feature points are explained. The invariance properties and robustness of extracted feature points and Zernike moments are investigated and tested, respectively. Their suitability for robust watermarking is also determined.
- A novel and effective watermarking scheme using Zernike moments is proposed and implemented. The local image regions that are robust against distortions are identified and constructed. The watermark bits are embedded by quantizing the magnitudes of local Zernike moments followed by false alarm analysis. The quality degradation of original host image is visually transparent. The proposed scheme is able to achieve high robustness against geometric attacks and image processing operations.
- A non-blind watermark synchronization technique is proposed. The technique utilizes discrete Fourier transform (DFT) and log-polar mapping (LPM) to embed watermarks in an invariant domain. The watermark bits are embedded by adaptively changing LPM magnitude coefficients and creating local peaks. The local peaks can be detected even when the watermarked image is geometrically distorted.
- Extensive experimental results and data are presented to evaluate and compare the performance of proposed watermarking schemes.

1.4 Organization of the Thesis

This thesis consists of seven chapters. In Chapter 2, an overview of digital watermarking is given. The basic terms, applications and requirements etc. in the field of digital watermarking are explained in this chapter. In Chapter 3, several commonly used

methods to combat geometric attacks and synchronize watermarks are described. The major advantages and disadvantages are also explained. Chapter 4 presents a normalization-based image watermarking scheme using SVD. High robustness and image fidelity can be achieved. Chapter 5 presents a feature-based robust watermarking scheme using Zernike moments. The watermark bits are embedded by quantizing the magnitudes of local Zernike moments. Experimental results show that this scheme is highly robust against geometric attacks as well as image processing operations. A watermark synchronization technique is proposed in Chapter 6. The technique adaptively embeds watermark bits in a geometric invariant domain. Concluding remarks and recommendations for future work are presented in Chapter 7.

Chapter 2

Overview of Digital Watermarking

This Chapter gives an overview of digital watermarking technology. Section 2.1 introduces the basic terms of digital watermarking such as applications, requirements and attacks. General requirements and schemes for robust image watermarking are described in Section 2.2. The structure of a typical digital watermarking scheme, which includes watermark embedding and extracting processes, is given in Section 2.3.

2.1 Introduction to Digital Watermarking Technology

With the rapid development of digital multimedia technologies and computer networks worldwide, it is relatively easy to make and distribute unauthorized copies of digital files. Copyright protection and multimedia security become important issues. Digital watermarking is proposed as an effective solution to the problems of copyright protection and data authentication in networked and distributed environment. Digital watermarking is a process of embedding hidden and secret information called watermark into original image, audio or video signals. The embedded watermark is usually coded in binary format and it should not significantly degrade the quality of original host signals.

The specific applications of digital watermarking include copyright protection, content authentication, copy prevention, fingerprinting and broadcast monitoring, etc. [28]. In

addition, there are many other applications like card security, error recovery or transaction tracking where digital watermarking could be of potential use. Each application imposes different requirements and requires different types of watermarking schemes or a combination of such schemes. It is clear that a digital watermarking scheme needs to satisfy a number of requirements such as robustness, security, fragility or imperceptibility. The requirements vary with the specific applications. Generally, the robust watermark should be imperceptible, the amount of conveyed information should be as large as possible and it should be robust against content modifications and geometric attacks. In most watermarking applications, the embedded watermark should be secure. The security of embedded watermark is usually related to the ability to protect some secret information or parameters such as secret keys used in embedding or extracting process so that attackers cannot access the watermark information. As a summary, the most relevant requirements for digital watermarking include robustness, imperceptibility, fidelity, security, capacity and fragility.

It is very common that the above requirements are usually conflicting. Thus, the development of a digital watermarking scheme, depending on a particular application, involves several trade-offs between different requirements [37, 46]. As an example, increasing the watermark embedding capacity necessarily yields decreased imperceptibility or decreased robustness of the embedded watermark. The relationship among these requirements is depicted in Figure 2.1. Depending on the particular application, a trade-off has to be reached to balance among the conflicting requirements to suit the specific problem. Hence the “best” watermarking scheme can be determined only if the specific application is identified.

Figure 2.1: Three common conflicting parameters of a digital watermarking scheme: watermark capacity, imperceptibility and robustness

As mentioned, the key requirement of robust watermarking is high robustness against various attacks and distortions. The aim of these attacks is not always to completely remove or destroy the embedded watermark but usually to disable its extraction. Attacks are limited to those not producing excessive perceptual quality degradations. These attacks could also introduce degradation to the performance of the watermarking schemes. A watermarked signal may be altered either intentionally or accidentally. In both cases the watermarking scheme should be able to detect and extract the embedded watermark accurately. The various attacks and distortions for digital watermarking, which may be intentional or unintentional, include additive noise, filtering, compressions, statistical averaging, watermark removal and geometric attacks, etc. Besides the above attacks, there are many other attacks and operations such as ambiguity attacks, cropping and protocol attacks, etc. Some benchmarking software packages like StirMark, CheckMark

or CertiMark [38] can also be applied to test and assess the robustness of digital watermarking schemes and methods.

2.2 Robust Image Watermarking

For image watermarking, the embedded watermarks should be visually transparent, which means the image quality degradation caused by the embedded watermarks should not be visible. The watermarks embedded in original host images can be broadly classified into three types, robust, fragile and semi-fragile watermarks [26-29]. Robust watermarks are used for copyright protection and ownership verification. The embedded watermarks must be robust enough to survive various image processing operations and geometric attacks. The watermarks must remain extractable from watermarked images even if the images are subject to a variety of possible operations or attacks. The copyright owner is able to prove the ownership by extracting the embedded watermarks from watermarked images. Hence high robustness is the key requirement for robust watermarking techniques. Fragile watermarks, on the other hand, are generally used for data authentication and integrity verification [27]. They are made sensitive to any modifications of the watermarked images. Hence, fragile watermarks are readily altered or destroyed even if the modification does not change the image content. As a sensible combination of fragile and robust watermarks, semi-fragile watermarks have been devised for certain applications [28]. Such watermarks are robust against acceptable content-preserving operations such as JPEG compression or filtering, but highly fragile to malicious attacks, such as cut-paste attacks or collage attacks [28, 29].

Many robust image watermarking schemes have been proposed [38-43, 48-60]. These schemes can be grouped into three categories according to the domain used: schemes performed in spatial domain, schemes performed in transform domain [45, 47] and schemes performed in compression domain [25, 28]. The schemes performed in spatial domain directly embed watermarks into pixel locations of original host image. One commonly used spatial domain technique is the Least Significant Bits (LSB) technique [30]. In this technique, the watermark is embedded in the LSB of some randomly selected pixels. This kind of watermarking method is the earliest and simplest watermarking method but it has low robustness against common image processing operations. The schemes performed in transform domain are based on the modifications of frequency coefficients. The discrete cosine transform (DCT), DFT, discrete wavelet transform (DWT), and complex wavelet transform (CWT) [26] are commonly used. Since high-frequency coefficients will be eliminated by lossy compression or filtering, the robust watermarks are normally embedded by modifying the mid-frequency coefficients that contain visual information of the host image. After that, the inverse transform should be applied to obtain the final watermarked image. Since watermarks embedded in the transform domain will be dispersed over the entirety of the spatial image upon inverse transform, this kind of technique is more robust against distortions than the spatial domain technique. Schemes performed in compression domain make use of compressed image signals such as vector quantization to hide the watermarks [33]. To combat geometric attacks and synchronize embedded watermarks, some schemes and methods are devised by modifying image moments or extracting feature points. This kind of methods is able to achieve high robustness against local or global geometric attacks.

A general scheme for image watermarking contains two processes: watermark embedding process and watermark extracting process. The watermark is embedded in the original host image using a secret key at the encoder [29]. Only the owner of the original host image knows the key and it is not possible to extract or remove the embedded watermark from the watermarked image without the knowledge of this key. Then, the watermarked image passes through the transmission channel. The transmission channel usually includes possible attacks and operations, such as lossy compression, geometric attacks, noise addition and signal processing operations, etc. After the watermarked image passes through these possible attacks and operations, the embedded watermark has to be extracted at the decoder by the owner. In the next section, the structure of a typical digital watermarking scheme is explained in detail.

2.3 A Typical Digital Watermarking Scheme

Every digital watermarking scheme consists of at least two different parts: the watermark encoder and the watermark decoder. Each part can be considered as a separate process, described in the following two subsections.

2.3.1 Watermark Embedding Process

A typical watermark embedding process for still images is presented in Figure 2.2. Let I denote an original host image, W denote a watermark, I_W denote the watermarked image after watermark embedding and K denote a secret key (see Figure 2.2). The watermark embedding function E takes its input image I , watermark W and secret key K and it generates a watermarked image I_W . Introduction of the secret key K is necessary for

enhancing the security aspect of the watermarking scheme. Before the embedding process, the original image can be either transformed in the frequency domain or the embedding can be performed in the spatial domain. The domain selection depends on the selected watermarking technique. If the embedding is performed in frequency domain, inverse transform must be applied in order to obtain the final watermarked image. Mathematically expressed, the embedding function for the spatial domain techniques can be represented as follows:

$$E(I, W, K) = I_W \quad (2.1)$$

For the frequency domain technique, the following expression is valid,

$$E(f, W, K) = I_W \quad (2.2)$$

where f represents the vector of frequency coefficients of the transform applied.

Figure 2.2: Watermark embedding process, as a part of digital watermarking scheme

When the watermarked image is obtained and placed on Internet or possibly transmitted over the communication channel, the operations and attacks occur and the attacked image I_A is generated.

2.3.2 Watermark Extracting Process

In Figure 2.3, a typical watermark extracting process for still images is presented. A decoding function D (see Figure 2.3) takes an attacked image I_A whose ownership is to be determined. The image I_A can be a watermarked or an un-watermarked image. In a general case, it can also be an altered image. The watermark decoding function either recovers a watermark, W_e , from the image, or checks the presence of the embedded watermark, W , in a given attacked image I_A . In this procedure, the same secret key, K , is used for symmetric watermarking schemes. For asymmetric schemes, a different key or public key is used in extracting process to enhance watermark security. In this process, the original image, I , can also be included and it depends on the selected watermarking technique.

Mathematically expressed, the extracting procedure for blind extraction (extraction without using the original host image I) can be expressed as follows:

$$D(I_A, K) = W_e \quad (2.3)$$

For non-blind extraction (extraction using the original host image) the following equation holds:

$$D(I_A, I, K) = W_e \quad (2.4)$$

The blind watermark extraction generates its output, a binary sequence, which indicates the presence or absence of the original watermark W . By this, the following can be assumed:

$$D(I_A, W, K) = \begin{cases} 1, & \text{watermark is present} \\ 0, & \text{watermark is absent} \end{cases} \quad (2.5)$$

The extracted watermark can be used to prove the ownership. For robust watermarking, the embedded watermark must be readily located and extracted from watermarked images even if the watermarked images are subject to a variety of possible operations or attacks.

Figure 2.3: Watermark extracting process, as a part of digital watermarking scheme

2.4 Chapter Summary

An overview of digital watermarking is given in this chapter. The main points covered in this chapter are listed in the following.

- Basic terms and background of digital watermarking technology.
- Some important watermarking applications such as copyright protection, data and content authentication, fingerprinting and broadcast monitoring, etc.
- Some requirements for digital watermarking like robustness, imperceptibility, capacity and security, etc. as well as trade-offs between them.

- Common attacks and distortions applied to test the robustness of watermarking schemes.
- General requirements and schemes for robust image watermarking.
- Description of basic structures of a typical watermarking scheme, including watermark embedding and extracting processes.

Chapter 3

Related Works

As explained in the aforementioned chapters, geometric attacks and distortions are very damaging to digital watermarking schemes because they do not introduce visible quality degradation but make the watermark extracting process very complex and difficult. Many watermarking schemes and methods have been proposed and investigated to combat geometric attacks. In this chapter, a brief introduction to several commonly used methods is given. The implementation procedures, major advantages and disadvantages of the methods are also explained in this chapter.

3.1 Image Normalization

Image normalization is a well-known technique in computer vision and pattern recognition [23]. A robust watermarking method was proposed using moment-based image normalization in [24]. In this method, both watermark embedding and extracting processes are performed using a normalized image with a standard image size and orientation. Thus, it is suitable for public watermarking where the original image is not available [25]. The method in [24] was used to embed a 1-bit watermark. Ping Dong et al. proposed a multi-bit public watermarking scheme based on image normalization [25], which aimed to be robust against general geometric attacks. The scheme is different from

the one in [24]. It addressed more general geometric distortions, e.g., scaling attacks, rotation attacks, and shearing attacks in x or y direction. Another approach is presented in [26] where the watermarking space is defined as a normalized space based on the geometric image moments [28]. These image moments are used to transform the image into a form that is independent to rotation, scaling or translation. The watermark is embedded in this normalized space and then the inverse transform is applied to obtain the final watermarked image [28]. During the watermark extracting process, the moments are calculated again and used to eliminate the effects of geometric distortions. Once the image is normalized, the watermark can be located and extracted accurately. The major disadvantages of the watermarking methods based on image normalization include low robustness against local variations such as random bending, warping or cropping attacks because image moments or centroid will be affected [34]. The details of image normalization technique will be covered in the next chapter.

3.2 Mesh Embedding

A watermarking method was proposed in [29] to recover the original image and combat the geometric attacks. This method introduces some preprocessing operations that are directly performed on the attacked image. A regular mesh is created and applied to both original and watermarked images (see Figure 3.1). The effects of geometric attacks can be eliminated by slightly shifting the meshes on the attacked image. Thus, the synchronization errors between original meshes and the corresponding attacked meshes are minimized. The watermark embedding positions can be located and the embedded

watermark bits can be extracted. This method is effective only in the case of minor distortions [28]. If the watermarked image is subject to significant local or global distortions, the method will cause some mismatches between the meshes and the decoder. Hence it is unable to extract the watermark bits accurately. In addition, the requirement for the original host image restricts the application of this approach.

Figure 3.1: (a) Meshes on an image (b) A typical mesh (c) A distorted mesh after geometric distortion (d) Another distorted mesh after geometric distortion

3.3 Exhaustive Search

Exhaustive search is the simplest approach for watermark detection and extraction after geometric attacks [28]. This method simply inverts each possible geometric attack and distortion that might have been applied to the watermarked image and then performs watermark extracting once for each possible distortion parameter. It is clear that this approach is very inefficient and it rapidly becomes unfeasible as the number of possible attacks increases. Furthermore, it tends to largely increase the false alarm probability and this is reported in [30].

3.4 Template Matching

In this class of methods, a periodic template is inserted into the original host image besides the watermark bits. The parameters of general geometric attacks applied to the watermarked image can be determined through detecting the template. After eliminating the effects of geometric attacks, the watermark can be extracted by the watermark decoder. Hence this template can be used as a reference to detect and combat geometric attacks or distortions.

The scheme proposed in [31] implements the above idea. This scheme embeds multiple periodic shapes into the LSBs of the host image. Thus, any geometric distortion applied to the image will be reflected in the embedded shapes. These shapes can be used to determine the geometric distortions. The major disadvantage of this method is its weak robustness against photometric transformations [28]. Pereira and Pun suggested embedding a pseudorandom template in a ring corresponding to the mid-frequency band of the host image spectrum [26]. The template consists of local peaks that are created by increasing the magnitudes of selected frequency coefficients. The template can be easily detected in the watermark extracting process to determine the parameters of geometric attacks. The synchronization is performed by estimating the geometric attacks which allows for matching of the initial template with the locations of detected template. Kutter and Voloshynovskiy et al. proposed a similar method which inserts a periodic pattern in the DFT domain to generate the shapes of local peaks [28, 55]. A more effective approach [32] extends Kutter's scheme through block-based periodical placement of self-

reference watermarks so that the Fourier magnitude spectrum of periodical watermarks is composed of regular peaks distributed all over the watermarked image [33]. One major advantage of this kind of methods is the effectiveness to address de-synchronization of geometric attacks and distortions, but this kind of pattern is easily detectable in the frequency domain. It is then simple for a malicious attack to alter or remove the embedded watermarks. Therefore, the template or pattern must provide sufficient security against the attack of watermark removal.

More recently, a watermarking method against this kind of attack was given in [35]. The method modulates the resynchronization pattern with a secret binary mask in order to prevent the removal of specific peaks in the DFT domain. The binary mask is obtained from a signal-dependent partition which is robust against rotation and scaling [28].

3.5 Feature Point Extraction

This class of watermarking methods achieves recovery from geometric distortion using the features of the image contents. A particular interesting scheme in this category was given in [22, 80] where feature points are used as a content descriptor. The method works as follows. First a set of feature points that are robust against geometric attacks are extracted. These points are often near corners or edges of the image. A Delaunay triangulation is computed around the feature points and the watermark is then embedded into the resulting triangles [36, 63]. Bas et al. adopted Harris detector to extract feature points, then divided the image into a set of disjoint triangles that are around these feature

points, and finally, both watermark embedding and extracting are conducted in these triangles [62]. Since the Harris detector is not robust enough, this method performs poorly for many attacks. In [16], Mexican Hat wavelet filtering is used for feature point extraction, and watermark is embedded in the normalized local regions centered on the feature points. However, the size of local regions remains fixed, so that this scheme is vulnerable to rotation or scaling. Johnson et al. proposed inverting geometric attacks using only a set of feature points extracted from the original host image [28]. So the synchronization process is able to eliminate the effects of geometric attacks by using the correspondences between feature points extracted from the original host image and those from the attacked or distorted image.

The above techniques have two disadvantages. First, it always computes a Delaunay triangulation around the feature points. Therefore, if an attacker can successfully extract these feature points, the presence of a watermark can be easily determined and the watermark may be removed or destroyed [36]. Since some well-known feature point detectors are adopted, most feature points can actually be extracted by attackers. Second, the method is not as robust as expected because feature points are only robust against small-degree rotation or scaling. These methods are considerably ineffective when large-degree rotation occurs.

3.6 Invariant Domain

Another solution is to embed the watermark in a geometric invariant domain. The main idea of watermarking in a transform invariant domain is to perform the watermark embedding and extracting in a domain that is invariant against geometric attacks and distortions. The FMT is incorporated into some watermarking schemes to combat geometric attacks and distortions [71]. A rotation, scaling and translation invariant domain can be obtained using the LPM. In the resulting log-polar map, rotations, scaling and shearing come down to translations and different distorted images can yield the same log-polar map in the invariant domain (see Figure 3.2). In practice, this approach is effective only for simple affine transformations and it is inapplicable as soon as the image undergoes local geometric attacks and distortions [28]. Moreover, problems of approximation due to the discrete nature of the images and the reduction of the embedding domain make the embedded watermark weakly resistant to noise addition, low-pass filtering and lossy compression [28, 37]. In addition, these algorithms are computationally inefficient and relatively hard to implement.

Rather than embedding the watermark in an invariant domain, Solachidis and Pitas proposed creating a self-similar watermark and embedding it into the DFT domain [38]. The watermark is embedded as a reference between original host image and distorted image. Thus, their method is robust against general geometric attacks and it does not affect the DFT magnitudes. Since the watermark is made up of N identical parts, the detection is possible even after a $2k\pi/N$ degree rotation. The self-similar properties of the

embedded watermark also allows for the reduction of the numbers of different frequency sampling steps where the detection should be performed when the watermarked image is cropped, sheared or resized [28].

Figure 3.2: Demonstration of LPM transform in the spatial domain. (a) Original image and its LPM image (b) Rotated image and its LPM image (c) Resized image and its LPM image (d) Sheared image and its LPM image

3.7 Chapter Summary

This chapter describes several commonly used schemes and methods that are employed to combat geometric attacks and distortions. The implementation procedures, major advantages and disadvantages of these methods are also explained in this chapter. The techniques of image normalization, invariant domain and feature point extraction have

shown significant advantages when compared with other techniques. Our proposed watermarking schemes and techniques which adopt these three watermarking techniques are described in detail in the following chapters.

Chapter 4

Normalization-based Image Watermarking Using SVD

In recent years, digital watermarking schemes based on the SVD techniques have been shown to be effective for robust image watermarking [11]. This can be attributed to the facts that:

- 1) SVs of a digital image are stable and invariant. The SVs remain intact when disturbances are added to an image [45, 46].
- 2) SVD preserves both one-way and non-symmetric properties, which are not obtainable using DFT or DWT [45].
- 3) SVs are able to represent intrinsic algebraic properties of a digital image [11].
- 4) SVD can be performed on both square and rectangular matrices.

This chapter proposes and implements a novel and effective normalization-based robust image watermarking scheme that utilizes the desirable features of SVD. The proposed scheme is performed in a composite transform domain achieved by SVD and DCT. For the proposed scheme, the original host image is first normalized to a standard form for subsequent watermark embedding and extracting. However, in the traditional SVD-based watermarking schemes, the watermark bits are directly embedded by modifying the values of SVs. For the proposed scheme, the first SVs of adjacent sub-

image blocks of the normalized image are concatenated to form a SV block. DCT is then carried out on these SV blocks. A watermark bit is embedded in the mid-frequency band of a SVD-DCT block by imposing a particular relationship between two pseudo-randomly selected DCT coefficients. An adaptive frequency mask is calculated to control local watermark embedding strength. Details of the proposed watermarking scheme as well as experimental results and discussions are given in the following sections.

4.1 Image Normalization

Image normalization is a well-known technique in computer vision and pattern recognition. It is also gradually adopted in digital watermarking applications. The normalized image is obtained from a series of geometric transformations that are invariant and robust against any geometric attacks and distortions of the image [85-89]. This will ensure the integrity and synchronization of embedded watermark in the normalized image even when the image is distorted. In the proposed watermarking scheme, the original host image is first transformed to a normalized image with a standard image size and orientation based on which subsequent operations, watermark embedding and extracting, are performed. This section briefly describes geometric attacks as well as normalization parameters and procedures.

4.1.1 Effects of Geometric Attacks

A robust watermarking scheme should not only be robust against image processing operations but also against geometric attacks and distortions. The most commonly used geometric attacks include rotation, translation, scaling and shearing in x and y directions.

These distortions are very popular in image processing and do not necessarily cause any visual quality degradation.

From the view of image watermarking, geometric distortions mainly introduce desynchronization of embedded watermarks between the watermark encoder and the watermark decoder. The watermark data is still embedded in the image, but the decoder is no longer able to detect, locate or extract it [82]. The explanation is that the embedded watermark is also distorted with spatial alterations of the watermarked image [28]. Basically, there are two types of geometric attacks and distortions as described in the following paragraphs.

1) The typical geometric attacks commonly used to distort image are rotation, translation, scaling and shearing etc. These attacks are applied to the whole image and they can be easily represented by mathematical expressions. One basic solution to identify the specific attack is to perform an exhaustive detection considering all possible geometric attacks applied to the watermarked image [39]. In this case the computation cost will increase drastically. For example, if only a composition of rotation and scaling operation from 70% up to 120% of the original image size is performed, the processing cost is roughly multiplied by 3.5×10^3 . The commonly used geometric attacks and distortions like scaling, rotation and shearing etc. are shown in Figure 4.1.

2) The other category of geometric attacks is especially designed to desynchronize the embedded watermark without visual quality degradation. Khun and Petitcolas developed a benchmark called StirMark containing different attacks and operations [83]. One of the first developed attacks of this benchmark is composed of local random geometric attacks

and distortions that are able to defeat many classical watermarking schemes without visible alterations (see Figure 4.2).

Figure 4.1: Different types of geometric attacks and distortions. (a) Original image (b) Scaling with 110% (c) Rotating with the degree of 15° (d) Shearing 10% in x direction (e) Shearing 15% in y direction (f) Shearing 5% and 10% in x and y directions, respectively (g) Scaling with 105% and rotating with 10° (h) Shearing 10% and 15% in x and y directions, scaling with 80% and rotating with 20°

Figure 4.2: Effects of StirMark attack: the original image is on the left and the attacked image is on the right. By focusing observation on the grid, it is easy to distinguish the geometric distortions. Without the grid, the distortions are not noticeable, but this attack can defeat many watermarking schemes

As a result, it can be shown that, without the reference mark or original image, the decoder will extract the bits from the locations that are different from initial embedding locations. Thus, there will be a lot of extraction errors during watermark extracting process. Therefore the objective of image normalization is to eliminate the effects of geometric attacks and detect the embedded watermark accurately.

4.1.2 Decomposition of Geometric Attacks

The general concept of image normalization using image moments, which are the necessary tools for image normalization, is well-known in pattern recognition [18-20]. This section investigates image moments and decomposes general geometric attacks and distortions.

Let $I(x, y)$ denote a digital image of size $M \times N$. Its geometric moments m_{pq} and central moments, μ_{pq} , $p, q = 0, 1, 2 \dots$ are calculated, respectively, as

$$m_{pq} = \sum_{y=0}^{M-1} \sum_{x=0}^{N-1} x^p y^q \cdot I(x, y) \quad (4.1)$$

and

$$\mu_{pq} = \sum_{y=0}^{M-1} \sum_{x=0}^{N-1} (x - \bar{x})^p (y - \bar{y})^q \cdot I(x, y) \quad (4.2)$$

where

$$\bar{x} = \frac{m_{10}}{m_{00}}, \quad \bar{y} = \frac{m_{01}}{m_{00}} \quad (4.3)$$

Let another image, $I'(x, y)$, denote an attacked and distorted image of $I(x, y)$ with transformation parameters A and t , which is defined as

$$I'(x, y) = A \cdot I(x, y) + t \quad (4.4)$$

where A is a transformation matrix and t is a translation vector. The above equation can be further expressed as

$$\bar{X} = \begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} t_x \\ t_y \end{bmatrix} = A \cdot X + t \quad (4.5)$$

where (x, y) is a pixel coordinates of original input image $I(x, y)$, and (x', y') is the corresponding pixel coordinates of the transformed and distorted image $I'(x, y)$. Based on the SVD, the transformation matrix can be decomposed as

$$\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} = \begin{bmatrix} \cos \varphi & -\sin \varphi \\ \sin \varphi & \cos \varphi \end{bmatrix} \begin{bmatrix} \alpha & 0 \\ 0 & \beta \end{bmatrix} \begin{bmatrix} 1 & \gamma \\ 0 & 1 \end{bmatrix} \quad (4.6)$$

which consists of (1) rotation by angle φ ; (2) scaling in x and y directions; (3) shearing in the x direction. Consequently, the objective of normalization is to determine the transformation parameters in Equation (4.6) and eliminate all these effects caused by the geometric attacks and distortions.

4.1.3 Procedures of Image Normalization

The normalization procedures based on image moments are described in this subsection. The normalized image is able to achieve invariance properties against geometric attacks and distortions. The normalization procedures consist of the following steps for a given original input image $I(x, y)$.

- 1) Place the image, $I(x, y)$, at the density center. This step can be achieved by setting the transformation vector $t = \begin{pmatrix} t_x \\ t_y \end{pmatrix}$ with

$$t_x = -\frac{m_{10}}{m_{00}}, \quad t_y = -\frac{m_{01}}{m_{00}} \quad (4.7)$$

where m_{00} , m_{01} and m_{10} are the image moments of $I(x, y)$ as calculated in Equation (4.1). The aim of this step is to achieve translation invariance.

2) The covariance matrix based on the central moments is given by

$$\begin{bmatrix} \mu_{20} & \mu_{11} \\ \mu_{11} & \mu_{02} \end{bmatrix} \quad (4.8)$$

Based on the covariance matrix, the rotation angle φ is calculated as

$$\varphi = \frac{1}{2} \tan^{-1} [2\mu_{11} / (\mu_{20} - \mu_{02})] \quad (4.9)$$

Hence the rotation effects can be eliminated.

3) The two eigenvalues of the covariance matrix are given by

$$\lambda_i = \frac{1}{2} \left[(\mu_{20} + \mu_{02}) \pm \sqrt{4\mu_{11}^2 + (\mu_{20} - \mu_{02})^2} \right] \quad i = 1, 2 \quad (4.10)$$

Based on the eigenvalues, the scaling parameters are calculated as

$$\alpha = (\lambda_1 \lambda_2)^{0.25} / \sqrt{\lambda_1}, \quad \beta = (\lambda_1 \lambda_2)^{0.25} / \sqrt{\lambda_2} \quad (4.11)$$

Hence the scaling effects in x and y directions can be eliminated.

4) By setting $t = 0$, the following equation hold:

$$\mu'_{p,q} = \sum_{i=0}^p \sum_{j=0}^q \binom{p}{i} \binom{q}{j} t_{11}^i \cdot t_{12}^{p-i} \cdot t_{21}^j \cdot t_{22}^{q-j} \cdot \mu_{i+j, p+q-i-j} \quad (4.12)$$

where $\mu'_{p,q}$ is the central moments of the image without translation effects.

From Equation (4.12), the following equation can be derived.

$$\mu'_{30} = \mu_{30} + 3\beta\mu_{21} + 3\beta^2\mu_{12} + \beta^3\mu_{03} \quad (4.13)$$

By setting $\mu'_{30} = 0$, the following equation can be obtained.

$$\mu_{30} + 3\beta\mu_{21} + 3\beta^2\mu_{12} + \beta^3\mu_{03} = 0 \quad (4.14)$$

Hence the value of shearing parameter β can be calculated from Equation (4.14) and shearing effects can be eliminated.

Note that Equation (4.14) can have up to three roots in the case that $\mu_{03} \neq 0$ (which is generally true for most natural images). There are two scenarios: (1) one of the three roots is real and the other two are complex and (2) all three roots are real [25]. In the former case, the real root is chosen; in the latter case, the median value of the three real roots is chosen to ensure the uniqueness of the resulting normalized image. Hence any attacked image can be converted to a normalized form by identifying the geometric transformation parameters. Interested readers can refer to [88, 89] for more details of image normalization. Another image normalization technique is shown in Appendix A.

An experiment is performed to demonstrate these normalization procedures. Several original input images are shown in Figure 4.3 (a), the various attacked and distorted images are shown in Figure 4.3 (b) and (c). After applying the above normalization procedures, all these images yield the same corresponding normalized images with the standard image size and orientation that are shown in Figure 4.3 (d). Similar experimental results can be obtained if other images are used for the test. Hence image normalization is an effective way to combat geometric attacks and synchronize the embedded watermark.

Figure 4.3: Demonstration of image normalization procedures (a) Original host images (b) and (c) Various attacked and distorted images after geometric attacks (d) Normalized images with standard size and orientation

From this section, it is known that geometric attacks and distortions can cause de-synchronization of embedded watermark. Hence there will be a lot of errors during watermark extracting process with the loss of synchronization. Image normalization is able to transform both original image and its different attacked images to a normalized image with a standard image size and orientation. Hence the embedded watermark in the normalized image is invariant and robust against general geometric attacks such as rotation, scaling, translation and shearing etc. This section describes the general form of geometric attacks, normalization parameters and procedures. The following sections

present the proposed robust image watermarking scheme using SVD as well as experimental results and discussions.

4.2 Brief Introduction

4.2.1 Singular Value Decomposition

SVD is an effective numerical analysis tool which is used to analyze and diagonalize matrices. SVD was first explored for digital watermarking applications in 2003 [11].

A digital image can be viewed as a non-negative real matrix. Let I be the matrix of size $N \times N$ representing a digital image and according to SVD, matrix I can be decomposed as follows.

$$I = USV^T = [u_1, u_2, \dots, u_N] \cdot \begin{bmatrix} \lambda_1 & & & \\ & \lambda_2 & & \\ & & \ddots & \\ & & & \lambda_N \end{bmatrix} \cdot [v_1, v_2, \dots, v_N]^T = \sum_{i=1}^N \lambda_i u_i v_i^T \quad (4.15)$$

where U and V are two $N \times N$ unitary orthogonal matrices that specify the geometric details of the original image and S is an $N \times N$ diagonal matrix. The diagonal elements of S are nonnegative values in a descending order and they are known as SVs, which represent intrinsic algebraic properties of the image [40].

According to Equation (4.15), an image can be interpreted as a summation of N image layers and each layer is termed as an eigenimage. The singular value, λ_i , indicates the energy intensity of the corresponding eigenimage. Figure 4.4 illustrates the 8×8 SVD-based decomposition of Lena image. The results are eight SV eigenimages in component-wise and each is composed of the singular values in the same position of the diagonal

matrix S . For these images, the SVs are linearly normalized to the range $[0, 255]$ and rounded to the nearest integers for displaying purpose.

Figure 4.4: SVD of Lena image (a)-(h) are the first to the eighth SV eigenimages, respectively

In the SVD-based watermarking algorithms, U , S or V can be modified to embed the watermark information, resulting in U' , S' or V' , respectively [40]. The watermarked image is then constructed using

$$I' = U'S'V'^T = \sum_{i=1}^N \lambda'_i u'_i v'^T_i \quad (4.16)$$

4.2.2 Robust Image Watermarking Schemes Using SVD

Most early SVD-based image watermarking algorithms proposed are non-blind, that is, the original host image is required for watermark extraction [47, 91]. A typical example is a pure SVD-based watermarking scheme proposed by Liu and Tan in 2003 [59]. In the scheme, SVD of the original image is performed to obtain the U , S and V matrices, the watermark, W , is then added to the S matrix with a scaling factor ($S' = S + \alpha W$). SVD is

then applied to S' ($S' = U_W S_w V_W^T$). Finally, the watermarked image, I_W , is obtained by multiplying the matrices U , S_w and V^T ($I_W = U S_w V^T$). It is found that the SVD matrices of the reference watermark can be easily detected and discredited so this scheme is not effective for copyright protection. Blind SVD-based watermarking schemes, on the other hand, require only the secret key or some additional parameters in the watermark extraction process [90]. Some methods embed watermark bits directly into SVs of the original image [91-93]. The scheme in [45] adopts quantization method. This approach is robust because it embeds extra data into low-frequency band of original image in a distributed way. In [91], the SVs are calculated from the whole image and the watermark information is embedded by modifying the relationship between adjacent SVs in the low-frequency band. The scheme is blind and robust, but rather limited by its low embedding capacity. In [92], block-based SVD is first performed and then the second SV of each block is modified by embedding the weighted watermark bit. The capacity of this scheme is high but the imperceptivity is not well considered, resulting in visible quality degradation and lightening effect [8]. For another kind of blind watermarking schemes, watermark bits are embedded in the SVs of transformed coefficients [94-97]. In these composite transformation methods, SVs are calculated from mid-band DWT coefficients of original image, and the watermark bits can be embedded by modifying the SVs. High robustness and imperceptivity can be achieved with this approach. However, they are essentially DWT-based methods because DWT is first performed on the original image and SVD is only an auxiliary tool to modify the DWT coefficients. Thus the superior characteristics of SVD are not fully utilized [41]. The third kind of schemes makes use of SVD matrices instead of SVs to embed watermark information. In 2007, Chang et al.

proposed a SVD-based watermarking scheme by modifying the U matrix [47]. The original image is first partitioned into blocks and then SVD is performed on each block. The block that has a larger number of nonzero elements in the S matrix are selected to be embedded with one watermark bit by comparing the relationship between $U(2,1)$ and $U(3,1)$ with a predefined threshold. In [94], both the left and right SVD matrices are modified to enhance robustness and performance. However, for methods in both [94] and [95], the intrinsic orthogonality of the SVD matrices is altered. As a result, in the watermark extracting process, the orthogonality of the SVD matrices obtained from the watermarked image cannot be guaranteed to be identical to that of the original SVD matrices even if no attacks are performed and the visual quality is also degraded. For the method proposed in [96], certain extra processing to preserve the orthogonality is required. In [98], the U matrix is divided into three parts. The first part remains unchanged, the second part is modified by changing its signs and the third part is modified to preserve the orthogonality of the U matrix.

However, schemes based on SVD alone cannot guarantee high robustness against geometric attacks and distortions. Hence, image normalization technique is adopted in the proposed watermarking scheme to combat geometric attacks.

4.3 Watermark Embedding Process

For the proposed watermarking scheme, the original host image is first normalized to a standard form according to the normalization procedures described in Section 4.1. After that, the first SVs of adjacent sub-image blocks of the normalized image are concatenated

to form a SV block. This block is transformed into the frequency domain using DCT. A watermark bit is then embedded in the mid-frequency band of a SVD-DCT block in pseudo-randomly selected positions. The watermark embedding process is repeated over the whole normalized image. In other words, after the last watermark bit has been embedded, the embedding process begins with the first watermark bit again until all the positions that can be embedded are occupied.

The SVD-DCT blocks of normalized image can be obtained according to the following steps.

- 1) Divide the normalized image into non-overlapping 4×4 sub-image blocks.
- 2) Perform SVD on each sub-image block to produce 4 SVs.
- 3) Concatenate the first SVs from adjacent 16 sub-image blocks to form a 4×4 SV block.
- 4) For each SV block, DCT is performed to obtain its corresponding SVD-DCT block.

The result is a 4×4 SVD-DCT coefficients block from 16 adjacent 4×4 sub-image blocks of the normalized image. A SVD-DCT block with 16 DCT coefficients in the zigzag order is shown in Figure 4.5, where the shaded positions are potential positions for watermark embedding.

1	2	6	7
3	5	8	13
4	9	12	14
10	11	15	16

Figure 4.5: Illustration of a SVD-DCT block

The embedding process of one watermark bit in a SVD-DCT block is explained in the following steps.

- 1) Construct an adaptive frequency mask for each SVD-DCT block.
- 2) Use the mask to adaptively adjust local watermark embedding strength.
- 3) Select one pair of coefficients to be modified in the potential positions of the SVD-DCT block.
- 4) Modify the relationship between the two selected coefficients in the selected pair according to the specific watermark bit.

Before watermark embedding, two secret keys are first generated. One is an encryption key [15] which is used to encrypt the original ownership information. After encryption, the original ownership information becomes a pseudo-random bit sequence without any meaning. The other key is a position key [18] which is used to pseudo-randomly select one coefficient pair from the potential positions in the SVD-DCT block. The aim of these secret keys is to reduce the probability for an attacker to detect and alter the embedded watermark bits. These two secret keys must be available to extract embedded watermark bits and recover original ownership information. In this way, the security of proposed watermarking scheme is enhanced.

To achieve an appropriate trade-off between the robustness against a variety of possible attacks and the visual transparency of embedded watermark, the mid-frequency band of SVD-DCT block is chosen for watermark embedding. This is because the DCT block in the proposed scheme is derived from the visually important components (first SVs) of the normalized image. Embedding watermarks in the mid-frequency bands of

SVD-DCT blocks can preserve high visual quality of watermarked images. This differs from the traditional methods that embed watermarks in the low-frequency band of the DCT blocks [93, 95].

The watermark embedding process is implemented by changing the difference of the magnitudes of two pseudo-randomly selected coefficients. Only one pair of coefficients in the seven potential positions in Figure 4.5 is used for watermark embedding. In so doing, an attacker would not know which one pair of coefficients is embedded with the watermark bit, and attacking on all the seven coefficients will cause significant quality degradation to the original normalized image.

In the watermark embedding process, the difference between the magnitudes of two coefficients in a selected pair is first computed as follows.

$$D(x_1, y_1, x_2, y_2) = |F(x_1, y_1)| - |F(x_2, y_2)| \quad (4.17)$$

where F is the SVD-DCT block matrix; (x_1, y_1) and (x_2, y_2) are the coordinates of the two coefficients in the selected pair. It is found that the mid-frequency coefficients in the SVD-DCT block are close enough so that $D(x_1, y_1, x_2, y_2)$ can be considered as a zero-mean random Gaussian process [9]. Let the two modified coefficients be denoted by $F'(x_1, y_1)$ and $F'(x_2, y_2)$. These coefficients are modified to make the difference $D'(x_1, y_1, x_2, y_2)$ positive if the watermark bit to be embedded is “1” and negative if the watermark bit to be embedded is “0”. Details of the watermark embedding process are given in the following paragraphs.

In the case of embedding watermark bit “1” in a selected coefficient pair,

- 1) If the difference, $D(x_1, y_1, x_2, y_2) \geq 0$, no operation is performed.

2) If the difference, $D(x_1, y_1, x_2, y_2) < 0$, the following operations are carried out if one of the coefficients, $F(x_1, y_1)$ and $F(x_2, y_2)$, is nonzero.

$$\begin{cases} F'(x_1, y_1) = \text{sgn}(F(x_1, y_1)) \cdot \left(\frac{|F(x_1, y_1)| + |F(x_2, y_2)|}{2} + 0.5 \cdot \text{mask} \right) \\ F'(x_2, y_2) = \text{sgn}(F(x_2, y_2)) \cdot \left(\frac{|F(x_1, y_1)| + |F(x_2, y_2)|}{2} - 0.5 \cdot \text{mask} \right) \end{cases}$$

3) If $F(x_1, y_1) = F(x_2, y_2) = 0$, then

$$\begin{cases} F'(x_1, y_1) = 0.5 \cdot \text{mask} \\ F'(x_2, y_2) = -0.5 \cdot \text{mask} \end{cases}$$

The parameter *mask* in the above equations is the local frequency mask described in the next section. It is used to adaptively adjust the local watermark embedding strength of each SVD-DCT block.

In the case of embedding watermark bit “0” in a selected coefficient pair,

- 1) If the difference, $D(x_1, y_1, x_2, y_2) < 0$, no operation is performed;
- 2) If the difference, $D(x_1, y_1, x_2, y_2) \geq 0$, the following operations are carried out if one of the coefficients, $F(x_1, y_1)$ and $F(x_2, y_2)$, is nonzero.

$$\begin{cases} F'(x_1, y_1) = \text{sgn}(F(x_1, y_1)) \cdot \left(\frac{|F(x_1, y_1)| + |F(x_2, y_2)|}{2} - 0.5 \cdot \text{mask} \right) \\ F'(x_2, y_2) = \text{sgn}(F(x_2, y_2)) \cdot \left(\frac{|F(x_1, y_1)| + |F(x_2, y_2)|}{2} + 0.5 \cdot \text{mask} \right) \end{cases}$$

3) If $F(x_1, y_1) = F(x_2, y_2) = 0$, then

$$\begin{cases} F'(x_1, y_1) = -0.5 \cdot \text{mask} \\ F'(x_2, y_2) = 0.5 \cdot \text{mask} \end{cases}$$

After watermark embedding, the inverse DCT is first performed on the modified SVD-DCT blocks. The SV blocks consisting of 4×4 first SVs are then constructed

followed by the reconstruction of 16 adjacent 4×4 sub-image blocks of each corresponding SV block. Note that, watermark information is embedded only in the first SVs and other SVs are kept unchanged. Inverse SVD is performed on all blocks and the watermarked normalized image is obtained by concatenation of all these 4×4 sub-image blocks in the spatial domain. The final watermarked image is obtained by restoring to the original image size and orientation before normalization.

4.4 Frequency Masking

The degree of modification of SVD-DCT coefficients is directly proportional to the quality degradation of watermarked image. The modification threshold of each SVD-DCT block, which is block adaptive, can be used to determine the local watermark embedding strength and guarantee the visual transparency of embedded watermark. To achieve the highest-possible robustness without significant quality degradation, it is necessary to calculate a frequency mask to measure the modification threshold of each block. A frequency mask contains a lot of frequency components and several methods were developed to calculate the mask [27, 28]. In the proposed watermarking scheme, the calculation of frequency mask is kept simple to reduce computational load. The parameter, *mask*, as mentioned above is introduced to represent the local frequency mask and it is an average value of the summation of the absolute values of the SVD-DCT coefficients whose indices are from 4 to 10, representing the shaded coefficients in the mid-frequency band as shown in Figure 4.5.

To measure the modification threshold of each block, a linear frequency mask is calculated as follows.

$$mask = \frac{1}{7} \sum_{i=4}^{10} |F(i)| \quad (4.18)$$

where F is the SVD-DCT block matrix. As mentioned above, this frequency mask is able to adjust the local watermark embedding strength of each block. Hence highest-possible robustness can be achieved without significant quality degradation for the proposed scheme.

The complete flow of operations of the watermark embedding process is given in the block diagram form shown in Figure 4.6.

Figure 4.6: Block diagram of watermark embedding process

4.5 Watermark Extracting Process

For the proposed watermarking scheme, watermark extraction is blind, that is, the original host image is not required for watermark extraction. The steps taken to extract the embedded watermark bits are detailed in the following.

- 1) Normalization procedures are applied to the watermarked image which is possibly distorted or attacked. As a result, a normalized watermarked image is obtained. Its size and orientation are identical to those in the watermarking embedding process.
- 2) SVD-DCT operations are carried out on the normalized image to obtain the SVD-DCT blocks.
- 3) Using the position key, the embedding positions of the selected coefficient pair in each SVD-DCT block are identified.
- 4) The difference between two coefficients of each selected coefficient pair is computed as follows:

$$D^*(x_1, y_1, x_2, y_2) = |F^*(x_1, y_1)| - |F^*(x_2, y_2)| \quad (4.19)$$

where $F^*(x_1, y_1)$ and $F^*(x_2, y_2)$ are the two coefficients of a selected pair that might carry watermark information.

- 5) Since the watermark bit stream is repeatedly embedded, accuracy of watermark extraction can be enhanced using “majority vote”. For this purpose, the sum of $D^*(x_1, y_1, x_2, y_2)$ values for all selected coefficient pairs of a given watermark bit repeatedly embedded is first computed as follows.

$$B_i = \sum_{\phi_i} D^*(x_1, y_1, x_2, y_2) \quad (4.20)$$

where Φ_i is the set of selected coefficient pairs for the i^{th} watermark bit in the stream.

6) The extracted watermark bit is determined according to the value of B_i as follows.

$$W'_i = \begin{cases} 1 & \text{if } B_i \geq 0 \\ 0 & \text{if } B_i < 0 \end{cases} \quad (4.21)$$

7) The original ownership information in the binary format is recovered using the encryption key.

4.6 Security Considerations

For the schemes proposed in [43] and [48], the watermark embedding positions are deterministic. For such schemes, a simple attack, such as adding perturbations to the embedding positions or statistical attack will cause significant watermark extraction errors. For the proposed watermarking scheme, $K_1 (= 2)$ DCT coefficients are selected for embedding out of the $K_2 (= 7)$ potential positions. Suppose that the size of embedded watermark is N_w and the maximum capacity for watermark embedding is N_C , then the probability of successfully extracting $K_1/2$ watermark bit without the knowledge of position key is

$$P = \sum_{i=[N_C/(2N_w)]+1}^{[N_C/N_w]} \left(\frac{(K_2-K_1)!}{K_2} \right)^i \left(1 - \frac{(K_2-K_1)!}{K_2} \right)^{[N_C/N_w]-i} \binom{[N_C/N_w]}{i} \quad (4.22)$$

and the probability of successfully extracting all N_w watermark bits is

$$P_w = (P)^{\frac{N_w}{K_1/2}} (1 - P)^{[N_C/N_w]} \quad (4.23)$$

For illustration, suppose that $N_w=32$ and $N_c=256$, then $P \approx 6.8 \times 10^{-10}$ and $P_w \approx 7.3 \times 10^{-18}$. Thus, tremendous computational efforts are required to identify the embedding positions without knowing the position key. On the other hand, if the attackers resort to averaging the magnitudes of the DCT coefficients over all the potential positions for embedding in each SVD-DCT block, the visual quality of the attacked watermarked image will be severely affected. For the original Lena image, the attacked image with low visual quality is shown in Figure 4.7 and Peak Signal-to-Noise Ratio (PSNR) value in this case is only 15.6 dB.

Figure 4.7: Attacked watermarked image by averaging operations (PSNR=15.6 dB)

4.7 Experimental Results and Discussions

In this section, several experiments are carried out to assess the performance of the proposed watermarking scheme. The eight 256×256 gray-level images shown in Figure 4.8 and the images in Appendix B are used as original host images. A 32-bit binary sequence is used as original ownership information. As mentioned above, a secret encryption key is generated to encrypt the original ownership information, after which

the original ownership information becomes a pseudo-random bit sequence which is used as a watermark bit stream and it will be embedded in the original host images.

Figure 4.8: Original host images (a) Lena (b) Girls (c) Fruits (d) Goldhill (e) Plane (f) Lake (g) Boat (h) Peppers

4.7.1 Assessment of Visual Quality

All host images are first normalized, after which the watermark bit stream is embedded into the normalized images according the watermark embedding procedures described in Section 4.3. The final watermarked images of all host images in Figure 4.8 are shown in Figure 4.9.

Figure 4.9: Watermarked images (a) Lena (b) Girls (c) Fruits (d) Goldhill (e) Plane (f) Lake (g) Boat (h) Peppers

Perceptually, the watermarked images shown in Figure 4.9 are visually the same as the original host images and the image quality degradation caused by the embedded watermark is visually transparent.

The difference values between watermarked images and original images are considered as quantization errors. For ease of displaying, the values of quantization errors are normalized to the range of $[-127, 128]$ and rounded to the nearest integers. The resulting images of quantization errors are shown in Figure 4.10.

Figure 4.10: Difference images (a) Lena (b) Girls (c) Fruits (d) Goldhill (e) Plane (f) Lake (g) Boat (h) Peppers

For difference images of Lena and Girls, the numbers of occurrence for the values of quantization errors are plotted in the histograms shown in Figure 4.11. The x-axis shows true values instead of normalized values.

(a)

(b)

Figure 4.11: Histograms of quantization errors in difference images (a) Lena (b) Girls

It can be seen that the values of quantization errors are more or less evenly spread across the range and they are almost uniformly distributed. Thus the image quality degradation is also evenly spread over the entire spatial image instead of being concentrated in a small part. The visual quality of watermarked images can be preserved. In addition, it is difficult for an attacker to detect the existence of embedded watermark using statistical analysis or averaging attacks. Similar experimental results can be obtained for other original host images in Figure 4.8.

For objective measure of visual quality, PSNR is used. Mathematically, PSNR is expressed as follows.

$$PSNR = 10 \log_{10} \frac{MN \max[\max(I(x,y)^2)]}{\sum_{x=1}^M \sum_{y=1}^N [I(x,y) - I'(x,y)]^2} \quad (4.24)$$

where M is the height of the image; N is the width of the image; $I(x,y)$ stands for a pixel value in the original host image and $I'(x,y)$ stands for a pixel value in the watermarked image.

After calculation, the PSNR values of all eight watermarked images are tabulated in Table 4.1.

Table 4.1: PSNR (dB) values of watermarked images

	Lena	Girls	Fruits	Goldhill	Plane	Lake	Boat	Peppers
PSNR	43.3	45.1	45.9	45.2	39.5	46.2	42.4	46.8

The PSNR values of watermarked images are all higher than 39 dB, which means the embedded watermarks are visually transparent and high visual quality can be achieved for watermarked images.

The Structural SIMilarity (SSIM) index is a method for measuring the similarity between two images [52, 53]. In this experiment, SSIM index is applied to further evaluate visual quality. The SSIM index of eight watermarked images is calculated and tabulated in Table 4.2.

Table 4.2: SSIM index of watermarked images

	Lena	Girls	Fruits	Goldhill	Plane	Lake	Boat	Peppers
SSIM	0.978	0.986	0.985	0.989	0.962	0.991	0.972	0.994

It can be seen that all the values of SSIM index are close to 1, which means high similarity between original and watermarked images can be achieved. Similar experimental results can be obtained for the images shown in Appendix B. Hence it can be concluded that degradation of visual quality caused by the embedded watermarks is insignificant and invisible.

4.7.2 Assessment of Robustness

In this subsection, the robustness of proposed watermarking scheme against different types of attacks and operations is tested and evaluated. All the watermarked images in Figure 4.9 are put through the attacks and operations as listed below. Watermark extraction process as described in Section 4.5 is then carried out to extract the watermark bits. The accuracy of watermark extraction is measured by using extracting bit error rate (BER), defined as the ratio between the number of incorrectly extracted bits and the total number of embedded watermark bits.

1) Robustness against JPEG Compression: JPEG lossy compression is a very common operation in digital image processing applications and it is able to remove the redundancy

of the image and reduce the storage size. Hence an image watermarking scheme should be robust against JPEG compression. In this experiment, JPEG compressions with different quality factors are applied to compress the watermarked images. For a given watermarked image, the extracting BER value under specific compression factor is calculated and plotted as shown in Figure 4.12.

Figure 4.12: Extracting BER values under JPEG compression (a) Lena (b) Girls (c) Fruits (d) Goldhill (e) Plane (f) Lake (g) Boat (h) Peppers

From Figure 4.12, it can be seen that extracting BER values dramatically decrease as JPEG compression quality factors increase. When the factor reaches the value of 16, the BER values for all compressed images are 0. Even when the value of factor is only 2, which means the compression severely distort the watermarked images, the BER values

are still around a relatively low value. Hence it can be concluded that the proposed watermarking scheme is very robust against JPEG compression even if the values of quality factors are very low and watermarked images are severely distorted.

2) *Addition of Gaussian noise*: Noise addition is very common and it cannot be avoided or eliminated during signal transmission. In this experiment, Gaussian noises with different signal densities are added to the watermarked images. For a given watermarked image, the extracting BER values under different PSNR values are calculated and plotted as shown in Figure 4.13.

Figure 4.13: Extracting BER values under Gaussian noise addition (a) Lena (b) Girls (c) Fruits (d) Goldhill (e) Plane (f) Lake (g) Boat (h) Peppers

The proposed watermarking scheme embeds the watermark bits in the first eigenimage of the SVD by modifying its high frequency components. So the embedded watermark can survive the attacks of adding Gaussian noises even if the added noise is strong, which is confirmed by the experiments, as shown in Figure 4.13, in which all extracting BER values are very low even if the energy of added noise is high.

3) *Addition of uniform noise*: In this experiment, uniform noises with different signal densities are added to the watermarked images. For a given watermarked image, the extracting BER values under different PSNR values are calculated and plotted as shown in Figure 4.14.

Figure 4.14: Extracting BER values under uniform noise addition (a) Lena (b) Girls (c) Fruits (d) Goldhill (e) Plane (f) Lake (g) Boat (h) Peppers

Similar experimental results and extracting BER values can be obtained for this experiment. Hence it can be concluded that the proposed watermarking scheme is very robust against attacks of noise addition, including Gaussian noise addition and uniform noise addition.

4) *StirMark random bending attack*: StirMark random bending attack is very damaging to robust watermarking schemes. In this experiment, the robustness of proposed scheme against StirMark random bending attack is tested. Different degrees of StirMark random bending attack are applied. For a given watermarked image, the BER values under different degrees are calculated and plotted, as shown in Figure 4.15.

Figure 4.15: Extracting BER values under StirMark random bending attack (a) Lena (b) Girls (c) Fruits (d) Goldhill (e) Plane (f) Lake (g) Boat (h) Peppers

Extracting BER values increase as the degree of StirMark random bending attack increases. When the degree reaches the value of 7, the values of BER are all around 0.35 which is not acceptable for robust watermarking. When the degrees are not very high, BER values are around a relatively low value. The problem caused by high-degree StirMark random bending attack will be addressed in another proposed watermarking scheme that will be described in the following chapters.

5) *Rotation*: All the watermarked images are rotated through different angles. Zero padding operation is first performed to form rectangular images followed by image normalization. The watermark bits are then extracted from different rotated images and for each rotated image, extracting BER values are then calculated and tabulated in Table 4.3.

Table 4.3: Extracting BER values under rotation attacks

		Watermarked images							
		Lena	Girls	Fruits	Goldhill	Plane	Lake	Boat	Peppers
Rotation angles	1°	0	0	0	0	0.01	0	0	0
	2°	0	0	0	0	0	0	0.011	0
	5°	0	0	0	0	0	0	0.006	0
	10°	0	0	0	0	0.008	0	0	0
	20°	0	0	0	0	0	0	0.004	0
	45°	0	0	0	0	0.006	0	0.005	0
	90°	0	0	0	0	0	0	0	0

The watermark bits are first embedded in the normalized images of original host images. Hence the embedded watermarks are invariant against geometric attacks and they can be synchronized accurately during watermark extracting process. This can be proved by the extracting BER values in Table 4.3. It can be seen that almost all BER values are 0. Thus, the proposed scheme is very robust against rotation attacks.

6) *Scaling in x direction*: All the watermarked images are resized in x direction through different scaling parameters. The watermark bits are extracted from resized images and for each resized image, extracting BER values are then calculated and tabulated in Table 4.4.

Table 4.4: Extracting BER values under scaling attacks (x direction)

		Watermarked images							
		Lena	Girls	Fruits	Goldhill	Plane	Lake	Boat	Peppers
Scaling parameters	60%	0.03	0.02	0.008	0.011	0.058	0.035	0.043	0.01
	70%	0.005	0	0	0.004	0.055	0.027	0.021	0.01
	90%	0	0	0	0	0.002	0.004	0.02	0
	110%	0.001	0	0	0	0.009	0	0.019	0
	120%	0.001	0.002	0	0	0.012	0	0.022	0
	140%	0.001	0.01	0	0.003	0.016	0.017	0.015	0
	150%	0.031	0.033	0.004	0.022	0.016	0.016	0.018	0.007

The embedded watermarks in normalized image are also invariant against scaling. Hence all extracting BER values in Table 4.4 are very low even if the watermarked

images are significantly resized. Similar experimental results can be obtained if watermarked images are resized in y direction.

7) *Aspect ratio change*: All the watermarked images are resized in both x and y directions through the following different scaling parameters: (a) (80%, 110%), (b) (90%, 110%), (c) (110%, 90%), (d) (110%, 70%), (e) (120%, 90%), (f) (120%, 110%) and (g) (110%, 110%). The watermark bits are extracted from resized images and for each resized image, extracting BER values are then calculated and tabulated in Table 4.5.

Table 4.5: Extracting BER values under scaling attacks (x and y directions)

		Watermarked images							
		Lena	Girls	Fruits	Goldhill	Plane	Lake	Boat	Peppers
Scaling parameters	(a)	0.028	0.008	0.008	0.012	0.051	0.045	0.044	0
	(b)	0.004	0	0	0.006	0.045	0.037	0.024	0.01
	(c)	0	0.004	0	0	0.007	0.004	0.017	0.008
	(d)	0.002	0	0	0.005	0.009	0	0	0
	(e)	0	0	0.02	0	0.032	0	0.012	0
	(f)	0	0.01	0.011	0.013	0.02	0.017	0.025	0.02
	(g)	0.025	0.018	0.013	0.022	0.038	0.03	0.048	0.017

Even if the watermarked images are resized in x and y directions, all extracting BER values are still very low. They are not very different from the extracting BER values from the images that are resized in x or y direction. Based on above experimental results, it can be concluded that the proposed scheme is very robust against scaling attacks.

8) *Shearing*: Shearing is performed on the watermarked images with the following different shearing parameters: (a) (5%, 1%), (b) (2%, 5%), (c) (1%, 3%), (d) (4%, 3%), (e) (4%, 2%), (f) (3%, 5%) and (g) (5%, 5%), where each pair of numbers indicate the degree of shearing in x and y directions, respectively. After shearing attacks, the extracting BER values are shown in Figure 4.6.

Table 4.6: Extracting BER values under shearing attacks

		Watermarked images							
		Lena	Girls	Fruits	Goldhill	Plane	Lake	Boat	Peppers
Shearing parameters	(a)	0.028	0.008	0.008	0.012	0.051	0.045	0.044	0
	(b)	0.004	0	0	0.006	0.045	0.037	0.024	0.01
	(c)	0	0.004	0	0	0.007	0.004	0.017	0.008
	(d)	0.002	0	0	0.005	0.009	0	0	0
	(e)	0	0	0.02	0	0.032	0	0.012	0
	(f)	0	0.01	0.011	0.013	0.02	0.017	0.025	0.02
	(g)	0.025	0.018	0.013	0.022	0.038	0.03	0.048	0.017

Even if the shearing attack is able to cause significant de-synchronization of embedded watermark, the extracting BER values are still very low. Hence the proposed watermarking scheme is very robust against shearing attacks.

9) *Median filtering*: Median filtering is applied to filter the watermarked images with the following different sizes: (a) 3×3 , (b) 4×4 , (c) 5×5 , (d) 6×6 , (e) 7×7 , (f) 8×8 , and (g) 9×9 . The watermark bits are extracted from filtered images and for each filtered image, extracting BER values are then calculated and tabulated in Table 4.7.

Table 4.7: Extracting BER values under median filtering

		Watermarked images							
		Lena	Girls	Fruits	Goldhill	Plane	Lake	Boat	Peppers
Median filter sizes	(a)	0	0	0	0	0	0	0	0
	(b)	0	0	0	0	0.006	0	0.004	0
	(c)	0.009	0.002	0.002	0	0.052	0.003	0.007	0
	(d)	0.022	0.03	0.011	0.032	0.082	0.03	0.02	0.009
	(e)	0.039	0.041	0.024	0.04	0.105	0.035	0.034	0.034
	(f)	0.155	0.168	0.152	0.142	0.184	0.201	0.155	0.148
	(g)	0.215	0.209	0.197	0.218	0.311	0.217	0.305	0.201

When the sizes of median filters are small, the extracting BER values are very low. As the size increases, the BER values are only slightly increased. Even if the size is 9×9 , which is very damaging to embedded watermark, the extracting BER values are still around a relatively low value. Hence the proposed watermarking scheme is able to guarantee high robustness against median filtering.

10) *Common image processing operations*: Several common image processing operations as follows are applied to the watermarked images to test robustness. (a) Spatial

sharpening with the kernel of $\begin{pmatrix} 0 & -1 & 0 \\ -1 & 5 & -1 \\ 0 & -1 & 0 \end{pmatrix}$, (b) Spatial sharpening with the kernel

of $\begin{pmatrix} 0 & -2 & 0 \\ -2 & 6 & -2 \\ 0 & -2 & 0 \end{pmatrix}$, (c) Gaussian filtering with kernel of $(1/8) \begin{pmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 1 & 1 \end{pmatrix}$, (d)

Gaussian filtering with kernel of $(1/16) \begin{pmatrix} 2 & 2 & 3 \\ 1 & 3 & 5 \\ 3 & 1 & 4 \end{pmatrix}$, (e) Frequency mode Laplacian removal (FMLR) attack, and (f) Color quantization. Color quantization is similar to GIF compression. The watermark bits are extracted from distorted images and for each distorted image, extracting BER values are then calculated and tabulated in Table 4.8.

Table 4.8: Extracting BER values under common image operations

		Watermarked images							
		Lena	Girls	Fruits	Goldhill	Plane	Lake	Boat	Peppers
Common image operations	(a)	0.013	0.02	0.019	0.012	0.013	0.01	0.018	0.01
	(b)	0.018	0.022	0.024	0.02	0.023	0.024	0.022	0.018
	(c)	0.011	0.011	0.01	0.022	0.023	0.025	0.031	0.009
	(d)	0.01	0.017	0.011	0.009	0.027	0.009	0.038	0.012
	(e)	0.023	0.02	0.018	0.021	0.036	0.028	0.039	0.02
	(f)	0.038	0.04	0.024	0.044	0.049	0.039	0.055	0.023

All extracting BER values are very low as shown in Table 4.8. Hence it can be concluded that the proposed scheme is able to achieve high robustness against above image processing operations.

11) *General affine transformations*: Several general affine transformations as follows are applied to transform the watermarked images. (a) transformation matrix $\begin{pmatrix} 1.2 & 0.3 \\ -0.2 & 0.8 \end{pmatrix}$, (b) transformation matrix $\begin{pmatrix} 1.1 & 0.1 \\ -0.3 & 0.9 \end{pmatrix}$, (c) transformation matrix $\begin{pmatrix} 1.0 & 0.5 \\ -0.1 & 0.7 \end{pmatrix}$, (d) transformation matrix $\begin{pmatrix} 1.2 & 0.9 \\ -0.8 & 0.2 \end{pmatrix}$, (e) Horizontal flipping, and (f) Vertical flipping.

The watermark bits are extracted from transformed images and for each transformed image, extracting BER values are then calculated and tabulated in Table 4.9.

Table 4.9: Extracting BER values under general affine transformations

		Watermarked images							
		Lena	Girls	Fruits	Goldhill	Plane	Lake	Boat	Peppers
Affine transformations	(a)	0.03	0.025	0.019	0.018	0.033	0.02	0.041	0.017
	(b)	0.024	0.036	0.026	0.031	0.027	0.024	0.029	0.025
	(c)	0.029	0.034	0.018	0.022	0.021	0.018	0.036	0.012
	(d)	0.018	0.019	0.02	0.029	0.029	0.027	0.03	0.024
	(e)	0	0	0	0	0	0	0	0
	(f)	0	0	0	0	0	0	0	0

The proposed watermarking scheme is very robust against geometric attacks and affine transformations because the watermark is embedded in the normalized image which is also invariant against geometric attacks. This can also be confirmed by the low extracting BER values in Table 4.9.

12) *Combination attacks*: Combination attacks are damaging to robust watermarking. In this experiment, several combination attacks as follows are designed and applied to the watermarked images to test robustness. (a) JPEG 15+Rotation 35°, (b) JPEG 20+Scaling 90%, (c) JPEG 25+Scaling 120%, (d) Median filtering 5×5+Rotation 25° (e) Median filtering 6×6+Scaling 80%, and (f) Median filtering 7×7+Scaling 110%. The watermark

bits are extracted from attacked images and for each attacked image, extracting BER values are then calculated and tabulated in Table 4.10.

Table 4.10: Extracting BER values under combination attacks

		Watermarked images							
		Lena	Girls	Fruits	Goldhill	Plane	Lake	Boat	Peppers
Combination attacks	(a)	0.027	0.026	0.017	0.026	0.031	0.03	0.029	0.021
	(b)	0.01	0.015	0.013	0.02	0.035	0.024	0.033	0.016
	(c)	0.01	0.021	0.025	0.017	0.02	0.034	0.25	0.018
	(d)	0.03	0.028	0.022	0.025	0.022	0.029	0.028	0.025
	(e)	0.034	0.039	0.027	0.033	0.041	0.038	0.04	0.018
	(f)	0.046	0.041	0.038	0.044	0.056	0.039	0.05	0.04

Although combination attacks are very damaging, the proposed watermarking scheme can still guarantee relatively low extracting BER values.

As explained, the proposed scheme is based on image normalization and normalized image is obtained from image moments. The image moments and centroid are sensitive to local variations such as cropping attacks. Hence this kind of attack will cause the loss of watermark synchronization and there will be a lot of extracting errors. Even when the watermarked images are slightly cropped, the extracting BER values are still higher than 0.4. Hence this scheme is not robust against cropping attacks.

Besides BER, normalized correlation (NC) is another way to assess robustness of watermarking schemes. It is used to measure the similarity between original and

extracted watermark bits [57]. In this experiment, several operations and attacks given as follows are used to attack the watermarked images in Figure 4.9, (a) JPEG compression of factor 10, (b) JPEG compression of factor 16, (c) Median filtering of size 5×5 , (d) Median filtering of size 7×7 , (e) StirMark random bending attack of degree 4, (f) StirMark random bending attack of degree 6, (g) Rotation of 5° , (h) Rotation of 10° , (i) Aspect ratio of (80%, 110%), (j) Aspect ratio of (120%, 90%). After that, NC values are calculated and tabulated in Table 4.11.

Table 4.11: NC values under watermarking attacks

		Watermarked images							
		Lena	Girls	Fruits	Goldhill	Plane	Lake	Boat	Peppers
Watermarking attacks	(a)	0.9051	0.9316	0.9662	0.9380	0.9285	0.9049	0.9321	0.9133
	(b)	0.9905	0.9862	0.9977	0.9985	0.9960	0.9860	0.9862	0.9857
	(c)	0.9832	0.9969	0.9970	1.0000	0.8995	0.9951	0.9831	1.0000
	(d)	0.9331	0.9153	0.9446	0.9331	0.8578	0.9349	0.9401	0.9403
	(e)	0.6181	0.6930	0.6452	0.7537	0.5998	0.6459	0.7040	0.6976
	(f)	0.5760	0.5697	0.5761	0.5858	0.4989	0.5234	0.5319	0.5821
	(g)	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	0.9871	1.0000
	(h)	1.0000	1.0000	1.0000	1.0000	0.9862	1.0000	1.0000	1.0000
	(i)	0.9462	0.9869	0.9865	0.9773	0.9045	0.9150	0.9151	1.0000
	(j)	1.0000	1.0000	0.9538	1.0000	0.9398	1.0000	0.9677	1.0000

NC values are in the range [0,1]. High NC value indicates high similarity to original watermark bits as well as high robustness against attacks. It can be seen that all the values

are close to 1 except when high-degree StirMark random bending attacks are applied. These results are in complete agreement with the earlier experiments using BER values.

All the attacks listed above are also applied to the original host images given in Appendix B, similar experimental results and data can be obtained. Hence based on the above extensive experimental results and data, it can be concluded that the proposed robust watermarking scheme is able to preserve the visual quality of original host images and quality degradation caused by the embedded watermark is not visible. In addition, the proposed scheme is able to guarantee very high robustness against geometric attacks as well as image processing operations except cropping attacks or high-degree StirMark random bending attack. This problem will be addressed in the following chapters.

4.7.3 Comparison with Other Watermarking Schemes

To compare the performance and robustness with other SVD-based watermarking schemes, both the proposed scheme and the scheme reported in [58] are implemented. In [58], the watermark bits are directly embedded by quantizing the SVs. This scheme is able to achieve higher robustness than other SVD-based watermarking schemes. The images shown in Figure 4.8 are used as original host images and a 32-bit binary stream is used as watermark message. Different types of operations and attacks, including content-preserving image processing operations and geometric attacks, are applied to the watermarked images. The watermark bits are then extracted and extracting BER values are calculated. The comparison results are given in the following paragraphs.

a) Comparison of visual quality

After watermark embedding, the values of PSNR and SSIM index for each watermarked image are calculated and shown in Table 4.12 and Table 4.13, respectively.

Table 4.12: Comparison of PSNR (dB) values

	Lena	Girls	Fruits	Goldhill	Plane	Lake	Boat	Peppers
Proposed scheme	43.3	45.1	45.9	45.2	39.5	46.2	42.4	46.8
Scheme [58]	42.0	47.3	46.7	47.1	41.8	46.9	43.0	45.5

Table 4.13: Comparison of SSIM index

	Lena	Girls	Fruits	Goldhill	Plane	Lake	Boat	Peppers
Proposed scheme	0.978	0.986	0.990	0.989	0.962	0.991	0.972	0.994
Scheme [58]	0.980	0.989	0.979	0.991	0.977	0.980	0.982	0.980

When two schemes are applied, similar values of PSNR and SSIM index can be achieved for each watermarked image. Hence both schemes can preserve high visual quality of watermarked images.

b) Comparison of robustness against image processing operations

The following common image processing operations are applied to the watermarked images and extracting BER values are calculated to compare the robustness, (a) Gaussian noise addition (PSNR=10 dB), (b) Gaussian noise addition (PSNR=20 dB), (c) JPEG compression of factor 4, (d) JPEG compression of factor 10, (e) Median filtering of size 3×3 , (f) Median filtering of size 7×7 . The comparison results are given in Table 4.14.

Table 4.14: Comparison of BER values against image processing operations

		Watermarked images							
		Lena	Girls	Fruits	Goldhill	Plane	Lake	Boat	Peppers
(a)	Proposed scheme	0.225	0.233	0.221	0.201	0.272	0.238	0.238	0.208
	Scheme [58]	0.277	0.290	0.298	0.310	0.369	0.321	0.293	0.269
(b)	Proposed scheme	0.025	0.033	0.053	0.050	0.112	0.075	0.118	0.110
	Scheme [58]	0.106	0.056	0.045	0.063	0.097	0.108	0.105	0.110
(c)	Proposed scheme	0.158	0.151	0.118	0.113	0.129	0.138	0.139	0.129
	Scheme [58]	0.302	0.343	0.286	0.290	0.383	0.377	0.274	0.263
(d)	Proposed scheme	0.051	0.038	0.018	0.028	0.035	0.070	0.038	0.043
	Scheme [58]	0.113	0.089	0.064	0.069	0.088	0.121	0.080	0.076
(e)	Proposed scheme	0	0	0	0	0	0	0	0
	Scheme [58]	0.011	0	0	0	0.015	0	0.009	0
(f)	Proposed scheme	0.039	0.041	0.024	0.040	0.105	0.035	0.034	0.034
	Scheme [58]	0.129	0.117	0.097	0.122	0.234	0.176	0.201	0.183

It can be seen that almost all extracting BER values of proposed scheme are lower than those of the scheme in [58] especially when watermarked images are significantly compressed or filtered.

c) Comparison of robustness against geometric attacks

The following common geometric attacks are applied to the watermarked images and extracting BER values are calculated to compare the robustness, (a) Rotation of 1° , (b) Rotation of 2° , (c) Aspect ratio of (110%, 90%), (d) Aspect ratio of (120%, 110%), (e) StirMark random bending attack of degree 5, (f) StirMark random bending attack of degree 7. The comparison results are given in Table 4.15.

Table 4.15: Comparison of BER values against geometric attacks

		Watermarked images							
		Lena	Girls	Fruits	Goldhill	Plane	Lake	Boat	Peppers
(a)	Proposed scheme	0	0	0	0	0.010	0	0	0
	Scheme [58]	0.006	0	0	0	0.018	0	0.009	0
(b)	Proposed scheme	0	0	0	0	0	0	0.011	0
	Scheme [58]	0	0.006	0	0	0.007	0.011	0	0
(c)	Proposed scheme	0	0.004	0	0	0.007	0.004	0.017	0.008
	Scheme [58]	0	0.008	0.010	0	0.015	0.007	0.028	0.019
(d)	Proposed scheme	0	0.013	0.011	0.013	0.023	0.017	0.025	0.020
	Scheme [58]	0.011	0.059	0.042	0.066	0.108	0.037	0.113	0.095
(e)	Proposed scheme	0.355	0.350	0.306	0.312	0.483	0.283	0.388	0.380
	Scheme [58]	0.375	0.359	0.401	0.366	0.483	0.410	0.404	0.391
(f)	Proposed scheme	0.407	0.420	0.413	0.388	0.530	0.421	0.425	0.408
	Scheme [58]	0.509	0.498	0.502	0.477	0.552	0.443	0.501	0.470

It can be seen that all extracting BER values of proposed scheme are lower than or equal to those of the scheme in [58]. When high-degree StirMark bending attacks are applied, high extracting BER values are obtained for both schemes so they are not robust against such attacks.

Similar experimental results and data can be obtained for other images shown in Appendix B. From the above experimental results, it can be concluded that the proposed watermarking scheme is more robust than the scheme in [58] and other SVD-based watermarking schemes.

4.7.4 Effects of Changing Watermark Size

The effects of changing the size of watermark bit stream on BER values are investigated in this subsection. The watermark bit streams of sizes 16-bit, 32-bit, and 64-bit are embedded to the eight original host images in Figure 4.8, respectively. After that, the

attacks of JPEG compression, Gaussian noise addition, uniform noise addition and StirMark bending are applied to all the watermarked images, respectively. Under each type of attack, the average extracting BER value of all eight attacked images embedded with the same bit stream is calculated. For a given watermark bit stream, the average BER values versus the attacks is plotted and shown in Figure 4.16.

Figure 4.16: (a) Average BER vs. JPEG compression (b) Average BER vs. Gaussian noise addition (c) Average BER vs. uniform noise addition (d) Average BER vs. StirMark random bending attack

As expected, BER values decrease with reduced size of watermark bit streams because the embedded watermark bit stream is repeated more times and watermark extraction

accuracy is also increased. Note that smaller size implies smaller capacity of embedded watermark information.

4.8 Chapter Summary

In this chapter, a normalization-based robust image watermarking scheme which encompasses image normalization and SVD techniques is proposed. For the proposed watermarking scheme, the host image is first normalized to a standard form and divided into non-overlapping sub-image blocks. SVD is applied to each block. By concatenating the first SVs of adjacent blocks of the normalized image, a SV block is obtained. DCT is then carried out on the SV blocks to produce SVD-DCT blocks. A watermark bit is embedded in the mid-frequency band of a SVD-DCT block by imposing a particular relationship between two pseudo-randomly selected DCT coefficients. An adaptive frequency mask is used to adjust local watermark embedding strength. Watermark extraction involves mainly the inverse process. The watermark extracting method is blind and efficient. Experimental results and data show that the quality degradation of watermarked image caused by the embedded watermark is visually transparent. The results also show that the proposed watermarking scheme is robust against various image processing operations and geometric attacks.

Chapter 5

Feature-based Robust Watermarking Using Zernike Moments

In this chapter, a novel and effective feature-based image watermarking scheme is proposed and implemented. Zernike moments that have invariance properties are adopted in the scheme. In the proposed watermarking scheme, feature points are first extracted from original host images and several circular patches centered on these points are generated. The patches are used as carriers of watermark information because they can be regenerated to locate watermark embedding positions and synchronize the embedded watermarks even when the watermarked images are severely distorted. Zernike transform is then applied to the patches to calculate local Zernike moments. Dither modulation is adopted to quantize the magnitudes of the Zernike moments according to the watermark bit stream. Experimental results and data show that quality degradation of watermarked image is visually transparent. The proposed watermarking scheme is very robust against image processing operations and geometric attacks.

5.1 Feature Point Extraction

To extract watermarks without help of the original host image, some reference points should be used because these points are perceptually significant and can resist various

image processing operations and geometric attacks as well [56]. These reference points can also be used to locate and resynchronize the embedded watermark between watermark embedding and extracting. Normally, these reference points are known as feature points. Feature points are landmarks in an image, which are often intuitively obvious to a human. For example, these points include the corners of a building, the eyes on a human face and the edges of an object, etc. [61, 102]. Traditionally, the image features such as contour segments and corners are applied in many applications, e.g. tracking and classification [62]. Recently there has been an increased interest in finding new types of image features providing the properties which are robust against geometric transformations and attacks like rotation, scaling and translation etc. Typical applications in which the aforementioned feature points are successfully applied are image matching and pattern recognition.

The feature point detectors are being used to detect and locate the feature points in digital images. A wide variety of feature point detectors exist in the open literature [100-115]. In this thesis, the focus will be given to those feature point detectors that are commonly used for digital watermarking purposes. In recent years, two types of methods, Mexican Hat wavelet and Harris detector are widely used to extract feature points. The feature points extracted by Mexican Hat wavelet [105] are stable under attack of noise addition but they are sensitive to some geometric attacks. The feature points extracted by Harris detector [106] are stable under attacks such as rotation, translation and noise addition, etc. However, they are hardly survivable to scaling as well [109]. To resolve the weakness of current feature point detectors, a novel detector called multi-scale Scale Invariant Feature Transform (SIFT) detector [111] is adopted in this thesis because it is

more effective leading to significant improvements in matching accuracy and stability. The feature points can be detected using difference-of-Gaussian (DoG) function and a cascade filtering approach [109, 111]. Based on the procedures of feature point extraction using multi-scale SIFT, the feature points of the host images in Figure 4.8, which are highly invariant against geometric attacks, are extracted and shown in Figure 5.1.

(a)

(b)

(c)

(d)

Figure 5.1: Extracted feature points on original host images (a) Lena (b) Girls (c) Fruits (d) Goldhill (e) Plane (f) Lake (g) Boat (h) Peppers

The proposed watermarking scheme will embed the watermark bits in the regions around the extracted feature points. Hence the embedded watermark bits can be accurately synchronized and extracted even if watermarked images are subject to geometric attacks.

5.2 Construction of Local Watermarking Regions

For robustness as well as preservation of visual quality, it is necessary to identify suitable regions for watermark embedding. These regions are used as carriers of watermark information. They should be readily regenerated to locate the watermark embedding positions and synchronize the embedded watermark even when watermarked images are severely distorted. As explained in the last section, image feature points are invariant and robust against geometric attacks and distortions. In general, regions around the feature points, which are subsets of original host image, contain higher energy than other regions of the image so these regions are much more robust against image distortions than other regions. Thus, feature points can be used to determine the local watermarking regions of host image for watermark embedding and extracting. Generally speaking, the local watermarking regions can be of any shape like rectangle, triangle or hexagon etc. The local watermarking regions should be robust against rotation attack so the circular patches centered on feature points are selected as local watermarking regions in the proposed scheme. It is known that the feature scales of feature points vary with the local image contents and characteristics [111], the strong feature points should be used to construct circular patches and the radii of the patches depend on the feature scale of the selected feature points [109, 111]. As a result, a set of adaptive circular patches centered on the selected feature points are generated. The radius of these patches is calculated as

$$R = \alpha \cdot [\sigma] \quad (5.1)$$

where $[\cdot]$ denotes round operation and σ is the feature scale of selected point; α is a positive number to adjust the size of circular patch. Obviously, a large value of α would increase watermark embedding capacity. But the robustness of proposed scheme would

decrease with a large α . The experimental results show that optimal robustness and repeatability can be achieved if the value of α is in the range of [8, 12]. Note that, the value of α is a secret parameter, without which the same circular patch cannot be generated. If there is an overlap between two circular patches, the patch centered on the feature point with smaller feature scale is eliminated. Based on the above procedures, circular patches centered on the feature points that are shown in Figure 5.1 are generated and presented in Figure 5.2.

Figure 5.2: Circular patches centered on the feature points (a) Lena (b) Girls (c) Fruits (d) Goldhill (e) Plane (f) Lake (g) Boat (h) Peppers

5.2.1 Invariance of Circular Patches

As mentioned, the circular patches are used as stable carriers of watermark information. Hence the patches should be robust against image distortions for resynchronization and extraction of embedded watermark. In this subsection, several experiments are performed to assess the robustness and invariance of generated circular patches in Figure 6.1. Geometric attacks such as rotation, scaling, shearing are applied to distort and transform

the images in Figure 5.2. After attacks, the re-extracted feature points and corresponding circular patches on the distorted images are shown in the following figures.

1) *Rotation*: The Lena image in Figure 5.2 (a) is rotated through an angle of 3° , 10° , 20° and 30° , respectively. The re-extracted feature points and corresponding circular patches on the rotated images are shown in Figure 5.3.

Figure 5.3: Circular patches on rotated images (a) 3° (b) 10° (c) 20° (d) 30°

It can be seen that a majority of feature points can be re-extracted accurately on the rotated images. The locations and radii of the regenerated circular patches are identical to the corresponding patches on the original host image. Hence, the rotation-invariant watermark can be achieved by embedding watermarking information in those circular patches.

2) *Scaling*: The Lena image in Figure 5.2 (a) is resized to 80%, 90%, 110% and 120% of its original size, respectively. The re-extracted feature points and corresponding regenerated circular patches on the resized images are shown in Figure 5.4.

Figure 5.4: Circular patches on resized images (a) 80% (b) 90% (c) 110% (d) 120%

It can be seen that more than half of the feature points and circular patches can be accurately re-generated on the resized images even when the original image is significantly resized. Hence, it is possible to create scaling-invariant watermark by embedding watermarks in these circular patches.

3) *Shearing*: The shearing attacks are applied to transform the Lena image in Figure 5.2 (a) with the following different parameters: (a) (5%, 1%), (b) (6%, 1%), (c) (1%, 3%) and (d) (1%, 6%). The re-extracted feature points and corresponding regenerated circular patches on the transformed images are shown in Figure 5.5.

Figure 5.5: Circular patches on sheared images (a) (5%, 1%) (b) (6%, 1%) (c) (1%, 3%) (d) (1%, 6%)

It is observed that even when the image is severely distorted, there are still sufficient number of feature points that can be accurately re-extracted. Watermark embedded in the patches centered on these points will be robust against shearing attacks.

Similar results can be obtained for other images in Figure 5.2. Based on the above explanations and experimental results, it can be concluded that the circular patches centered on the image feature points are very invariant to geometric attacks and distortions. Hence the watermarks embedded in these patches are also very robust against geometric attacks and they can be synchronized even when the watermarked image is severely distorted.

5.3 Zernike Moments and Invariant Properties

Zernike moments consist of a set of complex polynomials that form a complete orthogonal set over the interior of a unit disk [115-120]. They are widely used in image processing, pattern recognition, and multi-resolution analysis. Zernike moments are ideal region-based shape descriptors and they have been shown to be invariant against rotation, flipping, scaling and noise addition [121,122]. In this section, Zernike moments and their invariance properties are explained in detail. Results of experiments to test invariance and stability of Zernike moments are also presented.

5.3.1 Zernike Moments

The Zernike moments Z_{nm} of order n with repetition m for a continuous function $f(x, y)$ that vanishes outside a unit disk are defined as

$$Z_{nm} = \frac{n+1}{\pi} \iint_{x^2+y^2 \leq 1} f(x, y) \cdot V_{nm}^*(x, y) dx dy \quad (5.2)$$

where n is a nonnegative integer; m is an integer such that $n - |m|$ is nonnegative and even. The complex-valued function $V_{nm}(x, y)$ is defined as

$$V_{nm}(x, y) = V_{nm}(\rho, \theta) = R_{nm}(\rho) \cdot \exp(jm\theta) \quad (5.3)$$

where $\rho = \sqrt{x^2 + y^2}$ and $\theta = \tan^{-1}(y/x)$. They represent polar coordinates over the unit disk and R_{nm} are Zernike radial polynomials of ρ which is given by

$$R_{nm}(\rho) = \sum_{s=0}^{\frac{n-|m|}{2}} \frac{(-1)^s [(n-s)!] \rho^{n-2s}}{s! \left(\frac{n+|m|}{2} - s\right)! \left(\frac{n-|m|}{2} - s\right)!} \quad (5.4)$$

Note that $R_{nm}(\rho) = R_{n,-m}(\rho)$. For a digital image signal, the integrals in Equation (5.2) are replaced by summations and its Zernike moments are calculated as

$$Z_{nm} = \frac{n+1}{\pi} \sum_x \sum_y f(x, y) \cdot V_{nm}^*(\rho, \theta) \quad (5.5)$$

Suppose that if the Zernike moments Z_{nm} of $f(x, y)$ up to a given order N ($N \leq n$) are known, the original signal function can be reconstructed as

$$\hat{f}(x, y) = \sum_{n=0}^N \sum_m Z_{nm} \cdot V_{nm}(\rho, \theta) \quad (5.6)$$

However, the quality of reconstructed image through this process is significantly degraded due to large cumulative computation errors. The demonstration of reconstruction process using Zernike moments is shown in Figure 5.6.

Figure 5.6: Reconstructed images using Zernike moments from orders 10 to 30

5.3.2 Testing of Invariance properties

In this subsection, several experiments are performed to test and assess the invariance properties of Zernike moments. The reasons Zernike moments are adopted for robust watermarking is that they have some very special and important properties. For example, the magnitudes of Zernike moments are invariant against rotation, flipping, and noise addition.

Suppose that the original signal function $f(\rho, \theta)$ is rotated through an angle α resulting in a rotated signal f' . The relationship between the rotated signal and the original signal in the same coordinates is given by $f'(\rho, \theta) = f(\rho, \theta - \alpha)$, and the Zernike moments of $f'(\rho, \theta)$ can be calculated using

$$Z'_{nm} = \frac{n+1}{\pi} \int_{-\pi}^{\pi} \int_0^1 f(\rho, \theta - \alpha) [R_{nm}(\rho) e^{-jm\theta}]^* r dr d\theta \quad (5.7)$$

Let $\theta' = \theta - \alpha$, Equation (5.7) can then be expressed as

$$\begin{aligned} Z'_{nm} &= \frac{n+1}{\pi} \int_{-\pi}^{\pi} \int_0^1 f(\rho, \theta') [R_{nm}(\rho) e^{-jm(\theta'+\alpha)}]^* r dr d\theta' \\ &= \left[\frac{n+1}{\pi} \int_{-\pi}^{\pi} \int_0^1 f(\rho, \theta') [R_{nm}(\rho) e^{-jm\theta'}]^* r dr d\theta' \right] e^{-jm\alpha} = Z_{nm} e^{-jm\alpha} \end{aligned} \quad (5.8)$$

From the above equations, it can be shown that $|Z'_{nm}| = |Z_{nm}|$ and their magnitudes are equal. Therefore, if a watermark is embedded in the magnitudes of Zernike moments, it is highly invariant and robust against rotation attacks. One experiment is performed to test the invariance property of the magnitudes of Zernike moments. The original Lena image and its rotated image are shown in Figure 5.7, respectively.

Figure 5.7: (a) Original Lena image (b) Rotated Lena image through 30°

The first ten elements from the Zernike moments series of the above two images and their magnitudes are calculated and tabulated in Table 5.1.

Table 5.1: Zernike moments and magnitudes of two images

(a) Original image		(b) Rotated image	
Zernike moment	Magnitude	Zernike moment	Magnitude
5.0420	5.0420	-5.0420	5.0420
0.4936 - 0.2967i	0.5759	-0.2967 - 0.4936i	0.5759
0.1753	0.1753	0.1753	0.1753
-0.0010 + 0.4354i	0.4356	-0.0010 - 0.4354i	0.4356
0.0533 - 0.5805i	0.5830	0.5805 + 0.0533i	0.5830
-0.3400 - 0.2357i	0.4137	0.2357 + 0.3400i	0.4137
0.0869	0.0869	0.0869	0.0869
-0.5671 - 0.1504i	0.5867	0.5671 + 0.1504i	0.5867
0.3562 - 0.0810i	0.3653	0.3562 - 0.0810i	0.3653
-0.3518 + 0.1602i	0.3866	-0.1602 - 0.3518i	0.3866

From Table 5.1, it can be seen that the magnitude of one specific element is invariant even if the original image is rotated. Similar results can be obtained if the image is rotated through other angles.

If the original signal is flipped horizontally, the Zernike moments Z'_{nm} of the flipped signal are

$$\begin{cases} Z'_{nm} = Z_{nm}^* , & \text{if } m \text{ is an even integer} \\ Z'_{nm} = -Z_{nm}^* , & \text{if } m \text{ is an odd integer} \end{cases} \quad (5.9)$$

Similarly, if the original signal is flipped vertically, the Zernike moments Z'_{nm} of the flipped signal is always Z_{nm}^* irrespective of m . In either case, the magnitudes of Zernike moments are invariant, which means that the watermark embedded in the magnitudes of Zernike moments is invariant and robust against flipping attacks. Besides the invariance properties, Zernike moments are orthogonal and orthogonal moments have been proven to be invariant in the presence of noise [117,128]. Thus the magnitudes of Zernike moments are also invariant against addition of noise.

Other than rotation attacks, the watermark embedded in the magnitudes of Zernike moments should also be robust against other geometric attacks as well as image processing operations such as JPEG compression or filtering. In the following experiments, the invariance and stability of Zernike moments with different orders are determined when some very damaging operations are performed on the images.

The following mathematical expressions are designed to measure the variations of the magnitudes of Zernike moments after specific operations.

$$M = |Z_{nm}| \quad , \quad M' = |Z'_{nm}| \quad (5.10)$$

where Z_{nm} are the original Zernike moments for order n and Z'_{nm} are the affected Zernike moments after specific operations; M and M' are the corresponding magnitudes. The quotient of M'/M is used as an indicator of the variations of magnitudes. The original Lena image is used as the test image. Zernike transforms with different orders are applied to the test image to calculate original Zernike moments.

1) *The influence of JPEG compression and sharpening*: JPEG compression with quality

factors of 15 and 20, spatial sharpening with the kernel of $\begin{pmatrix} 0 & -1 & 0 \\ -1 & 5 & -1 \\ 0 & -1 & 0 \end{pmatrix}$ are applied

to the test image. After that, Zernike transforms are applied again to calculate the affected Zernike moments after operations as well as the variations of magnitudes. The relationship between the variations of magnitudes and the orders is plotted in Figure 5.8.

Figure 5.8: Variations of magnitudes under JPEG compression and sharpening

If the variation value under a given order is close to 1.0, then the magnitude of Zernike moments under that order is stable and invariant against the operation. In Figure 5.8, it can be seen that the stable region starts with the order of 0 and ends with the order of 29, which means the Zernike moments that are generated under the orders from 0 to 29 are stable and invariant against the operations. This shows that for watermarking to be robust against JPEG compression or sharpening, the watermarks should be embedded in the magnitudes of the Zernike moments that are generated between orders of 0 and 29.

2) *The influence of filtering*: Median filters of sizes 5×5 and 7×7 , and Gaussian filter

with kernel of $(1/8) \begin{pmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 1 & 1 \end{pmatrix}$ are applied to the test image. The relationship between

the variations of magnitudes and the orders is plotted in Figure 5.9.

Figure 5.9: Variations of magnitudes under filtering

From Figure 5.9, it can be seen that the stable region starts with the order of 0 and ends with the order of 31 for all three cases.

3) *The influence of noise addition*: The following different types of noises like Uniform noise (0.3), Gaussian noise (0.3) and Salt & Pepper noise (0.05) are applied to the test image, respectively. The numbers in the brackets indicate the standard deviation of added noise which is normalized to the range of $[0, 1]$. The relationship between the variations of magnitudes and the orders is plotted in Figure 5.10.

Figure 5.10: Variations of magnitudes under noise addition

The stable region in Figure 5.10 starts with the order of 0 and ends with the order of 35 for all three cases.

4) *The influence of scaling and shearing*: The scaling attacks with the parameters of 70% and 120%, and shearing attack with the parameter of (5%, 6%) are applied to the test image. The relationship of the variations between magnitudes and the orders is plotted in Figure 5.11.

Figure 5.11: Variations of magnitudes under scaling and shearing attacks

The stable region in Figure 5.11 starts with the order of 0 and ends with the order of 32 for all three cases.

5) *The influence of StirMark random bending attack:* The StirMark random bending attacks with the degrees of 6, 7 and 8 are applied to the test image, respectively. The relationship of the variations between magnitudes and the orders is plotted in Figure 5.12.

Figure 5.12: Variations of magnitudes under StirMark random bending attacks

For StirMark random bending attacks, the stable region in Figure 5.12 starts with the order of 0 and ends with the order of 30. Note that, the proposed watermarking scheme in Chapter 4 is not robust against high-degree StirMark random bending attack and this problem can be solved by embedding the watermark in the magnitudes of Zernike moments, which is confirmed in Figure 5.12.

The Zernike moments of stable regions are invariant against the attacks because they represent visually important and low-frequency components of the test image. It is noticed that for all the attacks and operations mentioned above, the orders of the stable

regions overlap in the range from 0 to 29. Hence, robust watermarking against the attacks mentioned can be achieved by choosing the orders between 0 and 29. Similar results can be obtained when other images are used as test images.

Other than the orders to be used, some Zernike moments are more suitable for robust watermarking than others. This is because the invariance properties of some Zernike moments are compromised due to geometric error of a unit disk clipping, approximation error of Zernike polynomial integration or interpolation error of rotated and resized images [123-125].

In summary, the following considerations should be made in the selection of Zernike moments for watermark embedding. First, the moments with order higher than a certain threshold N_{max} cannot be computed accurately and reliably due to cumulative computation errors, and thus they are ruled out. Second, due to the deviation from orthogonality of sampled Zernike polynomials, it can be shown that those moments with repetitions $m = 4i$ ($i \in Z$) cannot be computed accurately, thus they are not suitable for watermark embedding [115, 119]. Several Zernike moments that are suitable for digital watermarking are shown in Table 5.2. The magnitudes of Zernike moments are suitable for embedding robust watermarks due to their invariance properties against different types of operations and attacks. However, careful selection must be made to achieve specific applications and objectives.

Table 5.2: Suitable Zernike moments for robust watermarking

Order	Suitable Zernike moments
0	$Z_{0,0}$
1	$Z_{1,0}, Z_{1,1}$
2	$Z_{2,0}, Z_{2,1}, Z_{2,2}$
3	$Z_{3,0}, Z_{3,1}, Z_{3,2}, Z_{3,3}$
4	$Z_{4,0}, Z_{4,1}, Z_{4,2}, Z_{4,3}$
5	$Z_{5,0}, Z_{5,1}, Z_{5,2}, Z_{5,3}, Z_{5,5}$
6	$Z_{6,0}, Z_{6,1}, Z_{6,2}, Z_{6,3}, Z_{6,5}, Z_{6,6}$
7	$Z_{7,0}, Z_{7,1}, Z_{7,2}, Z_{7,3}, Z_{7,5}, Z_{7,6}, Z_{7,7}$
8	$Z_{8,0}, Z_{8,1}, Z_{8,2}, Z_{8,3}, Z_{8,5}, Z_{8,6}, Z_{8,7}$
9	$Z_{9,0}, Z_{9,1}, Z_{9,2}, Z_{9,3}, Z_{9,5}, Z_{9,6}, Z_{9,7}, Z_{9,9}$
10	$Z_{10,0}, Z_{10,1}, Z_{10,2}, Z_{10,3}, Z_{10,5}, Z_{10,6}, Z_{10,7}, Z_{10,9}, Z_{10,10}$

5.4 Watermark Embedding Process

Having identified the image regions and Zernike moments for watermark embedding, the watermark is then embedded into these regions using Zernike's moments. To calculate local Zernike moments over a circular image patch, $f(x, y)$, with radius, r , Equation (5.5) is modified as

$$Z_{nm} = \frac{n+1}{\pi r^2} \sum \sum f(x, y) \cdot V_{nm}^*(\rho, \theta) \quad x^2 + y^2 \leq 1 \quad (5.11)$$

Order n and repetition m are selected as follows. As mentioned in Section 5.3, it is desirable to choose n to be between 0 and 29, and the moments with repetitions $m = 4i$ ($i \in \mathbb{Z}$) cannot be computed accurately. As $A_{nm}^* = A_{n,-m}$, it is only necessary to consider the case when m is larger than 0. For ease of further discussion, let the set of candidate Zernike moments for watermark embedding be denoted by $S = \{Z_{nm} : n \leq N_{max}, m \geq 0, m \neq 4i, i \in \mathbb{Z}\}$. For the experiments described in this chapter, N_{max} is set to 29 to maximize the watermark embedding capacity.

To enhance the security of proposed watermarking scheme, two secret keys, labeled as encryption key and position key, are first generated. The encryption key is used to generate a pseudorandom bit sequence. This bit sequence is used to encrypt original ownership information to obtain an encrypted watermark bit stream $W = \{w(i), 1 \leq i \leq L, L < |S|\}$. The position key is used to pseudo-randomly select L Zernike moments from the set S to form a Zernike moment vector $Z = (Z_{n_1 m_1}, Z_{n_2 m_2}, \dots, Z_{n_L m_L})$, where $Z_{n_i m_i}$ is some candidate Z_{nm} in the set S . The moment vector Z is then used for watermark embedding. The dither modulation [36, 124], which is a special form of quantization index modulation for signal quantization, is adopted to quantize the magnitudes of Zernike moments in Z and embed watermark bits. After quantization, a new Zernike moment vector $Z' = (Z'_{n_1 m_1}, Z'_{n_2 m_2}, \dots, Z'_{n_L m_L})$ is produced. $Z'_{n_i m_i}$ is the quantized version of $Z_{n_i m_i}$ satisfying

$$|Z'_{n_i m_i}| = \left\lceil \frac{|Z_{n_i m_i}| - d_i(w(i))}{\Delta} \right\rceil \cdot \Delta + d_i(w(i)), 1 \leq i \leq L \quad (5.12)$$

where $[\cdot]$ denotes rounding operation and Δ is the quantization step size; $d_i(\cdot)$ is the dither function for i^{th} quantizer such that $d_i(1) = \frac{\Delta}{2} + d_i(0)$. The key-independent dither variable $d_i(0)$ is uniformly distributed over $(0, \Delta]$ and it is randomly generated by the modulation generator [51, 69]. A suitable step size, Δ , must be determined because a large Δ can increase embedding strength and robustness, but it can also degrade the visual quality of the watermarked image.

The modified Zernike moments are then calculated as

$$Z'_{n_i m_i} = \frac{|Z'_{n_i m_i}|}{|Z_{n_i m_i}|} \cdot Z_{n_i m_i}, 1 \leq i \leq L \quad (5.13)$$

Note that the conjugate, $Z^*_{n_i m_i}$, of each $Z_{n_i m_i}$ should also be quantized to have the same magnitude, so the pixel values in the reconstructed image patches are real.

Due to cumulative computational errors of Zernike transform and quantization errors in the embedding process, it is difficult to reconstruct watermarked patch without visible quality degradation directly using Equation (5.6). In the proposed scheme, in order to reduce the quality degradation of watermarked patches, the watermarked patch is reconstructed by adding quantization errors and original patch in the spatial domain. Let $e_{n_i m_i} = Z'_{n_i m_i} - Z_{n_i m_i}$ and $e^*_{n_i m_i} = Z'^*_{n_i m_i} - Z^*_{n_i m_i}$ denote the quantization errors of $Z_{n_i m_i}$ and $Z^*_{n_i m_i}$, respectively. The quantization errors in the spatial domain are expressed as

$$e(x, y) = \sum_{i=1}^L [e_{n_i m_i} \cdot V_{n_i m_i}(\rho, \theta) + e^*_{n_i m_i} \cdot V^*_{n_i m_i}(\rho, \theta)] \quad (5.14)$$

The watermarked patch $f'(x, y)$ can then be obtained by adding original patch and quantization errors in the spatial domain as follows.

$$f'(x, y) = f(x, y) + e(x, y), \quad x^2 + y^2 \leq r^2 \quad (5.15)$$

The watermark bits are then embedded in the selected Zernike moments of one patch. The above watermark embedding process repeats until all the circular patches on original host image are embedded with watermarks. This repeating process improves the robustness and security of proposed scheme. The watermark embedding process is summarized in the form of a flow chart in Figure 5.13.

Figure 5.13: Illustration of watermark embedding process

5.5 Watermark Extracting Process

The watermark extracting method of proposed scheme is a blind method. The original image is not required for watermark extraction.

Similar to the embedding process, all feature points on the received or attacked image are extracted by multi-scale SIFT detector and circular patches centered on these points are then regenerated. Zernike transform is applied to calculate local Zernike moments up to the order of 29 over each regenerated patch. The position key is used to locate and select L moments, forming a set of Zernike moments $\tilde{Z} = (\tilde{Z}_{n_1 m_1}, \tilde{Z}_{n_2 m_2}, \dots \dots \tilde{Z}_{n_L m_L})$, where watermark bits are probably embedded.

Two dither variables, $d_i(0)$ and $d_i(1)$, are then generated. Two quantized versions of each $|\tilde{Z}_{n_i m_i}|$ in $\tilde{Z}$ with respect to the two dither variables are calculated using

$$|\tilde{Z}_{n_i m_i}|_j = \left\lfloor \frac{|\tilde{Z}_{n_i m_i}| - d_i(j)}{\Delta} \right\rfloor \cdot \Delta + d_i(j), \quad 1 \leq i \leq L, j = 0, 1 \quad (5.16)$$

By comparing the distances between $|\tilde{Z}_{n_i m_i}|$ and its two quantized versions, the watermark bit embedded in $|Z_{n_i m_i}|$ can be extracted by

$$\tilde{w}(i) = \operatorname{argmin}_{j \in \{0,1\}} (|\tilde{Z}_{n_i m_i}|_j - |\tilde{Z}_{n_i m_i}|)^2, \quad 1 \leq i \leq L \quad (5.17)$$

This method is called minimum distance decoder [116].

The sequence of watermark bits so extracted is then de-encrypted using the pseudorandom sequence generated with the encryption key to recover the original ownership information. The flow chart of the watermark extraction process over one circular patch is presented in Figure 5.14. This extraction process continues until all regenerated circular patches on the watermarked image are processed and decoded.

Figure 5.14: Illustration of watermark extracting process

Some patches cannot be regenerated when watermarked image is subjected to certain attacks. In addition, some regenerated patches on the attacked image may not contain sufficient watermark information for recovery of ownership information. Thus, a regenerated patch is classified as a watermarked patch if the number of correctly extracted bits from this patch is larger than a given threshold and these correctly extracted bits should be sufficient to recover original ownership information.

5.6 False Alarm Analysis

The false positive error also known as false alarm occurs when the watermark extraction result indicates the presence of a watermark in an un-watermarked image [126]. In the

proposed scheme, to minimize the probability of false alarm, a comparison between the extracted watermark bits and the original watermark bits is necessary. For an un-watermarked patch, the extracted bits from this patch are assumed to be independent random variables (Bernoulli trials) [126]. Each extracted bit has the same probability to match the corresponding watermark bit. For random binary data, this probability is assumed to be 0.5. Let r be the number of extracted bits from one un-watermarked patch that can match the watermark bits. As explained, a patch is classified as a watermarked patch if r is larger than a given threshold. Let L be the length of watermark bit stream and T be the threshold value ($T < L$). Hence, the false alarm probability of an un-watermarked patch is, therefore, the cumulative probability of the cases that $r \geq T$. It is calculated as

$$P_{patch} = \sum_{r=T}^L (0.5)^L \cdot \left(\frac{L!}{r!(L-r)!} \right) \quad (5.18)$$

Furthermore, an image is classified as a watermarked image if at least two patches on it are classified as watermarked patches. Under this criterion, the false alarm probability of an un-watermarked image is:

$$P_{image} = \sum_{i=2}^N (P_{patch})^i \cdot (1 - P_{patch})^{N-i} \cdot \binom{N}{i} \quad (5.19)$$

where N is the total number of patches on the image. If $L=25$ and $N=10$, then the relationship between false alarm probability P_{image} and threshold T is plotted in Figure 5.15.

Figure 5.15: False alarm probability curve

The curve in Figure 5.15 drops sharply for $T > 22$ and it is desirable to have a very small false alarm probability. However, the selection is application dependant. It is assumed that the probability should be less than 0.001. In this case, T should be larger than or equal to 23, and at $T = 23$, the probability is only 0.00086. Note that the false alarm probability decreases as the value of N decreases for a given threshold value.

An experiment is performed to examine the effectiveness of the threshold value to decrease the false alarm probability. An un-watermarked image is used as a test image and a random bit sequence with 25 bits is generated for comparison. Note that, the length of the bit sequence should be larger than 22. The un-watermarked image is attacked by the following operations: (a) JPEG lossy compression with the factor of 20; (b) Median filtering of size 6×6 ; (c) Rotation with the angle of 10° ; (d) Scaling with the parameter of 120% and (e) Shearing with the parameter of (5%, 6%). The feature points and circular patches on the original and attacked images are generated. The proposed extracting procedures are applied to extract 25 bits from each patch on the images. Under each type

of attack, the number of extracted bits from the patches on the images that can match the generated binary sequence are determined and shown in Figure 5.16.

Figure 5.16: The number of matched bits in un-watermarked images (a) JPEG compression (b) Median filtering (c) Rotation (d) Scaling (e) Shearing (f) Original image without any attack

The original image has ten circular patches and some patches cannot be regenerated after the above attacks. From the experimental results, it can be seen that all the numbers

of matched bits are lower than 23 in the regenerated patches. If other un-watermarked images are used as test images, similar experimental results and data can still be obtained. Hence the threshold value of 23 can effectively decrease false alarm probability.

5.7 Experimental Results and Discussions

In this section, several experiments are carried out to assess the performance of the proposed watermarking scheme. The eight 256×256 gray-level images shown in Figure 5.17 and the images in Appendix B are used as original host images. A 25-bit binary sequence is used as original ownership information. As mentioned above, a secret encryption key is generated to encrypt the original ownership information, after which the original ownership information becomes a pseudo-random bit sequence and it will be embedded in the original host images.

Figure 5.17: Original host images (a) Lena (b) Girls (c) Fruits (d) Goldhill (e) Plane (f) Lake (g) Boat (h) Peppers

5.7.1 Assessment of Visual Quality Degradation

A basic requirement for robust image watermarking is visual transparency of the embedded watermark. In other words, the embedded watermark should not significantly degrade the visual quality of the original host image. The proposed watermarking scheme with different quantization step sizes is applied to the host images shown in Figure 5.17. PSNR is used as an objective measure of visual quality of the watermarked images. After calculation, the PSNR values for all watermarked images embedded with different quantization step sizes are tabulated in Table 5.3.

Table 5.3: PSNR (dB) values under different step sizes

PSNR (dB)		Watermarked images							
		Lena	Girls	Fruits	Goldhill	Plane	Lake	Boat	Peppers
step size Δ	1	45.2	42.8	46.3	48.6	41.9	42.4	41.9	49.6
	2	40.9	40.9	43.7	46.5	40.3	40.7	41.2	44.8
	3	37.7	37.4	39.4	44.5	35.1	36.1	35.5	43.1
	4	36.3	32.5	36.1	41.1	32.9	33.0	31.6	42.5
	5	34.6	31.1	34.4	39.7	31.3	31.8	30.0	40.7

It can be seen that for the same watermarked image, PSNR values decrease as step size increases. For a given step size, the PSNR values for highly-textured images such as “Lena”, “Fruits”, “Goldhill” and “Peppers” are higher than those for images that have simple texture. However, it is found that higher watermark embedding strength can be achieved with larger step size. Hence for robust watermarking, large step size that does not result in significant degradation in visual quality is chosen. For the following

experiments conducted, step size of 3 is used for embedding original host images, “Girls”, “Plane”, “Lake” and “Boat”; and step size of 5 is used for embedding the more textured host images, “Lena”, “Fruits”, “Goldhill” and “Peppers”. Besides PSNR, SSIM index is applied to further evaluate the visual quality of watermarked images used in the following experiment.

Table 5.4: SSIM index of watermarked images

	Lena	Girls	Fruits	Goldhill	Plane	Lake	Boat	Peppers
SSIM	0.955	0.960	0.973	0.969	0.943	0.980	0.901	0.987

It can be seen that all the values of SSIM index are higher than 0.9, which means the chosen step size of proposed scheme is able to preserve high visual quality of watermarked images.

The case of Lena image is explained in detail in the following. The watermarked image is shown in Figure 5.18 (a). For ease of displaying, the quantization errors are normalized to the range of $[-127, 128]$ and rounded to the nearest integers. They are shown in Figure 5.18 (b) and each circular patch is given a number for the purpose of explanation.

Figure 5.18: (a) Watermarked image (b) Quantization errors

It can be observed that the watermarked image in Figure 5.18 (a) is visually the same as the original host image in Figure 5.17 (a). Thus quality degradation is not visible. The circular grey patches in Figure 5.18 (b) indicate quantization errors; the white regions are the regions with the largest errors and the black regions are the regions whose pixel values are not modified. The numbers of occurrence for the values of quantization errors in several patches are plotted in the form of histograms in Figure 5.19.

Figure 5.19: Histograms for quantization errors (a) Patch 1 (b) Patch 3

It can be seen that the values of quantization errors are well spread around zero value. Similar results can be obtained for other patches in Figure 5.18 (b) and similar experimental results can be obtained if other host images are used for watermark embedding.

5.7.2 Robustness against Image Processing Operations

In this subsection, the robustness of proposed watermarking scheme is tested and evaluated. Several common image processing operations such as JPEG compression, median filtering or noise addition etc. are applied to distort the watermarked images.

1) Robustness against JPEG compression: JPEG compression with the following quality factors of 5, 10, 15, 20, 25, and 30 is applied to compress the watermarked images, respectively. After that, the feature points on those compressed image are re-extracted and the circular patches centered on the points are regenerated. In this experiment, the watermarked Lena image in Figure 5.18 (a) is used for explanation. The feature points and circular patches on the watermarked Lena image are shown in Figure 5.20 with each patch having a given number for ease of explanation.

Figure 5.20: Circular patches on the watermarked Lena image before operations

As explained, some patches cannot be regenerated after certain operations and attacks.

In Figure 5.21, the regenerated patches on the compressed Lena images are presented.

Figure 5.21: Regenerated circular patches on compressed images with different quality factors (a) 5 (b) 10 (c) 15 (d) 20 (e) 25 (f) 30

The watermark extracting procedures are applied to extract the watermark bits from the regenerated patches. As mentioned above, a patch can only be classified as a watermarked patch if the number of correctly extracted watermark bits from this patch is larger than or equal to the threshold $T = 23$. Hence the extracted bits are first compared with the original watermark bit stream to determine the number of correctly extracted watermark bits. The number of correctly extracted watermark bits from each regenerated patch on the above compressed images is shown in Figure 5.22.

Figure 5.22: Numbers of correctly extracted bits from each regenerated patch on the compressed images with different quality factors (a) 5 (b) 10 (c) 15 (d) 20 (e) 25 (f) 30

Furthermore, an image can only be classified as a watermarked image if at least two patches on the image can be classified as watermarked patches. From experimental results in the above figures, it can be seen that even when the watermarked image is severely compressed, there are still two regenerated patches can be classified as watermarked patches. Hence the attacked image can be classified as a watermarked image. The same experiment is performed on other images in Figure 5.17 and similar experimental results can be obtained. All experimental results are summarized in Table 5.5. Note that, all data in Table 5.5 is expressed as fractional numbers, where denominators stand for the number of regenerated patches on the attacked image and numerators stand for the number of watermarked patches from regenerated patches.

Table 5.5: Numbers of watermarked patches under JPEG compression

		Watermarked images							
		Lena	Girls	Fruits	Goldhill	Plane	Lake	Boat	Peppers
Compression factors	5	2/5	2/4	2/5	3/5	2/3	3/5	3/4	4/6
	10	2/5	2/4	3/5	3/6	2/4	3/5	3/5	5/9
	15	3/7	3/5	3/7	4/8	2/4	4/6	4/6	6/9
	20	3/8	3/5	5/9	4/10	2/4	4/6	4/8	6/11
	25	3/8	4/8	5/10	5/10	3/6	4/9	4/9	8/12
	30	4/9	5/8	6/10	5/10	3/6	4/10	5/9	9/12

When the watermarked images are severely compressed by JPEG lossy compression, the smallest number of watermarked patches is two, as shown in Table 5.5. It means that all compressed images can be classified as watermarked images. Hence the proposed watermarking scheme is very robust against JPEG lossy compression.

2) *Robustness against median filtering*: Median filtering is applied to the watermark images with the following different sizes: (a) 3×3 , (b) 4×4 , (c) 5×5 , (d) 6×6 , (e) 7×7 and (f) 8×8 . After that, the feature points and circular patches are re-generated followed by extracting watermark bits. The numbers of watermarked and re-generated patches are tabulated in Table 5.6.

Table 5.6: Numbers of watermarked patches under median filtering

		Watermarked images							
		Lena	Girls	Fruits	Goldhill	Plane	Lake	Boat	Peppers
Median filter sizes	(a)	3/8	4/8	4/9	6/11	3/7	4/9	5/8	8/11
	(b)	3/8	4/7	4/7	5/11	3/6	4/9	5/7	8/11
	(c)	3/7	3/7	4/7	5/9	3/6	4/9	4/7	7/10
	(d)	3/6	3/7	3/7	4/6	2/4	3/8	4/6	5/9
	(e)	2/4	2/5	3/6	3/6	2/4	3/5	4/5	4/6
	(f)	2/4	2/3	3/6	2/4	2/2	3/5	3/5	4/6

The data in Table 5.6 shows that all filtered images can be classified as watermarked images. Hence the proposed watermarking scheme is able to achieve high robustness against median filtering.

3) *Robustness against noise addition*: The following different types of noise are added to the watermarked images to assess the robustness, (a) Uniform noise (0.2), (b) Uniform noise (0.3), (c) Gaussian noise (0.2), (d) Gaussian noise (0.3), (e) Salt & pepper noise (0.05), and (f) Salt & pepper noise (0.08). The numbers of watermarked and re-generated patches are tabulated in Table 5.7.

Table 5.7: Numbers of watermarked patches under noise addition

		Watermarked images							
		Lena	Girls	Fruits	Goldhill	Plane	Lake	Boat	Peppers
Added noise	(a)	5/9	8/8	6/10	6/11	5/7	7/10	7/9	10/12
	(b)	4/9	6/8	6/10	6/11	4/6	6/10	7/9	9/12
	(c)	5/8	6/8	5/9	6/10	4/7	5/9	7/10	9/11
	(d)	5/8	5/7	4/9	6/9	3/6	4/8	6/9	8/10
	(e)	3/7	3/5	4/8	4/7	3/5	4/6	3/6	6/9
	(f)	3/5	3/5	3/6	4/7	2/4	2/5	3/6	4/7

The numbers of watermarked patches are all very high in Table 5.7. Hence it can be concluded that the proposed watermarking scheme is very robust against noise addition.

4) *Common image processing operations*: Common image processing operations: (a)

Spatial sharpening with the kernel of $\begin{pmatrix} 0 & -2 & 0 \\ -2 & 3 & -2 \\ 0 & -2 & 0 \end{pmatrix}$, (b) Spatial sharpening with the

kernel of $\begin{pmatrix} 0 & -1 & 0 \\ -1 & 5 & -1 \\ 0 & -1 & 0 \end{pmatrix}$, (c) Gaussian filtering with kernel of $(1/6) \begin{pmatrix} 1 & 2 & 2 \\ 3 & 3 & 2 \\ 1 & 2 & 1 \end{pmatrix}$, (d)

Gaussian filtering with kernel of $(1/8) \begin{pmatrix} 6 & 2 & 2 \\ 1 & 5 & 5 \\ 3 & 1 & 3 \end{pmatrix}$, (e) FMLR attack, and (f) Color

quantization. The above operations are applied to the watermarked images and numbers of watermarked patches are shown in Table 5.8.

Table 5.8: Numbers of watermarked patches under common image processing operations

		Watermarked images							
		Lena	Girls	Fruits	Goldhill	Plane	Lake	Boat	Peppers
Image processing operations	(a)	5/8	3/6	3/6	5/8	3/4	5/9	5/6	6/10
	(b)	4/6	3/5	2/6	5/7	3/5	3/7	4/7	4/9
	(c)	4/7	4/6	4/9	4/6	4/6	4/9	3/7	5/11
	(d)	3/7	3/5	3/7	3/9	2/5	5/8	4/8	4/7
	(e)	5/8	4/7	5/9	5/10	3/4	3/6	3/5	6/9
	(f)	3/6	3/6	3/8	4/9	2/4	3/7	4/6	4/8

The numbers of watermarked patches are all high so the robustness against these operations is also high. Similar results can be obtained if the images in Appendix B are used for the experiments. Hence it can be concluded that the proposed watermarking scheme is highly robust against image processing operations such as JPEG compression, filtering, noise addition as well as other common image processing operations.

5.7.3 Robustness against Geometric Attacks

The proposed watermarking scheme has been proven to be robust against image processing operations. In this subsection, the robustness against geometric attacks are tested and evaluated.

1) Robustness against rotation: The watermarked images are rotated through an angle of (a) 1° (b) 2° (c) 5° (d) 10° (e) 30° and (f) 45° , respectively. The feature points and circular patches on the rotated images are regenerated. The watermark bits are extracted

and the numbers of watermarked patches are determined. All experimental results and data are given in Table 5.9.

Table 5.9: Numbers of watermarked patches under rotation attacks

		Watermarked images							
		Lena	Girls	Fruits	Goldhill	Plane	Lake	Boat	Peppers
Rotation angles	(a)	6/7	6/8	7/9	9/9	7/7	7/8	6/7	9/11
	(b)	7/7	7/7	8/9	8/9	6/7	6/8	8/8	9/10
	(c)	6/7	6/7	7/8	9/10	6/6	8/8	7/7	10/11
	(d)	6/7	6/6	7/8	8/8	4/6	7/10	7/9	7/9
	(e)	6/6	7/7	6/6	8/10	5/5	6/9	8/9	8/8
	(f)	6/6	6/6	7/7	8/9	5/7	7/7	6/8	8/10

Under each type of attack, all the numbers of watermarked patches on different rotated images are larger than two. Hence the proposed watermarking scheme is very robust against rotation attacks.

2) *Robustness against scaling*: The watermarked images are resized in x direction by the following different parameters (a) 70% (b) 80% (c) 90% (d) 110% (e) 120% and (f) 150%, respectively. The feature points and circular patches on the resized images are regenerated. The watermark bits are extracted and the numbers of watermarked images are determined. Table 5.10 presents all experimental results.

Table 5.10: Numbers of watermarked patches under scaling attacks

		Watermarked images							
		Lena	Girls	Fruits	Goldhill	Plane	Lake	Boat	Peppers
Scaling parameters	(a)	6/7	4/5	5/7	4/7	3/5	5/7	6/8	5/9
	(b)	6/8	4/6	6/8	6/8	5/5	6/9	6/9	7/10
	(c)	7/8	6/8	7/8	7/10	4/6	6/9	7/8	8/10
	(d)	7/9	7/8	8/8	7/10	5/6	7/9	7/9	8/11
	(e)	6/7	5/7	7/10	8/9	4/5	7/8	6/8	6/9
	(f)	5/6	5/7	4/8	6/9	3/4	5/7	5/7	6/8

All the numbers of watermarked patches are larger than two even when the watermarked images are significantly resized. If the images are resized in y or both directions, similar results can still be obtained. Thus, the proposed watermarking scheme is highly robust against scaling attacks.

3) *Robustness against shearing*: The shearing attacks in x and y directions with the following different parameters are applied to the watermarked images, (a) (5%, 1%), (b) (2%, 5%), (c) (1%, 3%), (d) (4%, 3%), (e) (4%, 2%), and (f) (3%, 5%). All experimental results and data are shown in Table 5.11.

Table 5.11: Numbers of watermarked patches under shearing attacks

		Watermarked images							
		Lena	Girls	Fruits	Goldhill	Plane	Lake	Boat	Peppers
Shearing parameters	(a)	4/7	3/6	7/9	6/8	3/4	5/6	3/5	7/10
	(b)	4/6	5/7	5/8	5/7	3/5	6/9	3/4	6/9
	(c)	5/7	4/7	4/6	7/9	5/6	4/7	5/6	7/10
	(d)	5/8	3/5	5/7	5/7	4/5	4/8	4/7	5/8
	(e)	3/5	4/6	6/7	6/8	2/3	5/7	4/5	6/8
	(f)	4/6	3/5	5/8	5/6	2/4	6/8	6/8	5/7

If the shearing attacks are performed only in x or y direction, similar results can still be obtained. Hence it can be concluded that the proposed scheme is also robust against shearing attacks.

4) *Robustness against StirMark random bending attack*: The StirMark random bending attacks with the following different degrees are applied to distort the watermarked images, (a) 3 (b) 4 (c) 5 (d) 6 (e) 7 and (f) 8. The feature points and circular patches on the distorted images are regenerated. The watermark bits are extracted and the numbers of watermarked patches are shown in Table 5.12.

Table 5.12: Numbers of watermarked patches under StriMark bending attacks

		Watermarked images							
		Lena	Girls	Fruits	Goldhill	Plane	Lake	Boat	Peppers
Degrees of attack	(a)	4/8	3/8	4/8	5/9	3/6	4/9	4/9	6/10
	(b)	3/7	3/7	4/7	4/9	2/5	4/7	4/8	6/10
	(c)	3/7	3/5	3/7	3/7	2/5	3/6	3/5	4/8
	(d)	3/5	2/5	3/6	2/6	2/3	3/6	2/5	4/7
	(e)	2/4	2/4	2/5	2/4	1/3	3/4	2/3	4/7
	(f)	2/4	1/3	2/4	2/4	1/2	2/4	2/3	3/5

After applying StirMark random bending attacks, the numbers of watermarked patches decreased slightly when compared with those in previous experiments because StirMark random bending attack is able to defeat many watermarking schemes. Even though, most of attacked images can still be classified as watermarked images and the results are significantly improved when compared with the scheme proposed in Chapter 4. Thus, the proposed watermarking scheme is very robust against geometric attacks even when the high-degree StirMark random bending attacks are applied to attack the watermarked images. This can be considered as a significant performance enhancement when compared to other methods in the open literature.

5.7.4 Robustness against Combination Attacks

In this subsection, several combination attacks are designed to further test the robustness of proposed watermarking scheme.

1) *Combination attacks*: (a) JPEG 15 + Rotation 10° , (b) JPEG 15 + Scaling 80%, (c) JPEG 20 + Rotation 20° , (d) JPEG 20 + Scaling 110%, (e) JPEG 25 + Rotation 30° , and (f) JPEG 25 + Scaling 120%.

Table 5.13: Numbers of watermarked patches under combination attacks

		Watermarked images							
		Lena	Girls	Fruits	Goldhill	Plane	Lake	Boat	Peppers
Combination attacks	(a)	3/6	3/6	4/9	3/8	2/3	3/8	3/6	4/8
	(b)	4/6	3/7	3/7	3/7	3/4	3/7	3/6	4/7
	(c)	3/7	4/6	5/8	4/7	3/5	4/7	3/7	5/9
	(d)	3/6	4/6	3/7	3/6	4/5	5/9	4/8	4/9
	(e)	5/8	5/8	4/9	5/9	3/6	6/10	5/8	6/10
	(f)	4/7	3/5	3/6	3/5	2/4	4/8	2/7	3/7

2) *Combination attacks*: (a) Median filtering 4×4 + Rotation 20° , (b) Median filtering 5×5 + Rotation 30° , (c) Median filtering 6×6 + Scaling 110%, (d) Gaussian noise (0.2) + Rotation 10° , (e) Gaussian noise (0.3) + Scaling 90%, and (f) Uniform noise (0.3) + Rotation 45°

Table 5.14: Numbers of watermarked patches under combination attacks

		Watermarked images							
		Lena	Girls	Fruits	Goldhill	Plane	Lake	Boat	Peppers
Combination attacks	(a)	5/9	4/7	5/8	6/10	4/5	5/9	4/6	5/9
	(b)	4/7	3/7	3/7	4/9	3/5	3/8	3/6	4/8
	(c)	3/6	4/6	2/7	3/8	2/3	3/5	3/5	3/8
	(d)	3/7	4/8	4/8	4/8	3/6	4/8	5/7	6/10
	(e)	4/8	3/6	3/7	3/7	3/4	3/6	4/7	5/9
	(f)	5/9	4/7	4/9	6/11	5/6	6/10	4/8	8/11

Combination attacks are very damaging to watermarking schemes. However all attacked images can still be classified as watermarked images. Hence the proposed scheme is also robust against the above combination attacks.

Based on the above experimental results and data, it can be concluded that the proposed watermarking scheme has very high robustness against image processing operations and geometric attacks. Even when the high-degree StirMark random bending attacks are applied, the proposed scheme can still give relatively high robustness. Comparing with our proposed scheme in Chapter 4, higher robustness can be achieved. However, the proposed scheme is not very robust against significant cropping because some patches that contain watermark information will be eliminated or modified if watermarked images are significantly cropped.

5.7.5 Comparison with Other Watermarking Schemes

A feature-based robust watermarking scheme reported in [108] is implemented to compare the performance. That scheme first extracts feature points using Harris-Laplace detector followed by constructing local feature regions. The DFT is then performed and watermark bits are embedded by modifying mid-frequency DFT coefficients. High visual quality and robustness can be achieved. Both proposed scheme and the scheme reported in [108] are applied to the original host images and a 25-bit binary sequence is used as watermark message. The detection threshold for both schemes is set to 23 for fair comparisons. The comparison results are given in the following paragraphs.

a) Comparison of visual quality

After watermark embedding, the values of PSNR and SSIM index for several watermarked images are calculated and shown in Table 5.15.

Table 5.15: Comparison of PSNR (dB) and SSIM index

	Lena		Fruits		Lake		Boat	
	Proposed scheme	Scheme [108]	Proposed scheme	Scheme [108]	Proposed scheme	Scheme [108]	Proposed scheme	Scheme [108]
PSNR	34.6	39.2	34.4	38.8	36.1	38.0	35.5	37.9
SSIM	0.955	0.974	0.973	0.991	0.980	0.983	0.901	0.889

It can be seen that almost all the values of proposed scheme are slightly lower than those of Scheme [108]. However, the quality degradation caused by the proposed scheme is still visually transparent. Similar experimental results can be obtained for other host images.

b) Comparison of robustness against attacks

The following common operations or attacks are applied to the watermarked images to compare the robustness of both schemes, (a) JPEG 10, (b) Median filtering 5×5 , (c) Gaussian noise 0.3, (d) Rotation 5° , (e) Scaling 90%, (f) StirMark random bending attack 7, (g) JPEG 15 + Rotation 10° , (h) Median filtering 6×6 + Scaling 110%, (i) Gaussian noise 0.2 + Rotation 10° . The watermarked patches are then determined based on the extracted bits. All the experimental data shown in Table 5.16 is expressed as fractional numbers, where denominators stand for the number of regenerated patches on the attacked image and numerators stand for the number of watermarked patches from regenerated patches.

Table 5.16: Comparison of watermarked patches against attacks

	Lena		Fruits		Lake		Boat	
	Proposed scheme	Scheme [108]	Proposed scheme	Scheme [108]	Proposed scheme	Scheme [108]	Proposed scheme	Scheme [108]
(a)	2/5	2/6	3/5	2/8	3/5	1/6	3/5	2/6
(b)	3/7	1/7	4/7	2/8	4/9	2/9	4/7	3/10
(c)	5/8	2/11	4/9	3/9	4/8	3/11	6/9	4/12
(d)	6/7	5/12	7/8	4/9	8/8	6/10	7/7	7/8
(e)	7/8	5/11	7/8	5/10	6/9	5/9	7/8	6/10
(f)	2/4	0/5	2/5	1/6	3/4	1/6	2/3	0/4
(g)	3/6	3/8	4/9	4/10	3/8	2/11	3/6	2/8
(h)	3/6	2/10	2/7	2/8	3/5	1/9	3/5	2/10
(i)	3/7	1/8	4/8	2/12	4/8	2/11	5/7	4/10

It can be seen that all the numbers of watermarked patches for proposed scheme are higher than or equal to those of Scheme [108] even when very damaging attacks are applied. Similar experimental results can be obtained for other host images. Hence it can

be concluded that the proposed scheme outperforms the scheme in [108] against different types of operations and attacks.

5.8 Chapter Summary

In this chapter, a novel feature-based image watermarking scheme is proposed and implemented. Zernike moments which have invariance properties are adopted in the scheme. In the proposed scheme, feature points are first extracted from original host image and several circular patches centered on these points are generated. The patches are used as carriers of watermark information because they can be regenerated to locate watermark embedding positions even when the watermarked images are severely distorted. Zernike transform is then applied to the patches to calculate local Zernike moments. Dither modulation is adopted to quantize the magnitudes of the Zernike moments followed by false alarm analysis. Experimental results show that quality degradation of watermarked images is visually transparent. In addition, the proposed watermarking scheme is very robust against image processing operations and geometric attacks.

Chapter 6

A DFT-based Watermark Synchronization Technique Using LPM

In this chapter, a novel watermark synchronization technique is presented. The DFT and LPM are applied to achieve a geometric invariant domain. The watermark bits are embedded by adaptively modifying the LPM magnitude coefficients in the invariant domain. And the embedded watermark bits can be extracted by comparing the modified coefficients with the original coefficients.

6.1 Background Theories

This section explains the special properties of DFT [78,130] and LPM [131-139] that make them particularly suitable for watermark synchronization against geometric attacks.

Let an image be a real valued function $I(x, y)$ defined on integer-valued Cartesian grid where $0 < x < M$, $0 < y < N$. The DFT is defined as follows.

$$F(u, v) = \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} I(x, y) e^{\frac{-j2\pi xu}{M} + \frac{-j2\pi yv}{N}} \quad (6.1)$$

The inverse DFT is calculated as

$$I(x, y) = \frac{1}{MN} \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} F(u, v) e^{\frac{j2\pi xu}{M} + \frac{j2\pi yv}{N}} \quad (6.2)$$

The DFT of a real image is generally a complex-valued function so the magnitude and phase representation of the image is

$$A(u, v) = |F(u, v)| \quad (6.3)$$

$$\Phi(u, v) = \angle F(u, v) \quad (6.4)$$

The DFT for real images has certain symmetries:

$$A(u, v) = A^*(u, v) \quad (6.5)$$

And the following equations can be derived based on Equation (6.5).

$$|A(u, v)| = |A(-u, -v)| \quad (6.6)$$

$$\Phi(u, v) = -\Phi(-u, -v) \quad (6.7)$$

6.1.1 General Properties of DFT

It is important to investigate the effects of arbitrary geometric attacks on the DFT spectrum of original host image. In the case of $M=N$, the kernel of the DFT is shown as

$$xu + yv = [x, y] \begin{bmatrix} u \\ v \end{bmatrix} \quad (6.8)$$

If a linear transformation is performed in the spatial domain,

$$\begin{bmatrix} x \\ y \end{bmatrix} \rightarrow T \begin{bmatrix} x \\ y \end{bmatrix} \quad (6.9)$$

then the value of the DFT is not changed if

$$\begin{bmatrix} u \\ v \end{bmatrix} \rightarrow (T^{-1})^T \begin{bmatrix} u \\ v \end{bmatrix} \quad (6.10)$$

1) *DFT rotation*: Let a rotation matrix be $T = \begin{bmatrix} \cos \varphi & -\sin \varphi \\ \sin \varphi & \cos \varphi \end{bmatrix}$. Therefore, $(T^{-1})^T =$

$\begin{bmatrix} \cos \varphi & -\sin \varphi \\ \sin \varphi & \cos \varphi \end{bmatrix}$. Rotating the image through an angle φ in the spatial domain:

$I(x \cos \varphi - y \sin \varphi, x \sin \varphi + y \cos \varphi)$ will cause the Fourier representation to be

rotated through the same angle: $F(u \cos \varphi - v \sin \varphi, u \sin \varphi + v \cos \varphi)$. The value of the image at the nearest valid grid point can be estimated by interpolation [128].

2) *DFT scaling*: Let a scaling matrix be $T = \begin{bmatrix} \alpha & 0 \\ 0 & \beta \end{bmatrix}$. Therefore, $(T^{-1})^T = \begin{bmatrix} \frac{1}{\alpha} & 0 \\ 0 & \frac{1}{\beta} \end{bmatrix}$.

Resizing the image in the spatial domain: $I(sx, sy)$ will cause an inverse resizing in the frequency domain: $\frac{1}{s} F(\frac{u}{s}, \frac{v}{s})$, where s is the scaling parameter.

3) *DFT Shearing*: Let a shearing matrix be $T = \begin{bmatrix} 1 & \gamma \\ 0 & 1 \end{bmatrix}$. Therefore, $(T^{-1})^T = \begin{bmatrix} 1 & 0 \\ -\gamma & 1 \end{bmatrix}$.

Shearing in x direction in the spatial domain will cause an inverse shearing in v coordinate of the frequency domain and vice versa.

4) *DFT translation*: Shift in the spatial domain: $I(x + t_x, y + t_y)$ will cause a linear shift in the phase component: $F(u, v)e^{-j(t_x u + t_y v)}$. It is clear that shifts in the spatial domain affect only the phase representation of the DFT of an image. This is the well-known result that the magnitudes of the DFT are translation invariant [128].

6.1.2 Log-polar Mapping

In LPM, pixels are indexed by ring numbers and wedge numbers that are related to ordinary image coordinates by the mapping [132]. A log-polar sampled image is an image whose samples are centered on the points mapping to integral ring numbers and wedge numbers. The separation between sample points is proportional to the distance from the sampling center [130, 132]. Log-polar sampled images are often displayed on the orthogonal axes and LPM can be expressed by the following equation:

$$L_m = P * C_m \quad (6.11)$$

where L_m and C_m are the points in LPM magnitude spectrum and Cartesian magnitude spectrum, respectively; P is the LPM computation operator [132].

Consider a point $(x, y) \in R^2$. Define $x = e^\mu \cos \theta$ and $y = e^\mu \sin \theta$ where $\mu \in R$ and $0 < \theta < 2\pi$. It can be seen that for every point (x, y) there is a point (μ, θ) that corresponds to it in the new coordinate system which is known as the log-polar map. The log-polar map has the following properties. Rotation can be converted to a translation: $(x \cos \varphi - y \sin \varphi, x \sin \varphi + y \cos \varphi) \rightarrow (\mu, \theta + \varphi)$ and scaling can also be converted to a translation $(sx, sy) \rightarrow (\mu + \log s, \theta)$. From Figure 6.1, it can be seen that all sampling points with the same radius in the spatial domain correspond to the sampling points in the LPM representation along the θ -axis. The LPM representation is also considered as a geometric invariant domain.

Figure 6.1: (a) Sampling points in the spatial domain (b) Corresponding sampling points in the LPM representation

As an illustration of LPM, some original images and corresponding LPM representations are shown in Figure 6.2. The reconstructed images, which are obtained by calculating inverse LPM, are also shown in Figure 6.2. The effects of interpolation are obvious at the edges in the LPM representations.

Figure 6.2: (a) Original images in the spatial domain (b) LPM representations (c) Reconstructed images

6.2 Watermark Synchronization Procedures

6.2.1 Watermark Embedding Process

The DFT is first performed to obtain the DFT magnitude spectrum of the original host images. The DFT coefficients in the mid-frequency band are selected for watermark

embedding. There are three reasons for this selection: (1) Most of the signal energy is concentrated in low-frequency band which contains the most important visual components of the image. Hence if the coefficients in this band are modified, the image quality degradation will become visible. (2) The coefficients in the high-frequency band are not robust. They can be easily removed by lossy compression or filtering. Hence the DFT coefficients in the mid-frequency band are suitable for watermark embedding and highest robustness can be achieved without significant quality degradation. The trade-off between the robustness and image fidelity can be achieved. (3) In addition, the LPM is just like a sampling process, the closer to the center, the higher the sampling rate [132]. So if the watermark is embedded in the low-frequency band, the value change of one point in the Cartesian magnitude spectrum will cause value changes of a lot of points in the LPM magnitude spectrum because of the bilinear interpolation [131-135], which may make the watermark extracting process very complex and cause a lot of extraction errors.

After selecting the DFT coefficients in the mid-frequency band, the LPM is applied to transform the DFT magnitude spectrum to LPM magnitude spectrum. The watermark is then embedded in LPM magnitude spectrum, as shown in the white region in Figure 6.3 (a). Accordingly, the embedding positions in Cartesian magnitude spectrum can be determined by inverse LPM, as shown in the white region in Figure 6.3 (b). The original Lena image is used for demonstration. The LPM magnitude spectrum of the DFT coefficients in the mid-band is presented in Figure 6.4. For displaying purpose, the magnitudes of LPM coefficients are normalized to the range of [50, 150]. It can be seen that for one specific angle many LPM magnitude coefficients with different radii are available for watermark embedding.

Figure 6.3: (a) Watermark embedding region in LPM magnitude spectrum (b) Watermark embedding region in Cartesian magnitude spectrum

Figure 6.4: LPM magnitude spectrum for Lena image

For security issues, a secret key is generated to pseudo-randomly select one LPM magnitude coefficient from the candidate coefficients under each angle. The watermark embedding positions cannot be located without the secret key. The watermark bits are then embedded in the selected LPM magnitude coefficients by adaptively modifying the values of coefficients and creating local peaks. A variable is introduced to control the

local watermark embedding strength. The value of this variable should be selected carefully to achieve the trade-off between robustness and image fidelity. High value can increase the robustness but image quality will also be degraded. Hence the value should be chosen according to specific application and watermarking scheme. PSNR values of watermarked images can be calculated to indicate quality degradation of watermarked images that are embedded using different values of the variable. The approximate inverse LPM and DFT are then performed to obtain the final watermarked image in the spatial domain. The above watermark embedding process can be summarized in the following steps.

- 1) Perform DFT on the original host image, $I(x, y)$, to obtain the DFT magnitude spectrum $F(u, v)$.
- 2) Select the DFT coefficients in the mid-frequency band.
- 3) LPM is performed to transform the selected coefficients and obtain the LPM magnitude spectrum $F(\mu, \theta)$ to achieve a geometric invariant domain.
- 4) Use the secret key to pseudo-randomly select one LPM magnitude coefficient for each angle.
- 5) Embed the watermark bits adaptively and create local peaks according to the watermark bit stream if the selected coefficient is $F(\mu, \theta)$.

$$F_w(\mu, \theta) = \begin{cases} \alpha \cdot \sum_{\theta=1}^{\theta+1} F(\mu, \theta), & \text{if the watermark bit is "1"} \\ F(\mu, \theta), & \text{if the watermark bit is "0"} \end{cases}$$

where α is the variable that controls local watermark embedding strength and $F_w(\mu, \theta)$ is the watermarked LPM magnitude coefficient. The local peaks should be prominent enough for detection and watermark extraction.

6) Perform approximate inverse LPM and DFT to obtain the final watermarked image.

A local peak is shown in Figure 6.5 if the Lena image is used and α is set to 1. The PSNR value of watermarked image is 46.7 dB.

Figure 6.5: (a) Original LPM magnitude spectrum (b) Modified LPM magnitude with a local peak

6.2.2 Watermark Extracting Process

The proposed watermark synchronization technique is a non-blind technique so the original host image is required for watermark extraction. The watermark extracting process is simple and efficient. DFT and LPM are performed on the original host image and watermarked image, respectively. The same secret key is used to select the LPM magnitude coefficients from original host image and watermarked image that possibly carries the embedded watermark bits. For each specific angle, the subtraction between two selected coefficients is performed to detect the presence of a local peak. A threshold is then determined to differentiate watermark bit “0” and “1” according to the difference value. If a local peak is detected, the watermark bit “1” is decoded whereas the

watermark bit “0” is decoded. The final extracted watermark bit stream is a concatenation of all single decoded watermark bits.

To increase extraction accuracy, a triangular window as shown in Figure 6.6 is constructed to search and detect local peaks. The cross-correlation is performed between the triangular window and selected LPM coefficients from watermarked image. Theoretically, for a local peak, the cross-correlation value should be close to 1 whereas for non-modified coefficients, it should be close to 0. Experimental results show that the cross-correlation values of the triangular window with local peaks are more than 0.58 and those with non-modified coefficients are less than 0.39. If a cross-correlation value is between 0.39 and 0.58, then the detection method mentioned above should be applied to further check whether it is a local peak. It is found that high detection accuracy can be achieved if both detection methods are applied together.

Figure 6.6: A triangular window for local peak detection

If the watermarked Lena image is rotated through an angle of 5° or resized with the parameter of 110%, the local peak in Figure 6.5 (b) can still be detected, as shown in Figure 6.7.

Figure 6.7: (a) Detected local peak in the rotated image (b) Detected local peak in the resized image

It can be seen that the local peaks can still be detected and they are prominent enough to distinguish embedded watermarked bits. Similar results can be obtained if other geometric attacks are applied to the image. Since this technique is devised to combat geometric attacks, it is not as robust as previous proposed watermarking schemes when watermarked image is subject to image processing operations like JPEG lossy compression or median filtering and this can be confirmed by the following experiment when the watermarked Lena image is compressed or filtered.

Figure 6.8: (a) Detected local peak in the compressed image (JPEG quality factor =15) (b) Dctected local peak in the filtered image (median filter size =6x6)

It can be seen that when the watermarked image is compressed, the detected local peak is not prominent enough to distinguish the embedded watermark bits and when the image is filtered, the original local peak is completely destroyed. Hence this technique is not very robust against common image processing operations when the local watermark embedding strength is low. This problem can be solved by increasing the embedding strength. However, the visual quality will be further degraded. In addition, high computational cost is required to reduce interpolation effects and achieve accurate LPM mapping.

6.3 Performance Evaluation

For the following experiments, the original Lena image is used as a test image and a 36-bit binary sequence is used as a watermark bit stream. The watermark bit stream is repeatedly embedded into the test image when the local embedding strength α , is set to 1.5, 2, 2.5, and 3, respectively. The watermarked images and the corresponding PSNR values are presented in Figure 6.9.

(a)

(b)

Figure 6.9: Watermarked images (a) $\alpha=1.5$, PSNR=43.2dB SSIM index=0.993 (b) $\alpha=2$, PSNR=36.4dB SSIM index=0.981 (c) $\alpha=2.5$, PSNR=31.7dB SSIM index=0.875 (d) $\alpha=3$, PSNR=23.8dB SSIM index=0.801

When the watermark embedding strength is equal to 3, the visual quality of watermarked image is significantly degraded, as shown in Figure 6.9 (d). Thus the largest value of α should be less than 3. However, high value of α can increase the robustness against attacks. Hence the value of α is set to 2.5 for the following experiments to achieve highest possible robustness. Two local peaks are shown in Figure 6.10 (b) when α is equal to 2.5.

Figure 6.10: (a) Original LPM magnitude spectrum (b) Modified LPM magnitude with two local peaks

6.3.1 Assessment of Robustness

The robustness of proposed watermarking scheme is then evaluated when watermarked images are subject to geometric attacks or image processing operations.

1) *Rotation attacks*: The watermarked image is rotated through different angles. The watermark bits are extracted and extracting BER value under each rotation angle is calculated. The results are shown in Figure 6.11.

Figure 6.11: Extracting BER values under rotation attacks

It can be seen that all extracting BER values are very low when watermarked image is rotated through different angles.

2) *Scaling attacks*: The watermarked image is resized with different scaling parameters in x direction. The watermark bits are extracted and extracting BER value under a specific scaling parameter is calculated. The results are shown in Figure 6.12.

Figure 6.12: Extracting BER values under scaling attacks

It can be seen from Figure 6.12, all BER values are low even when watermarked image is significantly resized. Similar results can be obtained if the image is resized in y or both directions

3) *Shearing attacks*: The watermarked image is sheared with different shearing parameters in y direction. The watermark bits are then extracted and extracting BER values are calculated and shown in Figure 6.13.

Figure 6.13: Extracting BER values under shearing attacks

Similar results can be obtained if watermarked image is sheared in the x direction. When the shearing parameters are large, the extracting BER values are slightly increased. However, the number of correctly extracted watermark bits is still large enough to recover original ownership information. Hence it can be concluded that the proposed watermark synchronization technique is highly robust against geometric attacks and distortions like rotation, scaling and shearing.

4) *JPEG lossy compression*: In this experiment, JPEG compressions with different quality factors are applied to compress the watermarked images. The extracting BER value under specific compression factor is calculated and plotted as shown in Figure 6.14.

Figure 6.14: Extracting BER values under JPEG lossy compression

When the watermarked image is severely compressed, the extracting BER values are high. The BER values decrease as quality factors increase. Thus the proposed technique is not robust against JPEG lossy compression if the quality factor is very low.

5) *Noise addition*: The following different types of noise are added to the watermarked images, (a) Uniform noise (0.4), (b) Uniform noise (0.5), (c) Gaussian noise (0.3), and (d) Gaussian noise (0.4). All BER values are calculated and tabulated in Table 6.1.

Table 6.1: Extracting BER values under noise addition

	Added noise			
	(a)	(b)	(c)	(d)
BER	0.035	0.059	0.053	0.086

From Table 6.1, it can be seen that even if the added noise is strong, the extracting BER values are still around a low level.

6) *Median filtering*: Median filtering is applied to filter the watermarked image with the following different sizes: (a) 3×3 , (b) 4×4 , (c) 5×5 , and (d) 6×6 . The watermark bits are extracted from filtered images and for each filtered image, extracting BER values are then calculated and tabulated in Table 6.2.

Table 6.2: Extracting BER values under median filtering

	Median filter sizes			
	(a)	(b)	(c)	(d)
BER	0.081	0.188	0.324	0.416

If the filter size is small, the BER values are around a relatively low level. However, when the size is large, the embedded watermark bits are completely destroyed. Thus, the proposed technique is not robust when the filter size is very large.

Based on the above experimental results and data, it can be concluded that the proposed synchronization technique is robust against geometric attacks. For image

processing operations like JPEG lossy compression or median filtering, if the compression factor is too low or the filter size is too large, the extracting BER values are very high and the embedded watermark bits are completely destroyed.

6.3.2 Comparison with Other Watermarking Schemes

The robust watermarking scheme reported in [132] is implemented to compare the performance. Both the proposed scheme and scheme reported in [132] embed a 32-bit binary sequence into original Lena image. The parameter that controls local embedding strength is set to 2.5 for proposed scheme and 32.5 for Scheme [132] in order to achieve the same PSNR value (31.7 dB) of two watermarked images. The following attacks are applied to the watermarked images followed by extracting watermark bits, (a) JPEG 15, (b) Median filtering 4×4, (c) Gaussian noise 0.3, (d) Rotation 5°, (e) Rotation 15°, (f) Scaling 90%, (g) Scaling 120%, (h) Shearing 5%, (i) Shearing 10%. Extracting BER values are calculated and tabulated in Table 6.3.

Table 6.3: Comparison of BER values against attacks

	Watermarked images								
	(a)	(b)	(c)	(d)	(e)	(f)	(g)	(h)	(i)
Proposed scheme	0.016	0.188	0.053	0.011	0.007	0.009	0.023	0.046	0.102
Scheme [132]	0.080	0.234	0.095	0.018	0.015	0.103	0.042	0.120	0.223

All the extracting BER values of proposed scheme are lower than those of the scheme reported in [132] especially when the watermarked images are severely distorted. Similar experimental results can be obtained if other host images are used. Hence it can be

concluded that the proposed scheme is able to achieve higher robustness than the scheme reported in [132] when different types of operations and attacks are applied.

6.4 Chapter Summary

A novel watermark synchronization technique that combats geometric attacks is described in this chapter. The DFT is first applied to the original host image followed by applying LPM. The watermark bits are embedded by adaptively modifying LPM magnitude coefficients and creating local peaks. The local peaks should be detected to distinguish embedded watermark bits. From experimental results, the proposed synchronization technique is able to give high robustness against geometric attacks and common image processing operations.

Chapter 7

Conclusion and Recommendations

7.1 Conclusion

Digital watermarking is a technique or process where secret information is embedded in the digital contents. In order to make this secret information secure, imperceptible and robust, the watermark should be embedded in some specially chosen locations. The watermark can be embedded either in spatial or transform domain. High robustness against a wide range of attacks and operations is a fundamental requirement to develop real robust watermarking applications. Even though a great deal of effort has been expended by the research community on this topic, there are still unresolved issues. For robust image watermarking, improving the robustness of the watermarking scheme against geometric attacks, without sacrificing imperceptibility and security, is one of the challenges in the field of digital watermarking. The work described in this thesis is focused on the design of robust image watermarking schemes that are able to achieve high robustness against geometric attacks and distortions.

Various types, requirements, applications and attacks of digital watermarks are explained and an overview of existing watermarking methods that combat geometric attacks has been presented in this thesis. Some well-known techniques in computer vision like image normalization, image moments or feature point extraction are gradually adopted in robust image watermarking schemes. The watermarking schemes and

techniques developed in this thesis utilize the desirable features of above techniques to synchronize embedded watermark and combat geometric attacks. Highest-possible robustness and security can be achieved without visible quality degradation in the proposed schemes.

In this thesis, a normalization-based robust image watermarking scheme using SVD is proposed and implemented in Chapter 4. Image normalization is a process that changes the range of pixel intensity values. The normalized image is obtained from a series of geometric transformations. This will ensure the integrity and synchronization of embedded watermark in the normalized image even when the image is geometrically distorted. Hence the original host image is first normalized to a standard form based on which subsequent watermark embedding and extracting are performed. SVD is adopted in the scheme because the SVs of an image are invariant against added disturbances. The normalized image is then divided into non-overlapping sub-image blocks. SVD is applied to each block. By concatenating the first SVs of adjacent blocks of the normalized image, a SV block is obtained. DCT is then carried out on the SV blocks to produce SVD-DCT blocks. A watermark bit is embedded in the mid-frequency band of a SVD-DCT block by imposing a particular relationship between two pseudo-randomly selected DCT coefficients. An adaptive frequency mask is used to adjust local watermark embedding strength. Watermark extraction involves mainly the inverse process. The watermark extracting method is blind and efficient. High PSNR values of watermarked images show that the quality degradation is visually transparent. The robustness is then assessed by using a wide variety of attacks like JPEG lossy compression, median filtering, rotation and scaling. Extracting BER values are relatively low under most attacks and number of

correctly extracted bits is large enough to recover original ownership information. However, there are a lot of extracting errors when the degree of StirMark random bending attack is very high.

Image feature points are perceptually significant and can resist various image processing operations and geometric attacks as well. These points can also be used to locate and resynchronize the embedded watermark between watermark embedding and extracting. Multi-scale SIFT detector is employed in this thesis to extract feature points, and the procedures of feature point extraction are demonstrated. In general, regions around the feature points contain higher energy than other regions of the image so these regions can be used as stable carriers of watermark information. They can be regenerated even when the image is severely distorted. Thus, feature points can be used to determine the local watermarking regions. Based on this fact, a feature-based watermarking scheme using Zernike moments is proposed and described in Chapter 5. Several circular patches centered on feature points are generated. The invariance properties of circular patches and Zernike moments are explained and tested. Experimental results show that Zernike moments with the orders from 0 to 29 are most invariant against distortions. Zernike transform is then applied to the patches to calculate local Zernike moments. Dither modulation is adopted to quantize the magnitudes of the Zernike moments according to the watermark bit stream. Minimum distance decoder is applied to extract watermark bits. The extraction threshold is set to 23 to effectively decrease false alarm probability. To preserve the visual quality of original host image, different images are embedded with different step sizes. Similar to the scheme in Chapter 4, this scheme is very robust against most geometric attacks and image processing operations. In addition, the limitation of the

scheme proposed in Chapter 4 is addressed in this scheme. Experimental results show that most attacked images can still be classified as watermarked images even when the degrees of StirMark random bending attacks are very high. This is a significant performance enhancement when compared to other methods in the open literature.

A watermark synchronization technique is also described to embed watermark in an invariant domain that can be achieved by applying DFT and LPM to original host images. The invariant domain is known as a LPM representation. The effects of geometric attacks like rotation or scaling come down to translations in the LPM representation. The watermark bits are embedded by modifying LPM magnitude coefficients and creating local peaks according to the watermark bit stream. During watermark extracting process, the local peaks should be detected to distinguish embedded watermark bits. The peaks can be detected by comparing the modified coefficients with the original coefficients. To increase the extraction accuracy, a triangular window can be constructed to search and detect local peaks. The cross-correlation is performed between the window and modified coefficients. The watermark bits are determined according to the correlation values. Experimental results show that the technique is robust against geometric attacks and common image processing operations.

7.2 Recommendations for Further Research

The watermarking scheme proposed in Chapter 4 is very effective and there remains much to be explored especially on the watermark extraction accuracy. The use of error-correction codes for decreasing extracting errors is still an open problem. Error-correction coding theory was proposed in 1948. It was used for resisting the corrupting effects of

noise in data communications [36]. The embedded watermark can be considered as transmitted information and encoded by error-correction codes before embedding to increase detection accuracy. The encoding process requires design of codes, which are able to take into account many different kinds of attacks. In [69] the enhanced robustness in image watermarking is achieved by using block turbo codes. Implementation of repetition codes, convolution codes, turbo codes or Bose-Chaudhuri-Hocquenghem (BCH) codes in the proposed watermarking schemes in this thesis should be a part of the future research.

The scheme proposed in Chapter 5 is highly robust against most attacks including high-degree StirMark random bending attacks. However, this scheme is not robust against significant cropping attacks because some patches containing watermark information will be lost or modified if watermarked images are significantly cropped. One idea to combat this problem is to embed watermarks in the whole image. The original host image is first divided into non-overlapping sub-image blocks. The Zernike moments of each sub-image block are then calculated and the magnitudes are quantized as described in Chapter 5. Hence the watermark is embedded in the whole image rather than a small patch. The watermark bits can be extracted even if some parts of the image are cropped. In the future, use of machine learning techniques like genetic algorithm (GA), support vector machine (SVM) or artificial neural network (ANN) are expected to achieve higher robustness and imperceptibility [125].

The technique presented in Chapter 6 is not very robust against certain operations that severely distort the image content. One suggestion to improve the technique is to adopt feature point extraction as described in Chapter 5. The circular patches centered on the

feature points can be used as watermark information carrier. The DFT and LPM are then applied to create local peaks and embed watermark bits. The patches can be regenerated even when the watermark image is attacked and distorted. Normally, the patches contain more image energy than other regions. Hence the local peaks created in the patches should be much more prominent and they are more robust against attacks. Besides the DFT and LPM, the Fourier-Mellin transform (FMT) can also be applied to achieve an invariant domain for watermark embedding and extracting.

Appendices

Appendix A

The normalization procedures consist of the following steps for a given original host image $I(x, y)$.

1) Place the image, $I(x, y)$, at the density center. This step can be achieved by setting the transformation matrix $T = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$ and the translation vector $v = \begin{pmatrix} v_1 \\ v_2 \end{pmatrix}$ with

$$v_1 = \frac{m_{10}}{m_{00}}, \quad v_2 = \frac{m_{01}}{m_{00}} \quad (\text{A.1})$$

where m_{00} , m_{01} and m_{10} are the image moments of $I(x, y)$ as calculated in (4.1). The aim of this step is to achieve translation invariance. Let $I_1(x, y)$ denote the resulting centered image.

2) Apply a shearing transformation to the centered image $I_1(x, y)$ in the x direction with the transformation matrix $T_x = \begin{pmatrix} 1 & \alpha \\ 0 & 1 \end{pmatrix}$ so that the resulting image, which is denoted by $I_2(x, y) \triangleq T_x[I_1(x, y)]$, achieves $\mu_{30}^{(2)} = 0$, where $\mu_{30}^{(2)}$ is a central moment of $I_2(x, y)$.

3) Apply a shearing transformation to the image $I_2(x, y)$ in the y direction with the transformation matrix $T_y = \begin{pmatrix} 1 & 0 \\ \beta & 1 \end{pmatrix}$ so that the resulting image, which is denoted by $I_3(x, y) \triangleq T_y[I_2(x, y)]$, achieves $\mu_{11}^{(3)} = 0$, where $\mu_{11}^{(3)}$ is a central moment of $I_3(x, y)$.

4) Apply the scaling to the image $I_3(x, y)$ in both x and y directions with the transformation matrix $T_{x,y} = \begin{pmatrix} \gamma & 0 \\ 0 & \delta \end{pmatrix}$ so that the resulting image, which is denoted by $I_4(x, y) \triangleq T_{x,y}[I_3(x, y)]$, achieves a final normalized image with standard image size and orientation.

The final image $I_4(x, y)$ is the normalized image, based on which subsequent operations, watermark embedding and extracting are performed.

As explained in Section 4.1, a general geometric attack can be decomposed as a composition of rotation, shearing in x direction, and scaling in both x and y directions. Hence the above normalization procedures can also be explained as follows. Basically, the objective of the above normalization procedures is to eliminate the effects of each distortion or transformation component. More specifically, step 1) eliminates the translation effects of the geometric attack by setting the center of the original host image at the density center of the transformed image, steps 2) and 3) eliminate shearing effects in both x and y directions, and finally, step 4) eliminates scaling effects by forcing the normalized image to a standard size and orientation [25]. Each step in the normalization procedure should be readily invertible in order to restore the normalized image back to its original size and orientation after the watermark is embedded [20, 25]. The detailed procedures to determine the values of the parameters associated with the transformation matrixes T_x , T_y and $T_{x,y}$ are given in [24, 25].

Appendix B

The following images are also used in the experiments mentioned above. All the images are grey scale of 256×256 pixels.

Author's Publications

Journal papers

1. S.W. Foo and Q. Dong, "A normalization-based watermarking scheme using Zernike moments," *International Journal of Computer and Information Engineering, World Academy of Science, Engineering and Technology*, Vol. 3 No. 4, pp. 255-261, Autumn 2009.
2. S.W. Foo and Q. Dong, "Audio watermarking based on compression-expansion method," *International Journal of Computer and Information Engineering, World Academy of Science, Engineering and Technology*, Vol. 4 No. 2, pp. 112-118, Spring 2010.
3. Q. Dong and S.W. Foo, "A normalization-based robust watermarking scheme using SVD and DCT" *International Journal of Information and Communication Engineering, World Academy of Science, Engineering and Technology*, Vol. 6 No. 4, pp. 205-212, Autumn 2010.
4. Q. Dong and S.W. Foo, "A Feature-based watermarking scheme using Zernike moments" *International Journal of Electrical and Computer Engineering, World Academy of Science, Engineering and Technology*, Vol. 5 No. 6, pp. 375-382, Autumn 2011.
5. Q. Dong and S.W. Foo, "Geometric invariant watermarking using DFT and LPM," *Signal Processing: Image Communication*, (minor revisions).

Conference papers

1. S.W. Foo and Q. Dong, "Robustness of One Bit per Sample Audio Watermarking," *Proc. IEEE International Symposium on Circuits and Systems (ISCAS)*, pp. 932-935, 2009.
2. S.W. Foo and Q. Dong, "Audio Watermarking of Stereo Signals Based on Echo-Hiding Method," *Proc. IEEE International Conference on Information, Communications and Signal Processing (ICICS)*, pp. 1-4, 2009.

3. Q. Dong and S.W. Foo, "Robust image watermarking scheme using Zernike moments," *Proc. IEEE International Conference on Audio, Language and Image Processing (ICALIP)*, pp. 518-521, 2010.
4. Q. Dong and S.W. Foo, "A DWT-DCT watermarking scheme based on image normalization," *Proc. IEEE International Conference on Information, Communications and Signal Processing (ICICS)*, pp. 176-179, 2011.
5. Q. Dong and S.W. Foo, "A Block-based Semi-fragile Watermarking Scheme Using Fuzzy C-means Clustering Algorithm," *Proc. IEEE International Symposium on Circuits and Systems (ISCAS)*, 2012 (accepted).

Bibliography

- [1] M. Swanson, B. Zhu, and A. Tewfik, "Transparent robust image watermarking," *Proc. IEEE Int. Conf. on Image Processing*, vol. 3, pp. 211-214, Sept.1998.
- [2] I. J. Cox, M. L. Miller, and J. A. Bloom, "Digital Watermarking," *San Mateo, CA: Morgan Kaufmann*, 2001.
- [3] J. J. Ruanaidh, W. J. Dowling, and F. M. Boland, "Watermarking digital images for copyright protection," in *Proc. Inst. Elect. Eng.*, vol. 143, no. 4, pp. 250–256, Aug. 1999.
- [4] I. Cox, J. Kilian, T. Leighton, and T. Shamon, "Secure spread spectrum watermarking for multimedia information hiding," *Lecture Notes in Computer Science*, No. 1174, Springer-Verlag, pp. 39-48, 2000.
- [5] Anthony T. S. Ho, Y. Q. Shi, H. J. Kim, and M. Barni: *Digital Watermarking, 8th International Workshop, IWDW 2009*, Guildford, UK, August, 2009. *Proceedings Lecture Notes in Computer Science*, Volume 5703, Springer 2009.
- [6] H. J. Kim, T. H. Kim, I. K. Yeo, "A robust audio watermarking scheme," in *IEEE International Symposium on Circuits and Systems (ISCAS)*, pp. 696-699, 2004.
- [7] N. Bi, Q. Sun, D. Huang, Z. Yang, and J.W. Huang, "Robust image watermarking based on multiband wavelets and empirical mode decomposition," *IEEE Transactions on Image Processing*, Volume 16, Issue 8, Aug. 2007.
- [8] Z.M. Lu, D.G. Xu, and S.H. Sun, "Multipurpose image watermarking algorithm based on multistage vector quantization," *IEEE Transactions on Image Processing*, volume 14, Issue 6, pp. 822-831, June 2005.
- [9] C. C. Chang, C. C. Lin and Y. S. Hu, "An SVD oriented watermark embedding scheme with high qualities for the resorted images," *International Journal of Innovative Computing, Information and Control*, Vol.3, No.3, pp.609-620, 2007.
- [10] W.C. Wu and G.R. Ren, "A DCT-based robust image watermarking using local moment," *International Conference on Data Mining and Intelligent Information Technology Applications (ICMIA)*, pp. 122-126, Oct. 2011.

-
- [11] H.P. Jun and W. Wang, "Image watermarking method in multiwavelet domain based on support vector machines," *Journal of Systems and Software*, Vol. 83, Issue 8, pp. 1470-1477, Aug. 2010.
- [12] L. Li, H.H. Xu, C.C. Chang and Y.Y. Ma, "A novel image watermarking in redistributed invariant wavelet domain," *Journal of Systems and Software*, Vol. 83, Issue 6, pp. 923-929, Jun. 2011.
- [13] V. Aslantas, "An optimal robust digital image watermarking based on SVD using differential evolution algorithm," *Optics Communications*, Vol. 282, Issue 5, pp. 769-777, Mar. 2009.
- [14] C.M. Chu, H.C. Huang and J.S. Pan, "An Adaptive Implementation for DCT-Based Robust Watermarking with Genetic Algorithm," *International Conference on Innovative Computing Information and Control (ICICIC)*, pp. 19-22, Jun. 2008.
- [15] X. Sun, J. Liu, J. Sun, Q. Zhang and W. Ji, "A Robust Image Watermarking Scheme Based on the Relationship of SVD," *International Conference on Intelligent Information Hiding and Multimedia Signal Processing (IIHMSP)*, pp. 731-735, Aug. 2008.
- [16] C.Y. Chang, H.J. Wang and S.W. Pan, "A robust DWT-based copyright verification scheme with Fuzzy ART," *Journal of Systems and Software*, Vol. 82, Issue 11, pp. 1906-1915, Nov. 2009.
- [17] S. D. Lin, S.C. Shi and J.Y. Guo, "Improving the robustness of DCT-based image watermarking against JPEG compression," *Computer Standards & Interfaces*, Vol. 32, Issues 1-2, pp. 53-60, Jan. 2010.
- [18] L.P. Feng, J. Xu and L.B. Zheng, "A normalization-based robust digital watermarking scheme using nonsubsampling contourlet transform," *International Conference on Electronics, Communications and Control (ICECC)*, pp. 1250-1253, Sept. 2011.
- [19] W. Lu, H. Lu and F.L. Chung, "Feature based robust watermarking using image normalization," *Computers & Electrical Engineering*, Vol. 36, Issue 1, pp. 2-18, Jan. 2010.

-
- [20] P.P. Niu, X.Y. Wang, H.B. Jin and M.Y. Lu, "A feature-based robust digital image watermarking scheme using bandelet transform," *Optics & Laser Technology*, Vol. 43, Issue 3, pp. 437-450, Apr. 2011.
- [21] Y. Chen, F. Qu, J. Hu and Z. Chen, "A geometric robust watermarking algorithm based on DWT-DCT and Zernike moments," *Wuhan University Journal of Natural Sciences, SpringerLink*, Vol. 13, Issue 6, pp. 753-758, 2008.
- [22] C.H. Lai and J.L. Wu, "Robust image watermarking against local geometric attacks using multiscale block matching method," *Journal of Visual Communication and Image Representation*, Vol. 20, Issue 6, pp. 377-388, Aug. 2009.
- [23] H. Zhu, M. Liu and Y. Li, "The RST invariant digital image watermarking using Radon transforms and complex moments," *Digital Signal Processing*, Vol. 20, Issue 6, pp. 1612-1628, Dec. 2010.
- [24] C.S Lu, "Towards robust image watermarking: combining content-dependent key, moment normalization, and side-informed embedding," *Signal Processing: Image Communication*, Vol. 20, Issue 2, pp. 129-150, Feb. 2005.
- [25] Ping Dong, Jovan G. Brankov, Nikolas P. Galatsanos, Yongyi Yang and Franck Divoine, "Digital Watermarking Robust to Geometric Distortions," *IEEE Transactions on Image Processing*, Vol. 15, No. 12, Dec. 2005.
- [26] C. Li and H. Song, "A geometrically robust watermarking scheme based on perceptual hashes and genetic Algorithm," *International Conference on Computer Science & Education (ICCSE)*, pp. 673-678, Nov. 2009.
- [27] J. Liu, X. Niu, and W. Kong, "Image watermarking based on singular value decomposition," in *Proc. of the Int. Conf. Intelligent Information Hiding and Multimedia Signal Processing*, pp. 457-460, 2006.
- [28] Jean-Luc Dugelay, Stéphane Roche, Christian Rey, and Gwenaël Doërr, "Still-Image Watermarking Robust to Local Geometric Distortions," *IEEE Transactions on Image Processing*, Vol. 15, No. 9, Sept. 2006.
- [29] F. Divoine, P. Bas, P.-A. Hebert, and J.-M. Chassery, "Watermarking et résistance aux déformations géométriques," in *Proc. ompression et Représentation des Signaux Audiovisuels*, Jun. 1999.

-
- [30] J. Lichtenauer, I. Setyawan, T. Kalker, and R. Lagendijk, "Exhaustive geometrical search and the false positive watermark detection probability," *Proc. SPIE—Security and Watermarking of Multimedia Contents*, vol. 5020, pp. 203–214, Jan. 2003.
- [31] D. Gruhl and W. Bender, Affine Invariance 1995 [Online]. Available: <http://nif.www.media.mit.edu/DataHiding/affine/affine.html>.
- [32] S. Voloshynovskiy, F. Deguillaume, and T. Pun, "Multibit digital watermarking robust against local nonlinear geometrical distortions," in *Proc. IEEE Int. Conf. Image Processing*, Thessaloniki, Oct. 2001, pp. 999–1002.
- [33] Chun-Shien Lu, Shih-Wei Sun, Chao-Yong Hsu, and Pao-Chi Chang, "Media Hash-Dependent Image Watermarking Resilient Against Both Geometric Attacks and Estimation Attacks Based on False Positive-Oriented Detection," *IEEE Transactions on Multimedia*, vol. 8, No. 4, Aug. 2006.
- [34] I. Nasir, F. Khelifi, J. Jiang and S. Ipson, "Robust image watermarking via geometrically invariant feature points and image normalization," *IET Image Processing*, vol. 6, pp. 353–363, 2012.
- [35] D. Delannay and B. Macq, "A method for hiding synchronization marks in scale and rotation resilient watermarking schemes," *Proc. SPIE—Security and Watermarking of Multimedia Contents IV*, vol. 4675, pp. 548–554, Jan. 2002.
- [36] Shiyuan Hu, "Geometric-Invariant Image Watermarking by Key-Dependent Triangulation," *Informatica* 32 (2008) pp. 169–181.
- [37] I. Cox, M. Miller, and A. McKellips, "Watermarking as communications with side information," *Proceedings of the IEEE*, vol. 87, no. 7, pp. 1127–1141, Jul 1999.
- [38] V. Solachidis and I. Pitas, "Self-similar ring shaped watermark embedding in 2-D DFT domain," in *Proc. Eur. Signal Processing Conf.*, Sep. 2000, vol. 4.
- [39] Patrick Bas and Alejandro Lobo Guerrero, "First wavila challenge: several considerations on the security of a feature-based synchronization scheme for digital image watermarking," www.encrypt.eu.org.
- [40] Zhen Li, Kim-Hui Yap and Bai-Ying Lei, "A new blind robust image watermarking scheme in SVD-DCT composite domain," *18th IEEE International Conference on Image Processing*, pp. 11–14 Sept. 2011.

- [41] Bai Ying Lei, Ing Yann Soon and Zhen Li, "Blind and robust audio watermarking scheme based on SVD-DCT," *Signal Processing*, Volume 91, Issue 8, August 2011, Pages 1973–1984.
- [42] M. Kutter and S. Winkler, "A vision-based masking model for spread spectrum image watermarking," *IEEE Trans. Image Process.*, vol. 11, no. 1, pp. 16–25, Jan. 2002.
- [43] K.L. Chung, C.H. Shen, L.C. Chang, "A novel SVD- and VQ-based image hiding scheme," *Pattern Recognition Letter*, pp. 1051–1058, 2001.
- [44] D.V. Chandra, "Digital image watermarking using singular value decomposition," in *Proc. of 45th IEEE Midwest Symposium on Circuits and System*, pp. 264-267, Tulsa, 2002.
- [45] P. Bas, J. Chassery, and F. Davoine, "Using the fractal code and SVD to watermark images," *Proc. IEEE Int. Conf. on Image Processing*, vol. I, pp. 469-473, Oct, 1998.
- [46] A. Watson, G. Yang, J. Solomon, and J. Villasenor, "Visibility of wavelet quantization noise of watermarked images using SVD", *IEEE Transactions on Image Processing*, vol. 6, no.8, pp. 1164-1175, Aug, 1997.
- [47] Z. Li, W.M. Zheng, and Z.M. Lu: A robust geometry-based watermarking scheme using SVD for 3D meshes. *The first International Conference on Innovative Computing, Information and Control, Beijing*, Vol. II: pp. 166-169, China 2005.
- [48] C.C. Chang, P. Tsai, and C.C. Lin, "SVD-based digital image watermarking scheme", *Pattern Recognition Letter*, pp.1577-1586, 2005.
- [49] X.Z. Zhu, J.M. Zhao, and H.Y. Xu, "A digital watermarking algorithm and implementation Based on improved SVD", in *18th ICPR'06*, Volume 3, pp. 651 – 656, 2006.
- [50] J. Liu, X. Niu, and W. Kong, "Image watermarking based on singular value decomposition," in *Proc. of the Int. Conf. Intelligent Information Hiding and Multimedia Signal Processing*, pp. 457-460, 2006.
- [51] K.S. Kim, M.J. Lee, and H.K. Lee, "Blind image watermarking scheme in DWT-SVD domain," in *IIH-MSP*, Volume 2, pp. 477 – 480, 2007.
- [52] Z. Wang, A. C. Bovik, "Reduced- and No-Reference Image Quality Assessment," *IEEE Signal Processing Magazine*, vol. 28, no. 6, pp. 29-40, Nov. 2011.

- [53] Z. Wang, A. C. Bovik, H. R. Sheikh and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE Transactions on Image Processing*, vol. 13, no. 4, pp. 600-612, Apr. 2004.
- [54] S. Voloshynovskiy, S. Pereira, T. Pun, J. Eggers, and J. Su, "Attacks on digital watermarks: classification, estimation based attacks, and benchmarks," *IEEE Communications Magazine*, vol. 39, no. 8, pp. 118–126, Aug 2001.
- [55] M. Kutter, S. Voloshynovskiy, and A. Herrigel, "The watermark copy attack," in *Proceedings of the IEEE*, pp. 371–380, 2000
- [56] P. Dong, and N. P. Galatsanos, "Affine transformation resistant watermarking based on image normalization," *Proc. IEEE Int. Conf. on Image Processing*, vol. I, pp. 451-455, Oct. 2008.
- [57] K. Anusudha and A. Sangeetha, "Digital Watermarking Of Satellite Images Using Secured Spread Spectrum Technique," *International Journal of Recent Trends in Engineering*, vol. 1, no. 1, pp. 597-599, May 2009.
- [58] A. A. Mohammad, A. Alhaj and S. Shaltaf, "An improved SVD-based watermarking scheme for protecting rightful ownership," *Signal Processing*, vol. 88, no. 9, pp. 2158–2180, Sept. 2008.
- [59] R. Liu and T. Tan, "An SVD-Based Watermarking Scheme for Protecting Rightful Ownership," *IEEE Transactions on Multimedia*, vol. 4, no. 1, pp. 121-128, March 2002.
- [60] D. Shen and H. S. Inp, "Generalized affine invariant image normalization," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 19, no. 5, pp. 431–440, May 1999.
- [61] S. Voloshynovskiy, F. Deguillaume, and T. Pun, "Multibit digital watermarking robust against local nonlinear geometrical distortions," in *Proc. IEEE Int. Conf. Image Processing*, Oct. 2001, vol. 3, pp. 999–1002.
- [62] S. Voloshynovskiy, A. Herrigel, and Y. B. Rystar, "Watermark template attack," in *Proc. SPIE—Security and Watermarking of Multimedia Contents III*, vol. 4314, pp. 394–405, 2001.
- [63] D. Bas and B. Macq, "A method for hiding synchronization marks in scale and rotation resilient watermarking schemes," *Proc. SPIE—Security and Watermarking of Multimedia Contents IV*, vol. 4675, pp. 548–554, Jan. 2002.

- [64] P. Dong and N. P. Galatsanos, "Affine transformation resistant watermarking based on image normalization," in *IEEE Int. Conf. Image Processing*, Rochester, NY, Sep. 2002.
- [65] DIGIMARC Corp. [Online]. Available: <http://www.digimarc.com>.
- [66] P. Bas, J.-M. Chassery, and B. Macq, "Geometrically invariant watermarking using feature points," *IEEE Trans. Image Process.*, vol. 11, no. 9, pp. 1014–1028, Sep. 2002.
- [67] C. Harris and M. Stephen, "A combined corner and edge detector," in *Proc. 4th Alvey Vision Conf.*, vol. 15, pp. 147–151, 1988.
- [68] D. Boneh and J. Shaw, "Collusion-secure fingerprinting for digital data," *IEEE Trans. Inf. Theory*, vol. 44, no. 9, pp. 1897–1905, Sep. 1999.
- [69] A. Nikolaidis and I. Pitas, "Robust watermarking of facial images based on salient geometric pattern matching," *IEEE Trans. Multimedia*, vol. 2, no. 3, pp. 172–184, Sep. 2000.
- [70] C. Rey, K. Amis, J.-L. Dugelay, R. Pyndiah, and A. Picart, "Enhanced robustness in image watermarking using block turbo codes," *Proc. SPIE—Security and Watermarking of Multimedia Contents V*, vol. 5020, pp. 330–336, Jan. 2006.
- [71] J. R. Jain and A. K. Jain, "Displacement measurement and its application in the interframe image coding," *IEEE Trans. Commun.*, vol. 29, no. 12, pp. 1799–1808, Dec. 1981.
- [72] D. Delannay, I. Setyawan, R. Lagendijk, and B. Macq, "Relevant modeling and comparison of geometric distortions in watermarking systems," *Proc. SPIE—Application of Digital Image Processing XXV*, vol. 4790, pp. 200–210, Jul. 2002.
- [73] G. Doërr, C. Rey, and J.-L. Dugelay, "Watermark resynchronization based on elastic graph matching," in *Proc. Int. Conf. Sciences of Electronic, Technologies of Information and Telecommunications*, Mar. 2005.
- [74] M. Lades, J. Vorbrüggen, J. Buhmann, J. Lange, C. Malsburg, R. Würtz, and W. Konen, "Distortion invariant object recognition in the dynamic link architecture," *IEEE Trans. Comput.*, vol. 42, pp. 300–311, Mar. 1993.

- [75] J. L. Dugelay and F. A. P. Petitcolas, "Possible counter-attacks against random geometric distortions," in *Proc. SPIE Conf. Security and Watermarking of Multimedia Content II*, vol. 3971, pp. 338–345, 2000.
- [76] P. Dong, J. Brankov, N. P. Galatsanos, and Y. Yang, "Geometric robust watermarking through mesh model based correction," presented at the IEEE Int. Conf. Image Processing, Rochester, NY, Sep. 2002.
- [77] S. Pereira and T. Pun, "Robust template matching for affine resistant image watermarks," *IEEE Trans. Image Process.*, vol. 9, no. 6, pp. 1123–1129, Jun. 2000.
- [78] M. Kutter, S. K. Bhattacharjee, and T. Ebrahimi, "Towards second generation watermarking schemes," in *Proc. IEEE Int. Conf. Image Processing*, vol. 1, no. 10, pp. 320–323, Oct. 1999.
- [79] V. Solachidis and I. Pitas, "Self-similar ring shaped watermark embedding in 2D DFT domain," in *Proc. Eur. Signal Processing Conf.*, Sep. 2000, vol. 4.
- [80] J. G. Brankov, Y. Yang, and M. N. Wernick, "4D processing of gated SPECT images using deformable mesh modeling," in *the 6th Int. Meeting on Fully Three-Dimensional Image Reconstruction in Radiology and Nuclear Medicine*, Pacific Grove, CA, Oct. 30–Nov. 2 2001.
- [81] Y. Altunbasak and A. M. Tekalp, "Closed-form connectivity-preserving solutions for motion compensation using 2-D meshes," in *IEEE Trans. Image Process.*, vol. 6, , pp. 1255–1269, Sep. 1999.
- [82] K. Aizawa and T. S. Huang, "Model-based image coding: Advanced video coding techniques for very low bit-rate applications," in *Proc. IEEE*, vol. 83, no. 2, pp. 259–271, Feb. 1995.
- [83] Y. Yang, M. Wernick, and J. G. Brankov, "A fast algorithm for accurate content-adaptive mesh generation," *IEEE Trans. Image Process.*, vol. 12, no. 8, pp. 866–881, Aug. 2003.
- [84] F. A. P. Petitcolas, R. J. Anderson, and M. G. Kuhn, "Attacks on copyright marking systems," in *Proc. Workshop Information Hiding*, Portland, pp. 15–17, Apr. 1998.
- [85] E. Koch and J. Zhao, "Towards robust hidden image copyright labeling," in *Proc. IEEE Workshop Nonlinear Signal and Image Processing*, pp. 452–455, Jun. 1995.

-
- [86] S. Qi, G. Zhu, and H.J. Luo, "Image Watermarking Based on Image Normalization," in *Proceedings of IEEE ISCIT2005*, vol. 2 pp. 1453-1457, 2005.
- [87] X.Y. Wang, L.M. Hou, and J. Wu, "A feature-based robust digital image watermarking against geometric attacks," *Image Vision Comput.* vol. 26, pp. 980-989, 2008.
- [88] M. Alghoniemy and A. H. Tewfik, "Geometric Distortion Correction in Image Watermarking," *Proceeding of SPIE on Security and watermarking of Multimedia Contents II*, vol. 3971, pp. 82-89, 2000.
- [89] Rothe, H. Susse and K. Voss, "The Method of Normalization to Determine Invariants," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 18, no. 4, pp. 366-376, 1996
- [90] S. Pei and Lin C. "Image Normalization for Pattern Recognition," *Image Vision Computing*, vol. 15, pp. 711-723, 1995.
- [91] J. Liu, and X. Tan, "Image watermarking based on singular value decomposition," in *Proc. of the Int. Conf. Intelligent Information Hiding and Multimedia Signal Processing*, pp. 457-460, 2003.
- [92] P. Bao and X.H Ma, "Image Adaptive Watermarking Using Wavelet Domain Singular Value Decomposition," *IEEE Trans. On CSVT*, Vol. 15, No. 1, Jan. 2005.
- [93] C. Bergman, and J. Davidson, "Unitary Embedding for Data Hiding with the SVD". *Security, Steganography, and Watermarking of Multimedia Contents VII SPIE*. Vol. 5681, pp. 619-630, 2005.
- [94] X.Z. Zhu, J.M. Zhao, and H.Y. Xu, "A Digital Watermarking Algorithm and Implementation Based on Improved SVD", in the *18th ICPR '06*, vol. 3, pp. 651 – 656, 2006.
- [95] A. Sverdlov, S. Dexter, and A. M. Eskicioglu, "Robust DCT-SVD domain image watermarking for copyright protection: embedding data in all frequencies," in *EUSIPCO2005*, Antalya, Turkey, September 2005.
- [96] F. H. Huang and Z.H. Guan, "A Hybrid SVD-DCT Watermarking Method Based on LPSNR", *Pattern Recognition Letter*, pp.1769-1775, 2004.

-
- [97] K. L. Chung, W. N. Yang and Y. H. Huang, "One SVD-based watermarking algorithm," *Applied Mathematics and Computation*, vol. 188, issue 1, pp. 54-57, 2007.
 - [98] C. C. Chang, C. C. Lin and Y. S. Hu, "An SVD oriented watermark embedding scheme with high qualities for the resorted images," *International Journal of Innovative Computing, Information and Control*, vol.3, no.3, pp.609-620, 2007.
 - [99] H.C. Andrews and C.L. Patterson, "Singular Value Decompositions and Digital Image Processing," *IEEE Trans. ASSP*, pp. 26-53, Feb. 1976.
 - [100] M. Alghoniemy and A. H. Tewfik, "Image Watermarking by Moment Invariants," *IEEE International Conference on Image Proceedings*, vol. 2, pp. 73-76, 2005.
 - [101] L.D. Li and B.L. Guo "Rotation, scaling and translation invariant image watermarking using feature points", *Journal of China Universities of Post and Telecommunications*, vol.15, issue 2, pp.82-87, 2008.
 - [101] M. Kutter, S. K. Bhattacharjee, and T. Ebrahimi, "Toward second generation watermarking schemes," in *Proc. IEEE Int. Conf. on Image Process.*, vol. 1, pp. 320-323, 1999.
 - [102] C. Harris and M. Stephen, "A combined corner and edge detector", in *4th Alvey Vision Conf.*, pages 147-151, 1988.
 - [103] K. Mikolajczyk and C. Schmid, "Scale and Affine invariant interest point detectors", *International Journal of Computer Vision*, vol. 6 pp. 63-86, 2004.
 - [104] J. P. Antoine, P. Vandergheynst, and R. Murenzi, "Two-dimensional directional wavelets in image processing", *International Journal of Imaging System and Technology*, vol. 10, pp. 152-165, 1996.
 - [105] M. Tone and N. Hamada, "Affine Invariant Digital image watermarking using feature points", *International Workshop on Nonlinear Circuit and Signal Processing*, Hawaii, USA, Mar. 2005.
 - [106] T. Lindeberg. "Feature detection with automatic scale selection", *International Journal of Computer Vision*, vol. 30, pp. 79-116, 1998.
 - [107] K. Mikolajczyk and C. Schmid, "Indexing based on scale invariant interest points", in *Proceedings of the 8th International Conference on Computer Vision*, Vancouver, Canada, pp. 525-531, 2001.

- [108]Xiang-yang Wang, Li-min Hou and Jun Wu, “A feature-based robust digital image watermarking against geometric attacks,” *Journal of Image and Vision Computing*, Volume 26, Issue 7, Pages 980-989, Jul. 2008.
- [109]Pan-Pan Niu, Xiang-Yang Wang, Hai-Bo Jin and Ming-Yu Lu, “A feature-based robust digital image watermarking scheme using bandelet transform” *Optics & Laser Technology*, Volume 43, Issue 3, Pages 437–450, Apr. 2011.
- [110]H.Y. Lee, I. K. Kang, H.K. Lee and Y.H. Suh, “Evaluation of Feature Extraction Techniques for Robust Watermarking”, *IWDW 2005, LNCS 3710*, pp. 418-431, 2005.
- [111]D. Lowe, “Distinctive image features from scale invariant keypoints”, in *International Journal of Computer Vision* vol. 2(60), pp. 91-110, 2004.
- [112]A. Khotanzad and Y. H. Hong, “Invariant image recognition by Zernike moments,” *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 12, pp.489–497, May 1990.
- [113]H.S. Kim and H.K. Lee, “Invariant Image Watermark using Zernike Moments,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol.13, No.8, pp.766-775, August 2003.
- [114]M. K. Hu, “Visual pattern recognition by moment invariants,” *IRE Trans. Inform. Theory*, vol. 8, pp.179- 187, February 1962.
- [115]X.Y. Wang, T.X. Ma, and P.P. Niu, “Digital Audio Watermarking Technique Using Pseudo-Zernike Moments,” in *the Proceedings of ICICS 2009*, pp. 459-473, 2009.
- [116]X.Y. Wang and L.M. Hou, “A new robust digital image watermarking based on Pseudo-Zernike moments,” *Journal Multidimensional Systems and Signal Processing*, vol. 21, pp. 313-321, 2010.
- [117]Q. Chen, X.L Yang and J.Y. Zhao, “Robust image watermarking with Zernike moments”, in *Proceeding of IEEE CCECE/CCGEI*, vol. 21, pp. 1256-1260, May 2005.
- [118]T. Xia, H.Q. Zhu and H.Z. Shu “Image description with generalized pseudo-Zernike moments” *Journal of the Optical Society of America*, vol. 24, pp. 156-186, 2007.
- [119]X.Y. Wang, T.X. Ma, and P.P. Niu, “A pseudo-Zernike moment based audio watermarking scheme robust against desynchronization attacks,” *Computers & Electrical Engineering*, vol. 37, pp. 425-443, 2011.

- [120]C.H. Li and Z.Y. Lu, “Desynchronization attack on digital watermarks and their countermeasures,” *Journal of Image and Graphics*, vol. 10, no. 4, pp. 403-409, 2005.
- [121]V. Licks and R. Jordan, “Geometric attacks on image watermarking system,” *IEEE Trans on Multimedia*, vol. 1, no.3, pp. 68-78, 2005.
- [122]C. W. Tang and H. M. Hang, “A feature-based robust digital image watermarking scheme,” *IEEE Transactions on Signal Processing*, vol. 51, no. 4, pp. 950-958, 2003.
- [123]X.Y. Wang, L. Meng, and H.Y. Yang, “Geometrically invariant color image watermarking scheme using feature points,” *Science in China Series F: Information Sciences* vol. 52pp. 1605-1616, 2009.
- [124]J.G. Sun and W. He, “RST invariant Watermarking Scheme based on SIFT feature and pseudo-Zernike Moment,” *Second International Symposium on Computational Intelligence and Design*, pp. 10-13, 2009.
- [125]Y. Xin, S. Liao, and M. Pawlak, “A multibit geometrically robust image watermark based on Zernike moments,” in *IEEE Int. Conf. on Pattern Recognition*, vol. 4, pp. 861–864, Aug. 2004.
- [126]L.D. Li, B.L. Guo, and K. Shao, “Geometrically robust image watermarking using scale-invariant feature transform and Zernike moments” *Chinese Optics Letters*, vol. 5, Issue 6, pp. 332-335, 2007.
- [127]C.W. Tang and H.M. Hang, “A Feature-Based Robust Digital Image Watermarking Scheme,” *IEEE Transaction on Signal Processing*, vol. 51, no. 4, April 2003.
- [128]Joseph J.K., O. Ruanaidh and Thierry Pun, “Rotation, Scale and Translation Invariant Digital Image Watermarking,” *Signal Processing*, 21 Aug, 1997, pp.1-19.
- [129]F. Hartung, J. Su, and B. Girod, “Spread-spectrum watermarking: Malicious attacks and counterattacks”, in *Security and Watermarking of multimedia contents*, SPIE, San-Jose CA, USA, January 1999.
- [130]R. Dugad, K. Ratakonda, and N. Ahuja, “A new wavelet-based scheme for watermarking images,” in *Proceedings of the IEEE International Conference on Image Processing, ICIP 1998*, Chicago, IL, USA, October 1998.
- [131]E. Ganic and A. M. Eskicioglu, “A DFT-Based Semi-Blind Multiple Watermarking Scheme for Images, *4th New York Metro Area Networking Workshop*, The Graduate Center, The City University of New York, Sept. 10, 2004.”

-
- [132]R. Ridzon and D. Levicky, "Robust digital watermarking in DFT and LPM domain," *50th International Symposium ELMAR-2008*, Zadar, Croatia, pp. 651-655, Sept. 2008.
- [133]D. Zheng, Y. Liu, and J.Y. Zhao, "RST invariant digital image watermarking based on a new phase-only filtering method," *Signal Processing, Elsevier*, vol. 85, pp. 2355-2371, 2005.
- [134]R. Ridzon and D. Levicky "Log-polar Mapping in Robust Digital Image Watermarking," *Radioelektronika, 2007. 17th International Conference*, pp. 1-4, 2007.
- [135]D. Zheng, J. Zhao, and A. El Saddik, "RST-invariant digital image watermarking based on Log-polar mapping and phase correlation", *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 13, no. 8, pp. 753–765, Aug. 2003.
- [136]C. Lin, M. Wu, J. Bloom, I. Cox, and M. Miller, "Rotation, Scale, and Translation Resilient Watermarking for Images," *IEEE Trans. on Image Proc.*, vol.10, no.5, pp. 767-82, 2001.
- [137]D. Zheng and J. Zhao, "LPM-based RST invariant digital image watermarking," *IEEE CCECE Canadian Conference on Electrical and Computer Engineering*, vol. 3, pp. 1952-1956, May 2003.
- [138]S. Pereira, F. Deguillaume, G. Csurka and T. Pun, "Template based recovery of Fourier-based watermarks using log-polar and log-log maps," *IEEE International Conference on Multimedia Computing and Systems*, vol. 1, pp. 870-874, 1999.
- [139]W.W. Fung and A. Kunisa, "Rotation, scaling, and translation-invariant multi-bit watermarking based on log-polar mapping and discrete Fourier transform," *IEEE International Conference on ICME*, vol. 10, pp. 158-161, 2005.