



uOttawa

L'Université canadienne
Canada's university

FACULTÉ DES ÉTUDES SUPÉRIEURES
ET POSTDOCTORALES



FACULTY OF GRADUATE AND
POSTDOCTORAL STUDIES

Huiyan Qi

AUTEUR DE LA THÈSE / AUTHOR OF THESIS

M.A.Sc. (Electrical Engineering)

GRADE / DEGRÉ

School of Information Technology and Engineering

FACULTE, ÉCOLE, DÉPARTEMENT / FACULTY, SCHOOL, DEPARTMENT

Human Visual System Based Adaptive Digital Image Watermaking

TITRE DE LA THÈSE / TITLE OF THESIS

J. Zhao

DIRECTEUR (DIRECTRICE) DE LA THÈSE / THESIS SUPERVISOR

CO-DIRECTEUR (CO-DIRECTRICE) DE LA THÈSE / THESIS CO-SUPERVISOR

EXAMINATEURS (EXAMINATRICES) DE LA THÈSE / THESIS EXAMINERS

A. El Saddik

P.X. Liu

Gary W. Slater

Le Doyen de la Faculté des études supérieures et postdoctorales / Dean of the Faculty of Graduate and Postdoctoral Studies

Human Visual System Based Adaptive Digital Image Watermarking

Huiyan Qi

A thesis submitted to
the Faculty of Graduate and Postdoctoral Studies
in partial fulfillment of
the requirements for the degree of
Master of Applied Science in Electrical Engineering

Ottawa-Carleton Institute of Electrical and Computer Engineering
School of Information Technology and Engineering
University of Ottawa

Ottawa, Ontario, Canada

May, 2006



Library and
Archives Canada

Bibliothèque et
Archives Canada

Published Heritage
Branch

Direction du
Patrimoine de l'édition

395 Wellington Street
Ottawa ON K1A 0N4
Canada

395, rue Wellington
Ottawa ON K1A 0N4
Canada

Your file *Votre référence*
ISBN: 978-0-494-18460-8
Our file *Notre référence*
ISBN: 978-0-494-18460-8

NOTICE:

The author has granted a non-exclusive license allowing Library and Archives Canada to reproduce, publish, archive, preserve, conserve, communicate to the public by telecommunication or on the Internet, loan, distribute and sell theses worldwide, for commercial or non-commercial purposes, in microform, paper, electronic and/or any other formats.

The author retains copyright ownership and moral rights in this thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without the author's permission.

AVIS:

L'auteur a accordé une licence non exclusive permettant à la Bibliothèque et Archives Canada de reproduire, publier, archiver, sauvegarder, conserver, transmettre au public par télécommunication ou par l'Internet, prêter, distribuer et vendre des thèses partout dans le monde, à des fins commerciales ou autres, sur support microforme, papier, électronique et/ou autres formats.

L'auteur conserve la propriété du droit d'auteur et des droits moraux qui protègent cette thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.

In compliance with the Canadian Privacy Act some supporting forms may have been removed from this thesis.

Conformément à la loi canadienne sur la protection de la vie privée, quelques formulaires secondaires ont été enlevés de cette thèse.

While these forms may be included in the document page count, their removal does not represent any loss of content from the thesis.

Bien que ces formulaires aient inclus dans la pagination, il n'y aura aucun contenu manquant.


Canada

Abstract

It is known that the fidelity of the image is inversely proportional to the robustness of the watermark. Therefore, there should be the trade off between fidelity for robustness and vice versa. Based on the new spatial masking we explored in this thesis, a new adaptive digital image watermarking method is proposed. It keeps the invisibility of the watermark and maintains its robustness at the same time. The new spatial masking is built according to the image features such as flat areas with big brightness or darkness, edges, and regions with high activities. With the same watermarking energy, the quality of watermarked image with this masking is much better than the one without it. We also propose using the weighted Peak Signal-to-Noise Ratio (wPSNR) to evaluate the image quality. The watermark is detected by the key-dependent method without knowing the original image information. In addition, we extend this proposed spatial masking to the Discrete Cosine Transform (DCT) domain by using the method of searching the extreme value of the quadratic function subject to the bounds on the variables.

Acknowledgement

First of all, I would like to express my deep gratitude to my supervisor, Dr. Jiying Zhao. His excellent guidance, profound knowledge, stimulating suggestion, immense help, and constant encouragement help me throughout the graduate studies and completing the thesis.

Secondly, I am also grateful to the members of the Multimedia Communications Research Laboratory. The useful discussion also helps me to develop the research work.

Finally, I would like to give my special thanks to my husband, my parents and my brother, who always give me continuous support.

Acronyms

BCH	Bose-Chaudhuri-Hocquenghem Coding
CDMA	Code Division Multiple Access
CSF	Contrast Sensitivity Function
DCT	Discrete Cosine Transform
DFT	Digital Fourier Transform
DVD	Digital Versatile Disc or Digital Video Disc
DWT	Digital Wavelet Transform
HAS	Human Auditory System
HVS	Human Visual System
IDCT	Invert Discrete Cosine Transform
IIR	Infinite Impulse-Response
JND	Just Noticeable Difference
MAP	Maximum A posteriori Probability
MP3	MPEG Audio Layer 3
MRF	Markov Random Field
MSE	Mean Square Error
NVF	Noise Visual Function
PSNR	Peak Signal-to-Noise Ratio
QIM	Quantization Index Modulation
wPSNR	weighted Peak Signal-to-Noise Ratio

Contents

1	Introduction	1
1.1	Background	1
1.2	Digital Watermarking System	3
1.2.1	Concept of the Digital Watermarking System	3
1.2.2	Properties of Watermarking Systems	6
1.2.3	Categories of the Watermark	7
1.3	Thesis Objectives	9
1.4	Thesis Contribution	10
1.5	Publication Generated from Thesis Work	11
1.6	Thesis Organization	11
2	Literature Review	13
2.1	Human Visual System	13
2.1.1	Introduction	13
2.1.2	HVS Masking	14

2.1.3	Previous HVS Masking	18
2.2	Evaluation of the Watermarked Image	22
2.3	Watermark Detection	25
2.4	Data Hiding Application	26
2.5	DCT Domain Masking	28
3	Proposed Adaptive Digital Watermarking Method	31
3.1	Proposed HVS Masking	31
3.1.1	Luminance Masking	32
3.1.2	Texture Masking	35
3.1.3	Edge Masking	36
3.1.4	Final Masking	37
3.2	Watermark Embedding	38
3.3	Evaluation of the Watermarked Image	39
3.4	Watermark Detection	39
3.5	Data Hiding in Spatial Domain	42
3.6	Extend Proposed Spatial Domain Masking to DCT Domain	44
4	Implementation and Results Analysis	49
4.1	HVS Masking	49
4.1.1	Luminance Masking	49
4.1.2	Texture Masking	53

4.1.3	Edge Masking	53
4.1.4	Final Masking	58
4.2	Watermark Embedding	59
4.3	Quality Measurement	67
4.4	Watermark Detection	72
4.5	Data Hiding Application	75
4.6	HVS Masking Extended to the DCT Domain	80
5	Conclusion and Future Work	85

List of Figures

1.1	Watermarking System	4
2.1	Frequency Sensitivity Matrix	29
3.1	HVS Masking	32
4.1	Luminance Masking	50
4.2	Original Image “Lena”	52
4.3	Normalized Luminance Masking of Image “Lena”	52
4.4	Texture Masking of Image “Lena”	53
4.5	Edge Masking Created by Un-Sharp Filter	54
4.6	The Image “Lena” after the Laplacian Filter	55
4.7	Edge Detection of Image “Lena”	56
4.8	Dilated Edge Detection of Image “Lena”	57
4.9	Final HVS Masking of Image “Lena”	59
4.10	Original Image of “Lena”	60
4.11	Watermarked Image “Lena” without Masking	61

4.12	Watermarked Image “Lena” with NVF (Non-Stationary Model) $S = 6.4$	61
4.13	Watermarked Image “Lena” with Proposed Masking	62
4.14	Original Image of “Work”	63
4.15	Watermarked Image “Work” Without Masking	63
4.16	Watermarked Image “Work” With NVF (Non-Stationary Model) $S=13.1$	64
4.17	Watermarked Image “Work” With Proposed Masking	64
4.18	Original Image “Discussion”	65
4.19	Watermarked Image “Discussion” Without Masking	66
4.20	Watermarked Image “Discussion” With NVF (Non-Stationary) $S=11.6$	66
4.21	Watermarked Image “Discussion” With Proposed Masking	67
4.22	Block Value Changed On the Shoulder of Image “Lena”	68
4.23	Block Value Changed On the Background With Block Color of Image “Lena”	69
4.24	Evaluation Results of PSNR and wPSNR of Compressed Image	70
4.25	wPSNR Value of Watermarked Images with Different Masking	71
4.26	Watermarking Detection	72
4.27	Detection Value for the Watermarked Image Passed Compression	74
4.28	Detection Value of Watermarked Image Passed Noise Attack	75
4.29	Watermarked Image “Lena” in Data Hiding Application	79
4.30	Watermarked Image “Discussion” in Data Hiding Application	79
4.31	Watermarked Image “Work” in Data Hiding Application	80

4.32 Watermarked Image “Lena” with DCT Domain Masking	81
4.33 DCT Domain Watermark of Image “Lena”	81
4.34 Watermarked Image “Discussion” with DCT Domain Masking	82
4.35 DCT Domain Watermark of Image “Discussion”	82
4.36 Watermarked Image “Work” with DCT Domain Masking	83
4.37 DCT Domain Watermark of Image “Work”	83

List of Tables

3.1	Parameter G_1 in the Luminance Masking Calculation	33
3.2	Parameter G_2 in the Luminance Masking Calculation	34
3.3	Parameter G_3 in the Luminance Masking Calculation	34
3.4	Parameter G_4 in the Luminance Masking Calculation	34
3.5	Parameter B in the Luminance Masking Calculation	35
4.1	wPSNR Value of the Quality of Different Watermarked Images	72

Chapter 1

Introduction

1.1 Background

The growth of the digital multimedia technology and the successful development of the Internet allow people to process, deliver and store information more easily. People can not only share or exchange the information via email, but also download, manipulate and save an image, video or MP3 music. This kind of advantage also raises the issue of how to protect the copyright ownership, since the Internet is quite open and the security is always a challenge. Traditionally, the information hiding approach, such as cryptography, is able to deal with this problem. Before the data is transmitted, the cryptography uses the symmetric or asymmetric key and the hash function which helps to scramble the message in order to prevent others from decoding it. The cryptography only allows the authorized key-holders to decrypt the data. However, this method depends very much on the image format. If the image format has changed during the transmission, the encrypted information will be removed, consequently, this methods is out of the function. In addition, once the message is decrypted, it is quite difficult to prevent it from being reproduced or retrans-

mitted. Therefore, there is no way to do the further protection. Another technology named steganography regards the multimedia data as a carrier, and hides the information in the multimedia data itself and still keeps its fidelity. Therefore, it can reach the goal of keeping the message secret even after the decryption. However, the steganography technology counts on a secure communication environment such as point to point communication which is unknown to third parties [1]. As a result, it is not robust to the various attacks.

Watermarking technology has evolved from steganography. Different from steganography, the message embedded via the watermark is more related to the multimedia data and used as an identity, while the message embedded by steganography sometimes has no relationship to the data itself [1, 2]. In addition, the watermarking technology pays more attention to the robustness against the attack than the steganography.

Since the middle of nineties, the researches into digital watermarking have spread to a large range of applications.

- **Copyright Protection**

This is the most useful application served by watermarking technology. It embeds the unambiguous information which represents the copyright of the ownership into the images, video or audio in order to keep other parties from claiming the possession of the data. In this case, the robustness against different malicious attacks is the important requirement of the watermark. Besides the unique information about the owner, it also embeds the other information to show the origin, status, or destination of the data, and help to identify the ownership once there is copyright violation.

- **Copy Control**

Another classical scenario of using watermark is copy control, such as the protection of the DVD video. The mechanism behind it is that the distinct watermark is embedded in the DVD content to distinguish the legal copy from the illegal copy. There are two kinds of players: one is the copyright compliant player, and the other one is the non-copyright complaint player. When the DVD is being played, the copyright complaint player tries to find the location of the watermark and verify it. After the watermark is confirmed, the DVD will be played normally. Furthermore, the message of watermark sometimes includes: copy-freely, copy-never or copy-once. With different restrictions, it will reach the goal to prohibit the flow of pirate copy.

- **Authentication**

In these days, it is quite convenient for people to use Adobe Photo-shop tools to edit an image without leaving any perceptible traces. When the original data is very important, a watermark is the preferable solution to find out whether the initial data content has been tampered or not. Contrary to the scenarios mentioned before, this watermark is usually fragile or semi-fragile. The main purpose is that when the original data has been changed, the watermark bounded into data will also be damaged. Therefore, based on the results of verifying the watermark, it can detect if the data is original or not.

1.2 Digital Watermarking System

1.2.1 Concept of the Digital Watermarking System

Generally, a digital image watermarking system is similar to a communication system composed by three main elements: a watermark embedder, a communication channel and

a watermark detector. Usually, the embedder and the detector will be more attractive to the researchers. Of course, they also take the properties of the communication channel into account during the design of the digital watermarking system. The whole system is shown in the Figure 1.1.

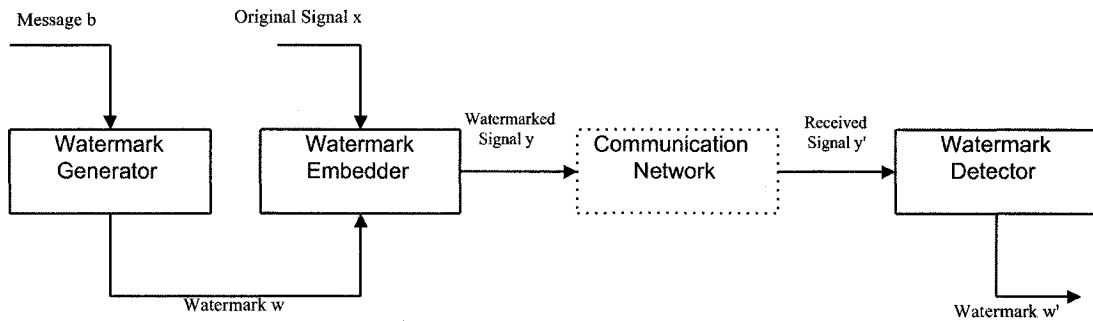


Figure 1.1: Watermarking System

The watermark embedder includes two inputs and one output. The two inputs are the original signal x and the watermark denoted as

$$w = \{w_1, w_2, w_3, \dots, w_m\} \quad (1.1)$$

The original signal, which is also called host signal, can be the original image, video or audio file in which we wish to embed the watermark. The watermark is usually the output from the watermark generator which does the information coding. In many watermarking systems, the information b which is to be hidden is denoted as

$$b = \{b_1, b_2, b_3, \dots, b_m\}, \quad b \in [0, 1] \quad (1.2)$$

and will not be embedded into the original signal directly. Prior to being embedded, the information message b is first converted into a watermark w via adopting different communication technologies. The popular one is the spread spectrum technology which is famous for its defense against noise attacks and is commonly used in CDMA. Through the spread spectrum sequence, the message is transformed into bipolar signals in which the ones are mapped into -1 and zeros are indicated by +1. Then this long bit stream is regarded as the watermark to be embedded in the original signal. In addition, a chaotic sequence is another choice to generate the watermark. The most attractive feature of a chaotic sequence is that it is quite good against geometric attacks [3, 4] and suitable for multiple watermark systems [5]. In addition, the information coding sometimes adds an encryption function to increase the security of the watermark. The public key or the private key is indispensable. Therefore, the algorithm of the embedder can be expressed as the following:

$$y = F(x, w, k) \quad (1.3)$$

where, y is the watermarked image, w is the watermark to be embedded, k represents the public or private key, x is the original image. After the watermark is embedded into the host signal, the watermarked signal is the output of the embedder and transmitted to the communication network.

At the watermark detector site, two tasks are accomplished after the detector receives the signal. Firstly, it needs to detect whether the received signal has the watermark or not. Secondly, the detector judges whether the watermark is real or not. Based on whether the detector needs the original signal or not, the detector can be divided into the blind detector which has no knowledge of the original signal and the non-blind detector which should

have original at hand. The function of the detector is indicated as follows:

$$w' = F'(y', k', x) \quad \text{or} \quad w' = F'(y', k') \quad (1.4)$$

where, w' is the detected watermark, y' is the received signal, k' is the public or private key, x is the original signal.

1.2.2 Properties of Watermarking Systems

Depending on the different digital watermarking application environments, the requirements of the watermark is not the same. Some important and most common properties are listed as follows:

- **Robustness**

The concept of robustness of a watermark refers to the capability of the hidden message to be detected after the manipulation or distortion of the host signal, including non-malicious signal processing and a malicious attack [2]. Non-malicious signal processing, such as lossy compression, does not intend to remove the watermark or destroy it. On the contrary, malicious processing tries to damage the watermark and negate its function.

- **Imperceptibility**

Making the watermark amplitude as high as possible makes it robust but degrades the quality of the watermarked signal. In order to keep the fidelity of the watermarked signal, the transparency of the watermark is quite important. Therefore, there is a need to decrease the strength of the hidden message. As a result, a tradeoff between robustness and imperceptibility is inevitable.

- **Capacity**

There is no clear definition of watermarking capacity. Usually, capacity refers to the channel capacity which is used in the communication. Some watermarking algorithms still borrows this concept [6, 7]. As for the image, it is quite difficult to divide the channel based on the pixels. Therefore, the common concept of the capacity of the watermark describes the number of bits of the hidden message or the data payload in the whole original signal. Of course, based on the application, the capacity of the watermark varies.

There is no doubt that the robustness, imperceptibility and capacity can not each be maximum at the same time. Once one of them is fixed, the other two are inversely proportional. For example, in order to detect a watermark without error, it needs longer information coding and more embedding strength. Obviously, the cost is the information quantity and the imperceptibility.

- **Security**

The security of the watermark increases the ability to resist malicious attacks involving unauthorized removal, and detection. It borrows the idea of cryptography to have a secret key held by the embedder and detector. The key can be private or public, single or multiple. In this way, the watermark can neither be detected nor read by unauthorized users. The risk of being forgery or other illegal activities is reduced.

1.2.3 Categories of the Watermark

According to the location of the watermark, the embedding and detection methods, the watermark is sorted into different categories.

- **Spatial/Time Domain and Transform Domain Watermark**

If the original signal is an image, the term spatial domain watermarking is used. It refers to modulating the watermark onto the spatial image pixels. If the original signal is audio, the time domain watermark is used to indicate that the watermark is embedded in the audio signal itself. Those two are similar to amplitude modulation in communication technology. Transform domain watermarking casts the watermark in the DFT (Digital Fourier Transform), DCT (Digital Cosine Transform) and DWT (Digital Wavelet Transform) domain. Usually, it only changes the amplitude of the frequency and leaves the phase untouched. Sometimes, in the application where it needs to be robust against the geometric attack, the watermark is also cast onto the phase value.

- **Additive and Multiplicative Watermark**

The most popular watermark embedding method is the additive watermark. The following equation shows the relationship of the watermark and the original signal.

$$y = x + \alpha w \quad (1.5)$$

where y is the watermarked signal, x is the original signal, and w is the watermark, α controls the embedding strength. The watermark can be detected via calculating the value of the correlation between the watermark and the received signal.

Another method is the multiplicative watermark, the equation is as follows:

$$y = x + \alpha wx \quad (1.6)$$

The multiplicative watermark is often used in the frequency domain where the masking of the frequency sometimes is the function of the value of the frequency itself. It should

be pointed out that the detection and decoding of this watermark is more complex and difficult than the additive one.

- **Blind and Non-Blind Detection Watermark**

The blind watermark recovery suggests the detector does not need the original signal. Conversely, the non-blind watermark recovery requires that the detector does have the original signal. Normally, the availability of the original signal allows the extraction much easier, compared with the blind detection, especially when the watermarked signal has suffered from the rotation or scaling geometric attacks. However, this requires the transmission of much more data than the blind detection. This issue limits the non-blind detection algorithm in many practical scenarios.

1.3 Thesis Objectives

In order to resist the normal signal processing and other different attacks, we wish the embedding strength to be as high as possible. However, because the watermark directly affects the original signal, it is obvious that the higher the embedding strength, the lower the quality of the watermarked signal. Therefore, the tradeoff between the robustness and the imperceptibility is inevitable. A balance is achieved by taking the concealment principles denoted by the Human Visual System (HVS) or Human Auditory System (HAS) which takes the human senses into account when embedding the watermark. In this way, the strength of the watermark is adapted to the features of the original signal to guarantee the maximum possible imperceptivity of the watermark. This thesis concentrates on the adaptive watermarking of the image. The objectives of this thesis are as follows:

- Analyze the effect of human visual system to the digital image watermark;

- Design and describe the adaptive digital watermarking method to balance the robustness and imperceptibility of the watermark;
- Evaluate the proposed adaptive digital watermarking approach;
- Detect the watermark without the knowledge of the original image;
- Try to extend the applications of the proposed adaptive digital watermarking method.

1.4 Thesis Contribution

This thesis presents a new HVS based adaptive digital watermarking approach, the main contributions are listed as follows:

- A new HVS based adaptive spatial domain additive watermarking method is proposed. The new spatial masking is built according to the image features such as flat areas with brightness and darkness, edges, and areas with high activity. With the same watermarking energy, the quality of watermarked image with the proposed masking is improved compared with the one without it.
- We also point out that PSNR can not distinguish between the quality of two images with the same watermarking energy and propose to use the wPSNR to evaluate the watermarked image quality.
- The watermark is detected by the key-dependent method. The results of experiments show that it is robust to common operations such as compression and noise attack.
- The new HVS masking can also be used in the other watermarking algorithms where it is needed to consider the fidelity and robustness at the same time.

- We extend this new spatial masking to the DCT domain by using the method of searching for the extreme value of the quadratic function subject to the bounds on the variables.

1.5 Publication Generated from Thesis Work

- Huiyan Qi and Jiying Zhao, “Human Visual System Based Adaptive Digital Image Watermarking”, Canadian Conference on Electrical and Computer Engineering (CCECE 06), pp. 985-988, Ottawa, Canada, May 7-10, 2006.

1.6 Thesis Organization

In Chapter 2, the human vision system will be introduced. It will describe the mechanism of the human vision system, the calculation of the human vision sensitivity threshold, and the criteria of the human vision system masking. It also reviews the previous proposed types of masking and analyzes the advantages and disadvantages of each masking type. Moreover, it will introduce the image quality measurement based on the PSNR which is the standard measurement of the image quality, and concludes that the PSNR cannot distinguish the difference between the quality of two images when both images contain the same watermarking energy. This chapter will also analyze some previous proposed wPSNR methods. Different watermark detection methods will be reviewed in this chapter. In addition, previous data hiding methods and the DCT domain masking will also be reviewed.

In Chapter 3, the proposed HVS based adaptive digital watermarking method will be introduced step by step. It will also introduce the proposed wPSNR and watermarking detection

methods based on the noise shaping technology. It will also introduce how to merge this masking into other watermarking algorithms. Finally, it will describe the method to extend this masking into the DCT domain.

Chapter 4 will detail the HVS masking method, the evaluation of the watermarked image with proposed wPSNR, and the experimental results of the watermark detection method. It will give an example of data hiding by using this method and show the results of the DCT domain masking which will be done by the method proposed in Chapter 3.

Chapter 5 will draw the final conclusion and express our future work.

Chapter 2

Literature Review

2.1 Human Visual System

2.1.1 Introduction

In order to find the threshold where the watermark is perceptible in the image, it is important to understand the Human Visual System (HVS), since the final watermarked image will be examined by human eyes. The human visual system is one of the most complex biological systems and describes the mechanism of how people process vision. Based on the light impacting on the retina, the brain receives a model of the outside world. There are three stages of the human vision: encoding, representation and interpretation [8, 9]. The function of the first stage, called encoding, is translating the light into electrical signals via the photoreceptors in the retina. The following stage, named representation, tunes the encoded signal into unambiguous characteristics by different visual pathways. The final stage, denoted as interpretation, integrates and senses the signals from the representation and gives the impression of the objects outside world.

For many years, people have been studying and understanding human visual system and simulating their function in order to help develop good quality vision equipment such as cameras, projectors and video cameras. There are many HVS models recommended, based on the experiments and applications. Because the chromatic channel and the luminance channel are treated separately in the HVS model, in this thesis, we will only concentrate on the gray scale images.

Researchers have found that many factors are responsible for the limited sensitivity of human vision. The surface of the cornea causes refraction, the circular entrance of the pupil causes diffraction, the optical lens have chromatic aberration effects, and the mosaic of photoreceptors only does spatial sampling process [8]. The saturation effect of human vision is another feature which limits the light wavelength where the eyes can react. These limitations and deficiencies imply that human vision does not respond to small stimuli very well, and it can only differentiate between signals below a certain threshold. However, this perceptive limitation gives us a good opportunity to embed a watermark in the image.

2.1.2 HVS Masking

- **Criteria of HVS Masking**

HVS masking is a phenomenon which describes interactions among visual signals or stimuli and their non-linear effect. The effect is that a stimulus that is noticeable by itself may become invisible when there is another higher level stimulus with similar characteristics. Therefore, the masking is actually the ability that one signal covers another signal. The amount of the masking relies on the visibility threshold. If the HVS masking is used in a watermarking system to control the embedding strength, which is shown in the previous Equ.(1.5), it will make the watermark adapt to the human visual system.

In order to determine the visibility threshold of human vision, many experiments are set up and many statistical methods are used. Before reviewing different variations of HVS masking, some basic terminology is introduced as follows [10]:

- Reflectance is the proportion of visible light reflected by the object surface. Usually, the object can absorb part of the light and reflect part of it. Dim objects absorb more light than bright ones. While bright objects reflect more light than dim ones do.
- Luminance is the amount of the light reflected by the object that is recorded by the retina. The sensitivity of eyes is not equal but variable according to the frequency of the light.
- Brightness or Darkness is the perceived reflectance.

Mauro Barni [10] did image watermarking experiments in which he kept on adding noise to the image in the spatial domain and tried to understand the sensitivity of human vision. Eventually, he summarized the observations and experiments and gave the following three rules [10], which are the foundation of the new spatial masking in this thesis:

1. Disturbances are much less visible on highly textured regions than on uniform area;
2. Contours are more sensitive to noise addition than highly textured regions but less than flat areas;
3. Disturbances are less visible over dark and bright regions.

- **Threshold of Visibility**

Generally, there are two methods to measure the visibility threshold: one is called Just Noticeable Difference (JND) and the other one is the Contrast Sensitivity Function (CSF).

The JND threshold is defined as the magnitude of the stimulus at which it becomes just visible or just invisible [11]. Weber's law is the first one to introduce the JND threshold. His opinion is that the reaction of human vision depends more on the ratio of local luminance difference to the surrounding luminance than on the absolute luminance value. He also points out that if the luminance of a stimulus is just noticeable from the surrounding luminance, the ratio between the just noticeable difference ΔL_{jn} and the background luminance L is almost constant [10, 12].

$$\frac{\Delta L_{jn}}{L} = 0.02 \quad (2.1)$$

According to Weber's law, the just noticeable difference is changed along with the luminance background in a proportional relationship. In practice, it is true that when the background luminance is increased, the just noticeable difference is also increased. However, when the background luminance is decreased below a certain threshold, the just noticeable difference will increase instead of decreasing. This is caused by the human vision saturation effect. Weber's law did not indicate this issue.

Different from Weber's law, Michelson used global luminance to define the contrast sensitivity function. The equation is as follows [13, 14]:

$$C = \frac{L_{max} - L_{min}}{L_{max} + L_{min}} \quad (2.2)$$

where, C is the contrast value, L_{max} and L_{min} are the maximum and minimum luminance respectively. This contrast equation uses the global luminance instead of the local luminance. This equation is not suitable when some very bright or very dark pixels appear in the image, since it can not determine the whole image contrast correctly.

Most contrast sensitivity functions are used in the frequency domain because most experiments are done in that domain. Peli [14] was the first person who indicated that human vision is sensitive to the local contrast instead of the global contrast of the whole image. In the FFT domain, he divided frequencies into different bands, and defined the local band limited contrast, which is the function of spatial frequency, as the ratio of the average luminance of the image filtered by the band-pass filter and the low-pass filter respectively [14, 15]. The low-pass filter is just below the band-pass filter. The equation is as follows:

$$C = \frac{\Psi_m * I}{\Phi_{m-1} * I} \quad (2.3)$$

where, C is the contrast value, Ψ_m is a band-pass filter at band m , Φ_{m-1} is related low pass filter just below band m . The operation $*$ is the convolution. In practice, this equation has been proved to be very good at analyzing image contrast [13, 14].

Winkler also provided another contrast calculation. He defined an isotropic local contrast as the contrast of frequency band at level j which is equal to the square root of the total energy over all orientations of band limited filtered image, and normalized by the corresponding low pass filtered image [13, 15].

$$C_j = \frac{\sqrt{2\sum_k |\Psi_{jk} * I|^2}}{\Phi_j * I} \quad (2.4)$$

where, C_j is the local contrast at frequency band level j , Φ_j is the low pass filter, I is the original image, Ψ_{jk} is the different orientation band pass filter at the level j .

2.1.3 Previous HVS Masking

There are many different human visual system models built for watermarking systems [16]. Some HVS masking is based on the contrast sensitivity function [13, 17]. Although contrast sensitivity functions are summarized from different experiment environment, they mostly represent low pass filter functions in the frequency domain [18]. Although the CSF is very successful at evaluating the image when used as HVS masking, it has some issues to be concerned with [8]. The first issue is that the definition of CSF assumes that the image is represented as a sinusoidal pattern, such as grating, and there is no standard definition of CSF for a natural image yet. The second issue is that it only considers the difference between different frequencies and assumes that interactions between frequencies can be ignored. The third issue is that the actual image feature is not an isotropic distribution [19]. Real masking is a non-linear function and depends on various factors including the effects between the different frequencies.

Martin [13] used Winklers isotropic contrast function [15] and set the threshold based on subjective experiments to create the masking. He calculated it in the wavelet domain and transformed to the spatial domain to embed the watermark. The result shows that the masking mostly happens around the edges and almost none in the flat area.

Instead of the using the contrast sensitivity function, Delaigle [20] proposed HVS masking which is built in the FFT domain by setting the threshold on the Garbor filter. The Garbor filter usually analyzes the image features, since its parameters can take care of both amplitude and phase of the spatial frequency. He concentrated on the local area energy and only embedded the watermark on the horizontal direction of the spatial frequency. As a result, he found that the algorithm made the watermark obviously visible around the edge, and did not embed high energy of the watermark in the high activity areas. The reason causing

the visible watermark around edge, which is also called the ring effects, is that the energy around edge is not even.

In order to avoid the ring effect around the edges, Hannigan [21] combined edge detection and the contrast masking together. He applied edge detection to locate the edge areas and then reduced the watermark embedding strength in these areas after using the contrast masking to embed the watermark. Although he solved the ring effect problem, he could not deal with the problem areas with narrow, parallel lines such as human hairs. These areas still have high embedding strength and only the edge areas are corrected, therefore, the watermarked image would have very obvious salt and pepper noise in these areas.

Bartolini and Barni [22] followed three rules to develop their masking. First, the image was filtered by the band pass filter, where the watermark was to be embedded. The second step was to use the edge detection to locate the edge area in the original image. The third and fourth steps applied the median filter with different thresholds to determine the areas with brightness and darkness. The final step was the summation of the previous four steps, the result of the summation is a binary image. An offset value was given to the final summation which results in the minimum amount watermark inserted everywhere. The advantage of this method is that the masking gives the location of the watermark where it should be high and where it should be low, but it also has some weakness. The first problem is that it is quite difficult to determine the offset value based on different images. The second problem is that it did not give the mask value of the image but only the locations of the watermark.

Different from previous ideas, Voloshynovskiy [19] proposed a new method to deal with spatial domain adaptive watermarking. He used stochastic method to analyze the image texture and summarized the image model. According to the distribution of the image texture, he calculated the noise visibility function. Finally, he combined the noise visibility

function with the embedding strength to embed the watermark.

Based on the Maximum A posteriori Probability (MAP) theory and Markov Random Field (MRF), Voloshynovskiy studied two image models. The first one is a non-stationary Gaussian model, the second one is a Generalized Gaussian model. Eventually, he gave us two noise visibility functions (NVF).

For the non-stationary Gaussian model, the noise visibility function is as follows:

$$NVF(i, j) = \frac{1}{1 + \sigma_x^2(i, j)} \quad (2.5)$$

$$\sigma_x^2(i, j) = \frac{1}{(2L + 1)^2} \sum_{k=-L}^L \sum_{l=-L}^L (x(i + k, j + l) - \bar{x}(i, j))^2 \quad (2.6)$$

where, $\bar{x}(i, j) = \frac{1}{(2L+1)^2} \sum_{k=-L}^L \sum_{l=-L}^L x(i + k, j + l)$. σ_x^2 is the variance value of the local image with the block size $(2L + 1)^2$.

As for the Generalized Gaussian model, the NVF is:

$$NVF(i, j) = \frac{w(i, j)}{w(i, j) + \sigma_x^2(i, j)} \quad (2.7)$$

where, $w(i, j) = \gamma[\eta(\gamma)]^\gamma \frac{1}{r(i, j)^{2-\gamma}}$ and $r(i, j) = \frac{x(i, j) - \bar{x}(i, j)}{\sigma_x}$, γ is the shape parameter, for the most real images, $0.3 \leq \gamma \leq 1$.

It is clear that the noise visibility function is related to the local variance. From the definition of local variance, it reflects the image texture distribution. If the area in the image is very flat, or the variation of the pixel value is very small, the local variance of these areas is close to zero because the individual pixel value is almost equal to the average value of the pixels within these areas. Therefore, the noise visibility function is close to one. On the contrary, if the area in the image with high activity, or the variation of the pixel value

is very high, the local variance approaches a very large number. This causes the noise visibility function to approach zero.

With the noise visibility function, the watermark embedding formula is given as follows:

$$y = x + (1 - NVF) n S \quad (2.8)$$

where, x is the original image, NVF is the noise visibility function, n is the watermark, S is the embedding strength.

After analyzing the relationship between the local variance and the noise visibility function, it is easy to understand how the noise visibility function adjusts the embedding strength of the watermark. When there are highly textured regions in the image, their noise visibility function is close to zero, the watermark is embedded in these regions with maximum strength. When the region is very flat such as the sky area, the noise visibility function approaches one, and as a result of the embedding function, there is no watermark embedded in this area. In this way, the spatial watermark is adaptive to the image features.

The advantage of this method is that using the stochastic model to analyze the image textures, the calculation is very simple. The disadvantage is that it is only concerned with the image texture, and therefore only follows one of the previous three rules. In the flat region, it still has the capability to hide watermark based on the darkness and brightness. In the end, it tries to overcome some previous drawbacks which do not pay attention to the human visual system, and adds another item to compensate for the watermark in the flat regions. The embedding strength of S_1 is 3, which is based on the experimental value, not only raises the embedding strength in the flat regions, but also raises the embedding

strength level of the high activity regions.

$$y = x + (1 - NVF)nS + NVFnS_1 \quad (2.9)$$

Another weakness of this method is that the noise visibility function only calibrates the embedding strength S , it can not give the exact embedding strength. In other words, it only makes the embedding strength S suitable to the image texture, but whether the number of S is proper or not for the image is beyond the ability of this method. It also should point out that since $(1 - NVF) < 1$, therefore, $(1 - NVF)n < n$, it is obviously that it redistributes the watermark by reducing the watermark energy in exchange for better image quality.

2.2 Evaluation of the Watermarked Image

Before digital watermarking technology, the popular and standard method to measure image quality was using the peak signal to noise ratio (PSNR). This combines the mean square error (MSE) to evaluate the image distortion very well. Depending on the PSNR, we can easily determine the effects of normal signal processing, such as compression or the channel noise impacts, on the image. The equation of PSNR is defined as follows:

$$PSNR = 10 \lg \frac{255^2}{MSE} \quad (2.10)$$

The simplicity and well-found mathematical and statistical basis of PSNR still make it powerful in the digital watermarking field. It is very accurate in measuring degradation for different embedding strengths. It is true that the higher the embedding strength, the worse the quality of the watermarked image will be.

However, when the PSNR is applied to measure the qualities of watermarked images with the same watermarking energy, it has its limitation. The detailed explanation is as follows:

$$y = x + \alpha w \quad (2.11)$$

where, y is the watermarked image, x is original image, w is the watermark, α is the embedding strength.

$$MSE = \frac{1}{N} \sum_{i=1}^N \sum_{j=1}^N (y(i, j) - x(i, j))^2 \quad (2.12)$$

where, MSE is the mean square error, $y(i, j)$ and $x(i, j)$ are the pixel value in the watermarked image and the original image respectively. N represents the image size.

Therefore, according to Equ. (2.11),

$$MSE = \frac{1}{N} \sum_{i=1}^N \sum_{j=1}^N (\alpha w(i, j))^2 \quad (2.13)$$

The additive watermarking theory indicates that the watermarked image is the original image plus the product of watermark and the embedding strength. The mean square error is related to the difference between the watermarked image and the original image, therefore, the mean square error actually computes the energy of the watermark. It is obvious that if we add the same watermarking energy into two different images, the value of MSE will be the same. As a result, the calculated result of PSNR will be the same as well.

Recently, there are some methods suggesting the use of the wPSNR (weighted PSNR). The main idea is that it adjusts the mean square error by a weighted number which takes the human visual system into account. However, up to now, there is no existing standard for the wPSNR. In the literature, there are two methods mentioned to calculate the wPSNR.

The first one is using the noise visibility function. The idea is to use the mean square error to multiple the square of the noise visibility function to achieve the weighted mean square error [23].

$$wPSNR = 10 \lg \frac{255^2}{((RMSE)(NVF))^2} \quad (2.14)$$

where, $RMSE = \sqrt{MSE}$.

As mentioned previously, when the image has high activity areas, the result of the noise visibility function is close to zero, and the distortion caused by the watermark is mainly reduced. While in the flat regions, the noise visibility function approaches one, therefore, the degradation affect of the watermark is counted in. In this way, it mainly focuses on distortion in the flat areas because the human vision is very sensitive to the changing in those areas.

Liu [24] suggested using the CSF function [25] to deal with the wPSNR. As we mentioned previously in the thesis, the CSF is usually a low pass filter used in the frequency domain. The idea is that it regards the CSF as a human visual filter. At first, the watermark, which is the difference between the watermarked image and the original image, is filtered by this filter. If the value is outside the range of the filter, the distortion caused by the watermark is ignored. Otherwise, if the value is within the filter range, this degradation will be calculated. The summation of these degradations is regarded as the mean square error value to be used in the wPSNR. Based on this method, when the watermark is inserted in the middle frequency or the high frequency, the wPSNR value will be high. When the watermark is inserted in the low frequency, the wPSNR value will be low.

Although both methods consider human vision, they still have their drawbacks. We noticed that in the NVF masking proposed by Voloshynovskiy [19], he eventually corrected it by adding the luminance value evenly, but in the calculation of wPSNR, he did not consider

the flat regions with black or white can cover more distortions than any other gray level regions.

The second method only considers the traditional CSF function. The image texture, the edges region, is not covered by this method. Human vision is more sensitive to the changes in low frequency, but it does not mean that low frequencies cannot hold the watermark. Recently, there are many researches exploring the capability of low frequencies to handle the watermark [26, 27]. Even in the DC component of the DCT domain, Huang [28] indicated that it has the capacity of holding the watermark. Therefore, if only the CSF function was used for evaluating these watermarked images, the final results would not be very fair.

2.3 Watermark Detection

We already know that depending on the availability of the original image at the detection site, watermark detection can use either blind watermark detection or non-blind watermark detection.

In the adaptive watermark algorithm proposed by Christine and Podilchuk [29], it used the DCT domain just noticeable visual model produced by Watson [30] as the HVS masking to modify the DCT coefficients. The watermark detection idea is quite similar to the one mentioned by Cox [31], and is non-blind watermark detection. The detection processing is actually the inverse processing of the embedder. It used the original image to calculate the JND masking, and then applied it to subtract the watermark, finally it compared the real watermark and the watermark extracted from the received watermarked image via computing the normalized correlation value. Based on this value, The determination of whether the watermark was from the received watermarked image or not will be given.

The advantage of non-blind detection is very clear. It is very simple. With the original image, it makes the detection more accurate. The disadvantage is that it needs the original image be transmitted to the detection site, and therefore limits the applications.

2.4 Data Hiding Application

The watermark is not only applied in copyright protection, but also sometimes used for data hiding application. With the traditional additive watermark mechanism, we use linear correlation to discover the watermark. It is composed with two parts: one is the correlation result between the original image and the watermark, the other part is the received watermark and the real watermark. The assumption for feasibility is that the correlation value between the original image and watermark is close to zero. Therefore, the final linear correlation only depends on the received watermark and the real watermark. In this way, it achieves blind watermark detection. When the sizes of the original image and watermark matrix are large such as 512×512 , and the watermark is a pseudo random sequence, there is almost no similarity between the original image and the watermark. Therefore, this assumption is correct. After the image is divided into multiple small blocks with the sizes like 8×8 or 16×16 , the linear correlation can no longer work. This is because the similarity between the original image and the watermark increases when the image block size is small. Consequently, The linear correlation value between the original image and the watermark is not close to zero any more. Without knowing the original image, it is quite difficult to detection the watermark. In other words, the value of the linear correlation between original image and the watermark interferes with the final correlation value.

Some researches suggest using the orthogonal watermark to embed multiple watermarks in order to satisfy high data volume. With the Hadamard [10] matrix, it is easy to get multiple

orthogonal watermarks. The orthogonal watermark can only guarantee that there is no interference between the watermarks, but still cannot solve the problem that there is linear correlation between each watermark and the original image. Therefore, the traditional additive watermark mechanism is not very suitable for the data hiding application except when the original image is available at the watermark detection site.

Chen [32] analyzed this problem quite in detail. To solve this problem thoroughly, the mechanism of the watermark embedding should be changed. He classified the watermark into two groups based on whether the host image interfered with the watermark detection or not. These two groups are the host interference non-rejecting method and the host interference rejecting method.

The traditional additive watermarking method is obviously the host interference non-rejecting method. An early host interference rejecting method is proposed by Wolosewicz [33]. It represented the watermark with the number of the signal peaks of the given amplitude band. For instance, it could use more than three peaks of the signal to represent bit 1, and use less than three peaks to indicate the bit 0. If the peak number of the original signal in the certain band is not that number which the message requires, then the original signal will be changed in order to meet the requirement. In the detection, it only needs to count the number of the peak and the original signal is not required for the calculation of the detection. It is easy to imagine that the distortion of the original image is quite difficult to control, because when the difference between the peak number of the original signal and the desired number is very large, the distortion of the original signal will be very severe.

Another host interference rejecting method is the quantization index modulation (QIM) mentioned by Chen [32]. The mechanism of the quantization index modulation is to embed information by modulating an index or sequence with the embedding information and then quantizing the host signal with the sequence of quantization steps [30]. Usually, its

implementation is done by using dither modulation. For example, the lattice coding which is introduced by Cox [2] is one of the QIM methods. In addition, Swanson [34] proposed a method which regards the HVS masking value as the quantization step to represent watermark in DCT domain. With our proposed spatial domain HVS masking, we can extend Swanson's method to the spatial domain.

2.5 DCT Domain Masking

Most DCT domain maskings are mainly derived from the DCT domain. A famous one is the Watson's visual model [30] which was previously used in the reduction of quantization noise. In the application of watermarking, most algorithms proposed in the DCT domain [29] adopt this masking to control the perception. Based on the frequency sensitivity matrix [2], Watson tuned them into the DCT domain visual masking. The following shows the matrix of the frequency sensitivity. The size of this matrix is 8×8 . DC component and low frequencies focuses on the upper left corner and the high frequencies component mainly concentrate on the lower right corner.

In the following matrix, the smaller the value is, the more sensitive the eyes feel. It is obvious that the sensitivity of the human vision of the low frequency is much more than in the high frequency.

1.40	1.01	1.16	1.66	2.40	3.43	4.79	6.56
1.01	1.45	1.32	1.52	2.00	2.71	3.67	4.93
1.16	1.32	2.24	2.59	2.98	3.64	4.60	5.88
1.66	1.52	2.59	3.77	4.55	5.30	6.28	7.06
2.40	2.00	2.98	4.55	6.15	7.46	8.71	10.17
3.43	2.71	3.64	5.30	7.46	9.62	11.58	13.51
4.79	3.67	4.60	6.28	8.71	11.58	14.50	17.29
6.56	4.93	5.88	7.60	10.17	13.51	17.29	21.15

Figure 2.1: Frequency Sensitivity Matrix

Based on the experiments, Watson created visual masking according to two elements: one is the luminance and the other one is the contrast. He pointed out that if the average background is brighter, then small changes in the luminance are not noticeable. The luminance masking is computed according to the following equation:

$$t_L[i, j, k] = t[i, j] \left(\frac{C_0[0, 0, k]}{C_{0,0}} \right)^{\alpha_T} \quad (2.15)$$

where, $t_L[i, j, k]$ is the luminance masking value at position (i, j) in the block k , t is the frequency sensitivity matrix, $C_0[0, 0, k]$ is the DC component of the image block k , $C_{0,0}$ is average of the DC components in the all image blocks. The α_T is the constant and the suggested value is 0.649.

After the luminance masking was created, Watson considered the contrast masking represents the reduction in visibility of one frequency due to the presence of another similar

frequency. The final masking is given according to the following equation,

$$s[i, j, k] = \max(t_L[i, j, k], |C_0[i, j, k]|^{0.7}, t_L[i, j, k]^{0.3}) \quad (2.16)$$

where, $t_L[i, j, k]$ is the luminance masking, $C_0[i, j, k]$ is the DCT transform value of the image block. $s[i, j, k]$ is the final masking at position (i, j) in the block k .

If we choose to use this masking, the energy of the watermark mainly concentrates in the high frequency because the high frequency has high margin. Many researchers select the middle frequency to embed the watermark because the high frequency will be discarded during compression and low frequency is more sensitive. However, the watermark in low frequency is more robust than in middle frequency.

In addition, we notice that in the luminance masking, Watson uses the ratio between the local block luminance and the global luminance to create the masking. This is the same theory as the Weber's law. The brighter the background, the higher value of the masking will be. As we analyzed previously, it is not very suitable when the background is dark. Due to the saturation of human vision, when the background becomes very dark, it also increases the masking value in this region.

Some algorithms [34] also suggested that after the DCT domain masking is applied, it also should use spatial masking to do the final adjustment in order to reduce the distortion. The image is finally represented in the spatial domain. Therefore, spatial masking is very important.

Chapter 3

Proposed Adaptive Digital Watermarking Method

After reviewing several different masking technologies, we can conclude that the previous works are only concerned with one or two rules, and it can be very difficult to determine the threshold values. Here, we introduce a new approach which takes all three rules into account.

3.1 Proposed HVS Masking

Following these criteria, we first create luminance masking, texture masking and edge masking based on the image features, and then combine these three masking together to get a comprehensive final masking for the watermark, as shown in Figure 3.1. The researches [26, 27] following these rules were carried out in the wavelet domain but not in the spatial domain.

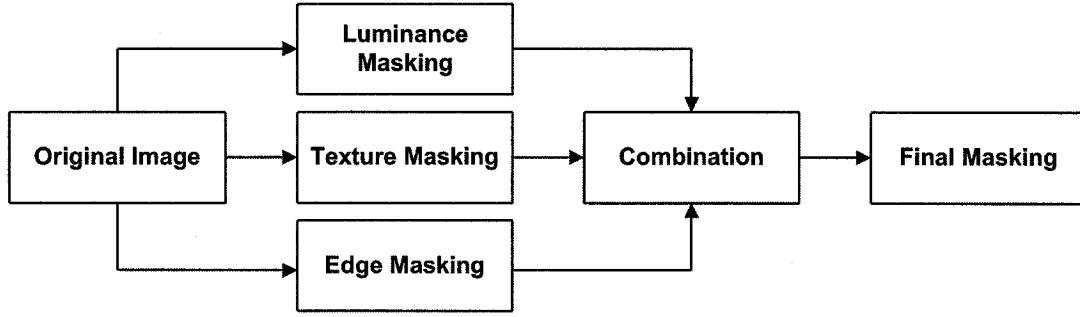


Figure 3.1: HVS Masking

3.1.1 Luminance Masking

The reason to use luminance is that the darkness and brightness in the image is represented by the luminance. The visibility of the luminance threshold in the spatial domain depends on two factors [12]: one is the average background luminance, the other one is the non-uniformity of the background luminance. A very good luminance masking proposed by Chou [12] is based on these two factors and satisfies our requirements, and it focuses on the local Just Noticeable Difference. All the calculations are based on the image block with size 5×5 , therefore in the implementation, we will use the sliding window with the size 5×5 . This luminance masking is also used in the video distortion measurement system [35]. In the Chou's algorithm, all the parameters are determined by the experimental environment and statistical results. The calculations of this masking are as follows [12]:

$$M_L(x, y) = \max\{f_1(bg(x, y), mg(x, y)), f_2(bg(x, y))\} \quad (3.1)$$

$$f_1(bg(x, y), mg(x, y)) = mg(x, y) \alpha(bg(x, y)) + \beta(bg(x, y)) \quad (3.2)$$

$$f_2(bg(x, y)) = \begin{cases} T_0(1 - (\frac{bg(x, y)}{127})^{0.5}) + 3 & bg(x, y) \leq 127 \\ \gamma(bg(x, y) - 127) + 3 & bg(x, y) > 127 \end{cases} \quad (3.3)$$

$$\alpha(bg(x, y)) = 0.0001bg(x, y) + 0.115 \quad 0 \leq x < H, \quad 0 \leq y < W \quad (3.4)$$

$$\beta(bg(x, y)) = \lambda - 0.01bg(x, y) \quad 0 \leq x < H, \quad 0 \leq y < W \quad (3.5)$$

where, M_L is the luminance masking, f_1 is the spatial masking function. $bg(x, y)$ is the background luminance and $mg(x, y)$ is the maximum weighted average of luminance differences around the pixel at (x, y) . H and W indicate the image height and width. f_2 is the visibility function based on the background luminance. The parameter α and β are the background luminance dependent functions. The value of some other parameters are: $T_0 = 17$, $\gamma = \frac{3}{128}$, $\lambda = \frac{1}{2}$.

The calculation of $mg(x, y)$ function is as follows:

$$mg(x, y) = \max_{k=1,2,3,4}\{|grad_k(x, y)|\} \quad (3.6)$$

$$grad_k(x, y) = \frac{1}{16} \sum_{i=-3}^3 \sum_{j=-3}^3 p(x-3+i, y-3+j) G_k(i, j) \quad 0 \leq x < H, \quad 0 \leq y < W \quad (3.7)$$

where $p(x, y)$ is the pixel value at the position (x, y) , H and W represent the image block size. The value of G_1 , G_2 , G_3 , and G_4 are represented as follows:

Table 3.1: Parameter G_1 in the Luminance Masking Calculation

0	0	0	0	0
1	3	8	3	1
0	0	0	0	0
-1	-3	-8	-3	-1
0	0	0	0	0

Table 3.2: Parameter G_2 in the Luminance Masking Calculation

0	0	1	0	0
0	8	3	0	0
1	3	0	-3	-1
0	0	-3	-8	0
0	0	-1	0	0

Table 3.3: Parameter G_3 in the Luminance Masking Calculation

0	0	1	0	0
0	0	3	8	0
-1	-3	0	3	1
0	-8	-3	0	0
0	0	-1	0	0

Table 3.4: Parameter G_4 in the Luminance Masking Calculation

0	1	0	-1	0
0	3	0	-3	0
0	8	0	-8	0
0	3	0	-3	0
0	1	0	-1	0

Finally, the calculation of $bg(x, y)$ is as follows:

$$bg(x, y) = \frac{1}{32} \sum_{i=1}^5 \sum_{j=1}^5 p(x-3+i, y-3+j)B(i, j) \quad 0 \leq x < H, 0 \leq y < W \quad (3.8)$$

where, $p(x, y)$ is the pixel value at the position (x, y) . H and W represent the image block size. The value of B is:

Table 3.5: Parameter B in the Luminance Masking Calculation

1	1	1	1	1
1	2	2	2	1
1	2	0	2	1
1	2	2	2	1
1	1	1	1	1

3.1.2 Texture Masking

As for texture masking, the simplest method is to use the normalized variance. The first step is to use a sliding window with the size 3×3 to calculate the local variance. The second step is to find the maximum value of the local variance. The third step is to normalize the variance value of the whole image with the maximum from step two. The weakness of this method is that it locates the image texture, but it does not determine the exact masking value. Voloshynovskiy [19] also used the local variance to determine the noise visibility function, but the same problem was that it could not determine the masking value directly but only did the adjustment to the embedding strength.

It is obvious that the local variance shows the relationship between the individual pixel value and the mean value of the group pixels inside the sliding window. This is the key element to represent whether the image texture is high activity or uniform. The texture masking we proposed here uses this relationship and takes the absolute value of the distance between each pixel and local average value of the pixels within the sliding window

as the masking. The sliding window size is still 3×3 .

$$M_T = |x(i, j) - \bar{x}(i, j)| \quad (3.9)$$

$$\bar{x}(i, j) = \frac{1}{(2L+1)^2} \sum_{k=-L}^L \sum_{l=-L}^L x(i+k, j+l) \quad (3.10)$$

where, $x(i, j)$ is the pixel at the position (i, j) . M_T is the texture masking, $(2L+1)^2$ represents the image block.

The reason that we did not directly choose the variance or the standard deviation, which is the square root of the variance, to create the texture masking was mainly because their value is very large compared with the difference between pixel value and the local mean value. If the variance or the standard deviation was used as the masking, the watermarked image would have very obvious visible salt and pepper noise.

3.1.3 Edge Masking

After introducing the luminance and texture masking, we also think about the edge masking. Many proposed methods did not handle this very well since edge detection can only locate the edge in a binary way, such as edge or not edge, and can not give the exact masking value such as how many pixels on the edge can be changed. Of course, when the texture masking is created, it also includes the edge parts, however, if using the texture masking directly, the edge is easily corrupted and results in the severe distortion to the image. Therefore, correcting the edge region is inevitable. Here we will use improved un-sharp masking method proposed by Keith Wiley [36] to generate edge masking.

The boundary of the object in the gray scale image is the sharp changing of the gray level. When human vision senses the object outside the world, it recognizes the outlines by en-

hancing the edge beyond its true natural appearance. The mechanism of this enhancement in the vision system decreases the dark side of the edge and increases the bright side of the edge at the same time. Therefore, it makes the edges of the object more obvious.

There are two ways to create the edge masking. One is un-sharp masking which blurs the original image first, and then gets the difference between original image and blurred image. This difference will be un-sharp masking. The other one is using the Laplacian filter which is also referred to as improved un-sharp masking. We choose the Laplacian filter to deal with the edge masking because it can cover very small gray level changes.

$$M_E = L(I) \quad (3.11)$$

where, M_E is the edge masking, I is the original image, $L(.)$ is the Laplacian filter operation.

3.1.4 Final Masking

After we create the luminance masking, texture masking and edge masking, the next step is to combine them together. We combine the texture masking and edge masking first. In the edge areas, we select the minimum value between the texture masking and edge masking. The reason we choose the minimum value is that the edge has less ability to cover the watermark than high activity areas. After this combination is finished, we will select the maximum value between the luminance masking and the masking from the previous step. The reason we pick the maximum value is that in the flat area, the masking value is decided by the luminance masking. In the high activity region, the masking value is determined by the texture masking. This final masking not only takes the luminance into account, but

also the effects of the texture and edges of the image.

$$M_F = \max(M_L, \min(M_E, M_T)) \quad (3.12)$$

where, M_F is the final masking, M_L is the luminance masking, M_E is the edge masking and M_T is the texture masking.

3.2 Watermark Embedding

Having produced the HVS masking, we can use it to control the embedding strength. The watermarking embedding equation is already introduced in Equ. (1.5). After we use the masking, the equation will be:

$$y = x + mw \quad (3.13)$$

where, y is the watermarked image, x is the original image, w is the watermark, m is the masking value.

The watermark is defined as a direct spread spectrum sequence, which is generated by the pseudo random sequence with number 1, and -1. From the above equation, it is understandable that after the watermark is multiplied by the masking value, it falls to an imperceivable range. Therefore, the watermark will be adaptive to the image and the quality of the watermarked image will be improved.

3.3 Evaluation of the Watermarked Image

Considering all the elements of the sensitivity of human vision, we suggest that we can use this proposed HVS masking as the human vision filter. First of all, we calculate the absolute value of the difference between the original image and the watermarked image, which actually is the watermark. Secondly, we apply the proposed HVS masking as a filter to gauge the difference. We compare the difference value and masking value pixel by pixel. If the difference value is smaller than or equal to the masking value, the impact of this difference on the quality of the image is ignored. Otherwise, the difference between the watermark and the value of masking is the candidate for the mean square error calculation. The final calculation is as follows:

$$wMSE = \frac{1}{N} \sum_{i=1}^N \sum_{j=1}^N d(i, j)^2 \quad (3.14)$$

$$d(i, j) = \begin{cases} 0 & \text{if } |w(i, j)| \leq m(i, j) \\ |w(i, j)| - m(i, j) & \text{if } |w(i, j)| > m(i, j) \end{cases} \quad (3.15)$$

$$wPSNR = 10 \lg \frac{255^2}{wMSE} \quad (3.16)$$

where, $d(i, j)$ is the difference between the watermark and masking value, $w(i, j)$ is the watermark value, $m(i, j)$ is the masking value.

3.4 Watermark Detection

It is known that traditional watermarking detection is done by calculating the linear correlation value between the watermark and the received image. Because the watermark is

a pseudo random sequence, the correlation between the watermark and the original image is usually close to zero. Therefore, the linear correlation depends primarily the received watermark and the real watermark. Normally, if there is no attack or signal processing, the linear correlation value is close to one. Based on this value, we determine if the received image has a watermark or not. Sometimes, due to attacks or the signal processing, a threshold is set to decide whether the received image has a watermark or not. It also can represent that the degree of the effect that the attacks or the signal processing impact on the watermark.

Linear correlation has good performance in finding a signal based on the similarity between two signals. During our embedding process, the shape of the watermark is changed by the HVS masking, therefore, the linear correlation will fail to detect the watermark if we directly calculate it between the watermark and received image. Without the availability of the HVS masking, it is quite difficult to use linear correlation to deal with the detection. This is another reason that the detection site [29] needs the original image. With the original image, it is easy to derive the HVS masking.

We propose watermarking blind detection based on noise shaping technology. Noise Shaping is the technique to move noise from one part of the spectrum to another. It reconstructs the noise spectrum according to the application and leaves the noise energy unchanged.

Previously, noise shaping technology is mostly used in the applications of the noise reduction or removal. When traditional noise removal technology such as Wiener filter or the wavelet domain de-noising methods are used to eliminate the noise, sometimes not only the noise but also some of the useful information in the image is also removed. Therefore, the quality of the image is degraded. The degradation is often proportional to the amount of the noise reduced or removed. Noise shaping is quite different. It just hides the noises in the spectrum where the human vision or the auditory can not perceive them, rather than

removing them from the original images.

Depending on the applications, there are different methods to achieve noise shaping. In the hardware design in order to get rid of the A/D or D/A quantization noise, a famous method used is the delta-sigma data converters [37]. In the audio application, the well-known method is temporal noise shaping [38]. The spatial noise shaping proposed by Kuo is also used in the image coding application [39].

Herre [38] mathematically proved that the linear prediction coefficients of the signal in the frequency domain will record the envelope of the same signal in the time domain. Most of the researches [38, 40] recommended that the frequency domain be the DCT domain. The reason is that compared to the FFT domain, DCT domain has no phase calculation, or the complex value, and therefore simplifies the whole computation. In addition, the DCT domain does not have mirror frequency which is the characteristics of FFT domain. Herre's proof is applied in an one-dimension such as audio, therefore we extend this for image applications as two one-dimension signals together.

At the embedding site, after the HVS masking is created from the original image, it is then transferred into the DCT domain. We will apply the linear prediction coefficients to the rows and columns in order to capture the envelope of this signal in the spatial domain. This linear prediction coefficients will be transferred to the detection site. Because it only records the shape of the HVS masking, we will also transfer the total shaped watermarking energy to the detection site.

At the detection site, with the availability of the those coefficients, we will use them as the IIR filter parameters to produce the frequency response. This frequency response records the shape of the HVS masking. Therefore, it will be very easy for us to un-shape the watermark. The reason we defined the image as two one-dimension signals was because the 2D IIR filter is not very stable [39].

3.5 Data Hiding in Spatial Domain

With the HVS masking we proposed previously, the method proposed by Swanson [34] is easily extended to the spatial domain. Here, we mainly focus on how the HVS masking is merged into other watermark algorithms. First, we divide the image into multiple small blocks with the size 4×4 . Each block will represent one message bit, and then use the spread spectrum which is a pseudo random sequence with the value -1 or 1 to represent the watermark. After the watermark is generated, it first makes the pixel value of the original image become the integer times of the HVS masking. If the message bit is one, it adds one quarter of the masking value, otherwise, it subtracts one quarter of the masking value. This kind of dither operation implements the watermark embedding. The detailed process is as follows:

1. Extracting the HVS masking from the original image;
2. Dividing the original image into 4×4 block;
3. Dividing the HVS masking into 4×4 block;
4. Finding the minimum HVS masking value of each 4×4 block;
5. Generating the pseudo random sequence as watermark, the size is 4×4 ;
6. If the message of bit is 1, using the following equation

$$A = (\text{round}(\frac{I}{T}) + 0.25 N) T \quad (3.17)$$

where, A is the watermarked block, I is the original image block, T is the value of the masking got from the step 4, N is the watermark with value 1 or -1.

7. If the message of bit is -1, using the following equation

$$A = (\text{round}(\frac{I}{T}) - 0.25 N) T \quad (3.18)$$

where, A is the watermarked block, I is the original image block, T is the value of the masking got from the step 4, N is the watermark with value 1 or -1.

8. After all the image blocks are done, the final watermarked image and the masking value are transferred to the watermark detection site.

The detection method is relatively quite simple. It also divides the image into blocks whose size is the same as the embedder site, and then it uses the following equation to extract the watermark from each block.

$$A' = \text{sign}(\frac{R}{T} - \text{round}(\frac{R}{T})) \quad (3.19)$$

In the equation, R is the received image block, T is the HVS masking value, and A' is the calculated watermark.

If the sign of the final matrix is the same as watermark, the message bit is 1. If the final matrix sign is opposite to the watermark, the message bit is -1. From the above analysis, it is easy to see that the original image is not required in the watermark detection.

3.6 Extend Proposed Spatial Domain Masking to DCT

Domain

In this thesis, we introduce a method to extend the proposed spatial domain HVS masking to the DCT domain. Discrete cosine transform and inverse discrete cosine transform calculation are the bridges that connect the spatial domain and the DCT domain.

We assume that the image is divided into multiple blocks with the size 8×8 . The 2-D DCT block calculates the two-dimensional discrete cosine transform of the input signal.

The equation [41] for the two-dimensional DCT is as follows:

$$F(m, n) = \frac{2}{\sqrt{MN}} C(m) C(n) \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x, y) \cos \frac{(2x+1)m\pi}{2M} \cos \frac{(2y+1)n\pi}{2N} \quad (3.20)$$

where, $C(m), C(n) = \frac{1}{\sqrt{2}}$ when $m, n = 0$; otherwise, $C(m), C(n) = 1$. $f(x, y)$ is the pixel value in the spatial domain, $F(m, n)$ is the pixel value in the DCT domain. M, N represents the block size.

The equation of the two-dimensional IDCT [42] is as following:

$$f(x, y) = \frac{2}{\sqrt{MN}} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} C(m) C(n) F(m, n) \cos \frac{(2x+1)m\pi}{2M} \cos \frac{(2y+1)n\pi}{2N} \quad (3.21)$$

where $C(m), C(n) = \frac{1}{\sqrt{2}}$ when $m, n = 0$; otherwise, $C(m), C(n) = 1$. $f(x, y)$ is the pixel value in the spatial domain, $F(m, n)$ is the pixel value in the DCT domain. M, N represents the block size.

The following matrixes represent the one block in the spatial domain and in DCT domain respectively.

$$\begin{bmatrix}
 f_{00} & f_{01} & f_{02} & f_{03} & f_{04} & f_{05} & f_{06} & f_{07} \\
 f_{10} & f_{11} & f_{12} & f_{13} & f_{14} & f_{15} & f_{16} & f_{17} \\
 f_{20} & f_{21} & f_{22} & f_{23} & f_{24} & f_{25} & f_{26} & f_{27} \\
 f_{30} & f_{31} & f_{32} & f_{33} & f_{34} & f_{35} & f_{36} & f_{37} \\
 f_{40} & f_{41} & f_{42} & f_{43} & f_{44} & f_{45} & f_{46} & f_{47} \\
 f_{50} & f_{51} & f_{52} & f_{53} & f_{54} & f_{55} & f_{56} & f_{57} \\
 f_{60} & f_{61} & f_{62} & f_{63} & f_{64} & f_{65} & f_{66} & f_{67} \\
 f_{70} & f_{71} & f_{72} & f_{73} & f_{74} & f_{75} & f_{76} & f_{77}
 \end{bmatrix} \tag{3.22}$$

$$\begin{bmatrix}
 F_{00} & F_{01} & F_{02} & F_{03} & F_{04} & F_{05} & F_{06} & F_{07} \\
 F_{10} & F_{11} & F_{12} & F_{13} & F_{14} & F_{15} & F_{16} & F_{17} \\
 F_{20} & F_{21} & F_{22} & F_{23} & F_{24} & F_{25} & F_{26} & F_{27} \\
 F_{30} & F_{31} & F_{32} & F_{33} & F_{34} & F_{35} & F_{36} & F_{37} \\
 F_{40} & F_{41} & F_{42} & F_{43} & F_{44} & F_{45} & F_{46} & F_{47} \\
 F_{50} & F_{51} & F_{52} & F_{53} & F_{54} & F_{55} & F_{56} & F_{57} \\
 F_{60} & F_{61} & F_{62} & F_{63} & F_{64} & F_{65} & F_{66} & F_{67} \\
 F_{70} & F_{71} & F_{72} & F_{73} & F_{74} & F_{75} & F_{76} & F_{77}
 \end{bmatrix} \tag{3.23}$$

In a simple example, we only embed the watermark in two middle frequencies such as F_{05} and F_{50} . Therefore, the location of the other frequency will be zero. The masking matrix for this example becomes the following:

$$\begin{bmatrix}
 0 & 0 & 0 & 0 & 0 & F_{05} & 0 & 0 \\
 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
 F_{50} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0
 \end{bmatrix} \quad (3.24)$$

Although there are only two frequency components in the matrix, it affects the whole spatial domain block value after the IDCT transform. This masking should be restricted by the spatial masking in the same block position. In order to do so, we calculate this masking back to the spatial domain with the above IDCT equation. Suppose the proposed spatial domain masking for one block is represented as follows:

$$\begin{bmatrix}
 SM_{00} & SM_{01} & SM_{02} & SM_{03} & SM_{04} & SM_{05} & SM_{06} & SM_{07} \\
 SM_{10} & SM_{11} & SM_{12} & SM_{13} & SM_{14} & SM_{15} & SM_{16} & SM_{17} \\
 SM_{20} & SM_{21} & SM_{22} & SM_{23} & SM_{24} & SM_{25} & SM_{26} & SM_{27} \\
 SM_{30} & SM_{31} & SM_{32} & SM_{33} & SM_{34} & SM_{35} & SM_{36} & SM_{37} \\
 SM_{40} & SM_{41} & SM_{42} & SM_{43} & SM_{44} & SM_{45} & SM_{46} & SM_{47} \\
 SM_{50} & SM_{51} & SM_{52} & SM_{53} & SM_{54} & SM_{55} & SM_{56} & SM_{57} \\
 SM_{60} & SM_{61} & SM_{62} & SM_{63} & SM_{64} & SM_{65} & SM_{66} & SM_{67} \\
 SM_{70} & SM_{71} & SM_{72} & SM_{73} & SM_{74} & SM_{75} & SM_{76} & SM_{77}
 \end{bmatrix} \quad (3.25)$$

Because there are many zeros in the DCT domain masking matrix, after the calculation of the IDCT, we can get a group of 64 expressions shown in the following:

$$0.25(F_{50} \cos \frac{5(2i+1)\pi}{16} + F_{05} \cos \frac{5(2j+1)\pi}{16}) \leq SM_{ij} \quad 0 \leq i \leq 7, 0 \leq j \leq 7$$

The above expressions show that after computing the IDCT of the desire DCT masking, the value is restricted by the spatial domain masking. In addition, we still wish that in the DCT domain, the energy of the watermark be as large as possible. Because the watermark is still a bipolar value of 1 or -1, the energy of the watermark will be the following expression:

$$E_1 = F_{50}^2 + F_{05}^2 \quad (3.26)$$

Therefore, the whole processing becomes a pure mathematical question which is a quadratic function and subject to bounds on some of the variables. The goal is to find the suitable variables of F_{50} and F_{05} and make the energy to reach the maximum, meanwhile, these variables are restricted by the above 64 expressions. Coleman [43] has proved the algorithm and the function has been implemented in Matlab. However we notice that this function can only find suitable variables to satisfy the function to reach the minimum value. Therefore, we change the energy function into the following expression:

$$E_2 = -F_{50}^2 - F_{05}^2 \quad (3.27)$$

The reason is that the variables which make the function E_1 to reach its maximum are also the variables which make the function E_2 to reach its minimum. In other words, the same variables can make these two functions to obtain the extreme value.

In Matlab, there is the expression to help get the variables.

$$\min_x \frac{1}{2} x^T H x + f^T x \quad Ax \leq b$$

where, x is the DCT domain masking which is the value we want to get, A will be the cosine matrix list in the above 64 expressions, b is vector of the spatial domain masking value which restricts the DCT domain masking, and $f = 0$. Because we have two variables F_{50} and F_{05} , in order to express E_2 , we can write the matrix H as the following:

$$\begin{bmatrix} -2 & 0 \\ 0 & -2 \end{bmatrix}$$

After the optimal calculation, we can get the value of the two variables F_{50} and F_{05} , which will be the masking values at those frequencies.

Chapter 4

Implementation and Results Analysis

After introducing the HVS based adaptive digital image watermarking method in detail, we will present some implementation results in this chapter.

4.1 HVS Masking

The implementation of the proposed HVS masking includes three elements: luminance masking, texture masking and edge masking. In the implementation, we choose the image “Lena” (shown in Figure 4.2) as an example because it is quite popular in the image processing field. In the following sections, we will present and analyze the results step by step.

4.1.1 Luminance Masking

The first masking we show here is the luminance masking. Because the luminance masking is based on Chou’s algorithm. According to the introduction in the Section 3.1.1, the

implementation result of the Chou's algorithm is shown below.

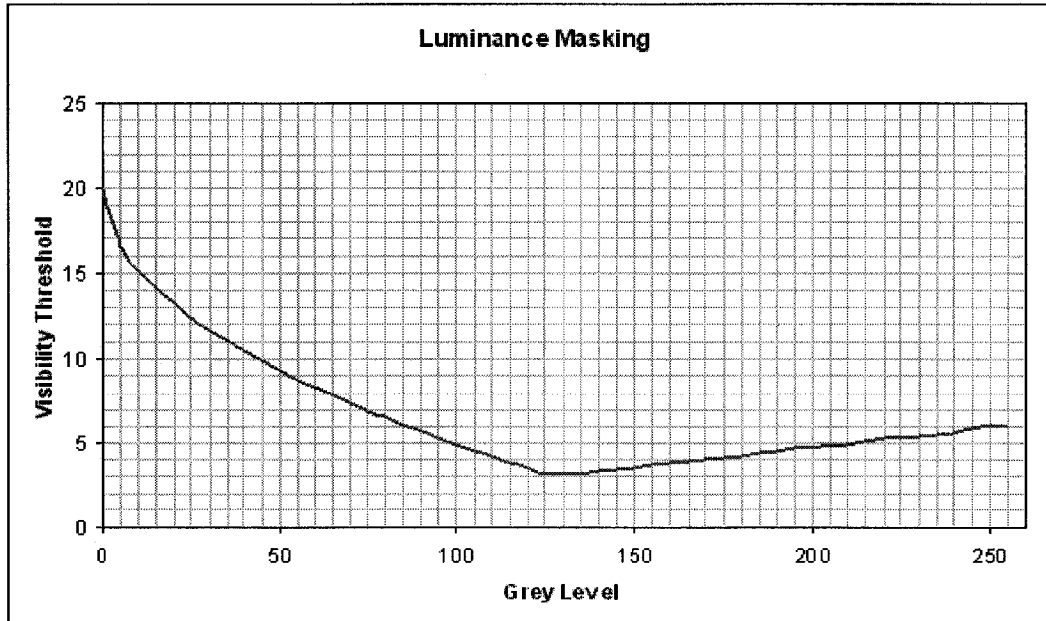


Figure 4.1: Luminance Masking

The advantage of Chou's algorithm is that it uses the local luminance and gives the redundant pixel value of masking directly. For example, if the grey level is 0, the masking value can be 20. This means that if we change the original pixel value from zero to 20, human vision will not tell the difference. Similarly, if the original pixel value of 127 is changed to something between 124 and 130, it is indistinguishable by human vision because the masking value is 3. Of course, because each pixel in the image is represented by one byte, it only has the valid value from 0 to 255. The operation of the watermark includes the plus and the minus, therefore, the issue of valid range can not be ignored. In the implementation, if the value is below 0 or exceed 255, the final value will accept 0 or 255 only.

The figure of the luminance masking also shows that when the value of the original pixel value is below or above certain level, the masking value will increase. This obeys the rule of the eyes with saturation of the brightness and darkness. When the gray level is 0 which means it is totally dark, the masking value is 20, and when the grey level reaches 255 that indicates it is bright, the masking value will be 6. This means that the darkness is more tolerant for the watermark than the brightness. However, in the middle range of grey level, the value of masking is not very high. The lowest masking value is around 3. As mentioned previously, Voloshynovskiy [19] chose the value 3 to be the embedding strength to fill the watermark in the flat region of the image. From Chou's algorithm, this value is quite feasible.

Here, we show the implementation result of normalized luminance masking (Figure 4.3). When the image has big areas covered with darkness or brightness. This luminance masking will show the advantage.

We did the normalization to the masking because the masking value is very small, the whole image will be black without normalization. After the normalization, it is clear that the white part in the masking has the high watermark embedding strength. Meanwhile, the black part has the low watermark embedding strength.



Figure 4.2: Original Image "Lena"



Figure 4.3: Normalized Luminance Masking of Image "Lena"

4.1.2 Texture Masking

By using the method suggested in Section 3.1.2, we present the implementation result of the texture masking shown in Figure 4.4. From the result, it is clear that the white area has the high margin to cover the watermark, while the black area has low margin to cover the watermark.



Figure 4.4: Texture Masking of Image “Lena”

4.1.3 Edge Masking

The result of the un-sharp masking by using Matlab un-sharp filter is shown in Figure 4.5.



Figure 4.5: Edge Masking Created by Un-Sharp Filter

From the result, we find although un-sharp filter gives the edges of the person, it does not cover all the edges in the background. The other way to achieve the edge masking is to use Laplacian filter [36]. Laplacian filter detects every pixel change happening in the image. Another advantage is that the result of the filtered image is not binary. The implementation result of filtered image is shown in Figure 4.6.



Figure 4.6: The Image “Lena” after the Laplacian Filter

From the mechanism we mentioned above, this edge masking considers the every change of different grey scale value. That is the reason we see the masking does not only have masking on the edges. In order to deal with this situation, during the implementation, we will use edge detection to help locate the edges. The algorithm we use for edge detection is Canny’s algorithm, because it is good at detecting the weak edges. The edge detection result of Canny’s algorithm is shown in Figure 4.7.



Figure 4.7: Edge Detection of Image “Lena”

The edge detection result is a binary image. It shows the location of the edges with the white lines. In order to avoid the salt and pepper noise appearing in the very thin lines areas such as human hairs, which is the unsolved problem mentioned and happened in Hannigan method [21], dilation processing is added after the edge detection image.

Dilation and erosion processing are a pair of morphological algorithm which are very powerful in the application of boundary extraction, image segment and pattern recognition. Morphology is a group of image processing operations based on the image shapes. At first, the dilation operation defines the structuring element which is a matrix consisting of only 0's and 1's. The matrix can have any arbitrary shape and size. The elements in the matrix with value of 1 define the neighborhood. Usually, The matrix size is much small than the image size and good for the local image processing. The rule [44] of the dilation operation is that the value of the output pixel is the maximum value of all the pixels in the input pixel's neighborhood defined by the structuring element. In a binary image, if any of the

pixels is set to the value 1, the output pixel is set to 1. In addition, pixels outside the border restricted by the structuring element are assigned the minimum value afforded by the data type. For binary images, these pixels are set to be 0. The shape of the structuring element is not fixed, it can be any of circle, diamond, line, square and so on. During the implementation, since we use dilation operation to reduce some texture masking value, and since the sliding window for producing texture masking is 3×3 , we chose the structuring element in the dilation operation to be square shape and with the size 3×3 . The result of dilated edge detection is a binary image and shown in Figure 4.8.



Figure 4.8: Dilated Edge Detection of Image "Lena"

With the help of the dilated edge detection of the image, we can use it as a filter to wipe off the non-edge regions in the edge masking derived from the Laplacian filter and only keep the edge areas.

4.1.4 Final Masking

The final masking is built based on the luminance masking, texture masking and edge masking produced separately. Based on the rule two that the edge is sensitive to the change more than the high activity areas, therefore, only in the edge areas, the minimum value between the texture masking and the edge masking is chosen. After testing 25 images with this modified texture masking to embed the watermark, we find that the edges in some images have some distortions. As a result, we give a weight number to adjust the masking of the edge areas. Based on the experiments, the final weight number is 0.4 ~ 0.5.

After combining the edge masking and texture masking together into the modified texture masking, we will combine the luminance masking with it. Between the modified texture masking and luminance masking, we will choose the maximum value. The reason is that in the texture masking, if there are flat areas, the masking will be determined by the luminance masking, while if there are high activity areas, the masking will mostly be determined by the edge and texture masking. Based on the results of the implementation, the final masking equation is changed to:

$$M_F = \max(M_L, p \times \min(M_T, F(M_E, E_{DI}(E_{DE}(I)))))) \quad (4.1)$$

where, M_F is the final masking, p is the weight number, range from 0.4 to 0.5, M_L is the luminance masking, M_T is the texture masking, I is the original image, $E_{DE}(\cdot)$ is the edge detection operation, $E_{DI}(\cdot)$ is the dilation operation. $F(\cdot)$ is the filter operation.

Figure 4.9 shows the final masking of the image. Different from other HVS masking which only concentrates on the edge of the images, this masking has high values in the high activity areas and darkness regions, the other parts of the masking values are lower.

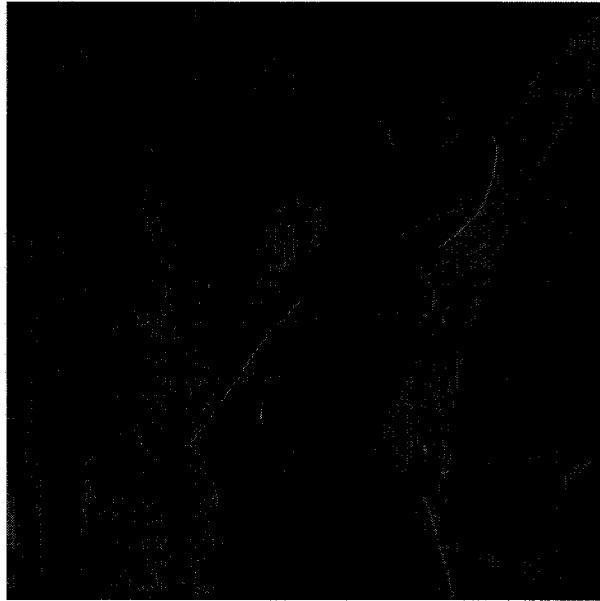


Figure 4.9: Final HVS Masking of Image “Lena”

4.2 Watermark Embedding

The implementation results of the adaptive watermarked image with proposed masking is shown in Figure 4.13. The result is also compared with the watermarked image without any masking (Figure 4.11) and the watermarked image (Figure 4.12) with noise visibility function proposed in [19]. From the results listed in [19], we find that the Generalized Gaussian model reduces more watermarking energy than the non-stationary model. With the same watermarking energy, the Generalized Gaussian model will have severe edge distortion. Therefore, we choose the non-stationary model to do the comparison with our proposed masking. With the same image group, all watermarked images contain the same watermarking energy. When we use proposed masking to embed the watermark, with the same watermarking energy, we find that the watermark can be made more robust.

The first group is the image “Lena”. The PSNR value of each watermarked image “Lena” is 32.5 dB. Without the masking, the embedding strength is 6. When using the NVF non-stationary model, the value of S is 6.4.



Figure 4.10: Original Image of “Lena”



Figure 4.11: Watermarked Image “Lena” without Masking



Figure 4.12: Watermarked Image “Lena” with NVF (Non-Stationary Model) $S = 6.4$



Figure 4.13: Watermarked Image “Lena” with Proposed Masking

Another example is performed on the image “Work”. The PSNR of each watermarked image “work” is 27.8 dB. Without the masking, the embedding strength is 11, the value of S is 13.1 when we use the NVF for the non-stationary model.



Figure 4.14: Original Image of “Work”



Figure 4.15: Watermarked Image “Work” Without Masking



Figure 4.16: Watermarked Image "Work" With NVF (Non-Stationary Model) $S=13.1$



Figure 4.17: Watermarked Image "Work" With Proposed Masking

The third example is performed on the image “Discussion”. The PSNR of each water-marked image is 28.2278 dB. Without the masking, the embedding strength is 10.3 and the S is 11.6 when we use the NVF for the non-stationary model.



Figure 4.18: Original Image “Discussion”

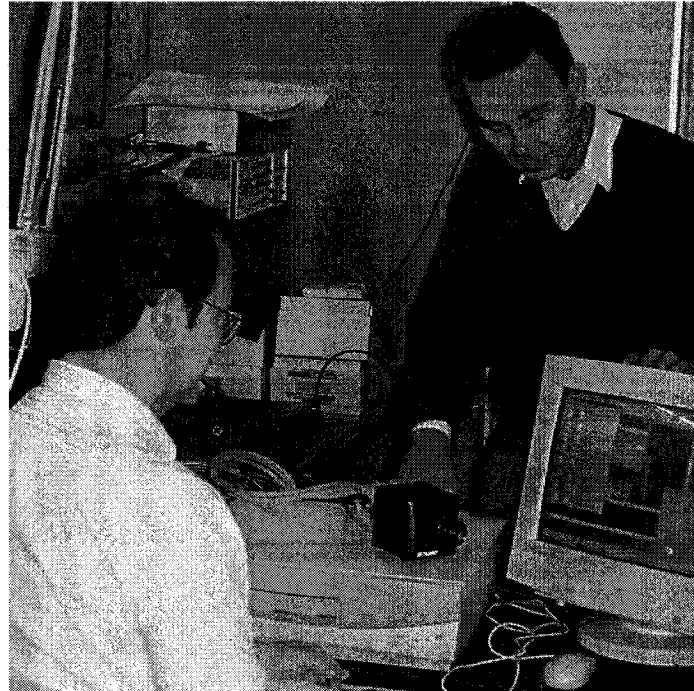


Figure 4.19: Watermarked Image "Discussion" Without Masking



Figure 4.20: Watermarked Image "Discussion" With NVF (Non-Stationary) S=11.6



Figure 4.21: Watermarked Image “Discussion” With Proposed Masking

4.3 Quality Measurement

From the implementation results, it is clear that the quality of watermarked image with the proposed masking is the best. All watermarked images in the same group have the same PSNR value. Here, we will use the wPSNR to evaluate the quality of these images.

Because the HVS masking is derived by ourselves, we first need do some experiments to highlight that this HVS masking has the ability to measure the quality of the images, and does not conflict with the PSNR.

First, we compare the same images with the same changing value. We only change the pixel value in one block with size of 8×16 , but we choose the locations of the block to be different. In Figure 4.22 and Figure 4.23, all the pixels in a block with size of 8×16 have

changed to the original value plus 12. In the first following image, the block we choose is on the shoulder. While in the second following image, the block we select is on the background with the black color.



Figure 4.22: Block Value Changed On the Shoulder of Image “Lena”



Figure 4.23: Block Value Changed On the Background With Block Color of Image “Lena”

It is obvious that the visual result of the second image is better than the first image, because we can tell there is the changes in the first image but we can not discover the changes in the second image. The reason is that the changes happened in the second image is under the masking which makes it invisible, while the changes happened in the first image is beyond the masking. Both locations are smooth areas but the masking margin in the black area is higher than in the gray area. If we use proposed masking as the filter to calculate the wPSNR, the results are: the wPSNR value in the first image is 62.8596 dB and the wPSNR value in the second image is 74.5311 dB. This measurement matches the visual results we mentioned previously. However, PSNR can not tell the difference between these two images due to the same amount of the change. The value of the PSNR of both images is 68.6914 dB.

Second, we compare the images with different quality factor of compression.

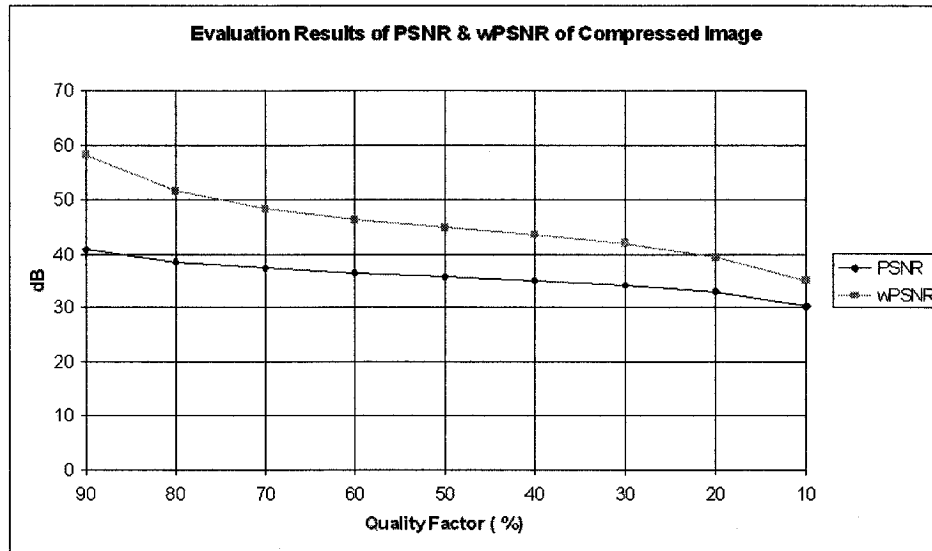


Figure 4.24: Evaluation Results of PSNR and wPSNR of Compressed Image

It is known that the image quality degrades when the quality factor is reduced. From the above chart, we can see that when the quality factor is reduced, both the PSNR value and the wPSNR value are also decreased. The shape of the wPSNR value follows the shape of the PSNR value. This proves that wPSNR value can reflect the image quality correctly. The wPSNR value does not conflict with the PSNR value.

Finally, we use wPSNR to measure 25 watermarked images which are without masking, with NVF masking and the proposed masking. For the same image, the watermarking energy is the same, the only difference is with different masking. The chart is shown in Figure 4.25:

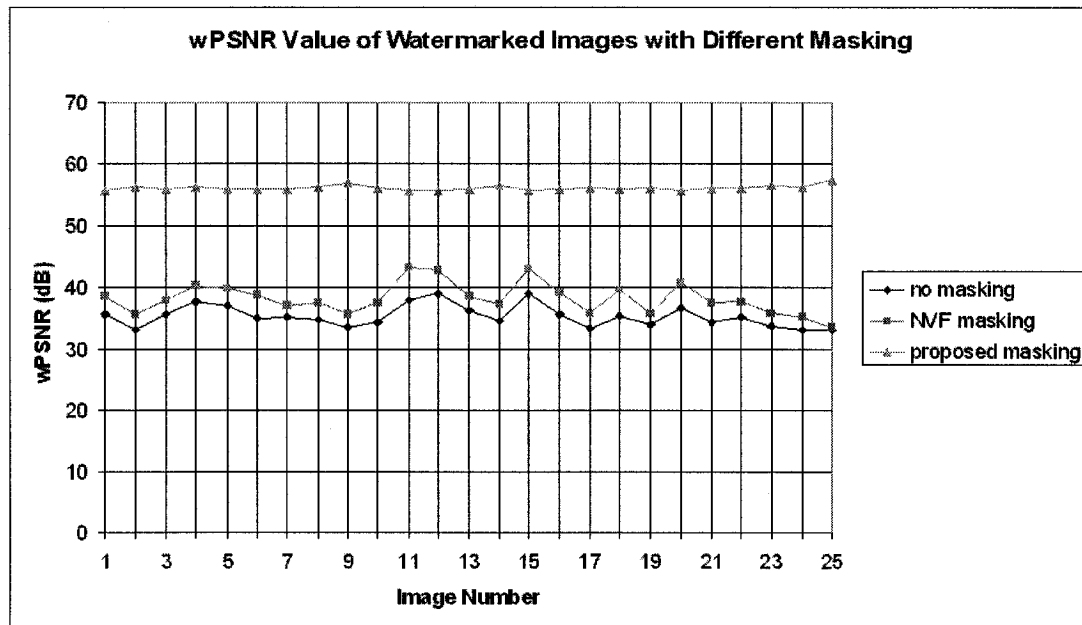


Figure 4.25: wPSNR Value of Watermarked Images with Different Masking

From Figure 4.25, it is clear that NVF Masking can help to improve the image quality, but it is not as good as the proposed masking. For some images which have large areas with darkness or brightness, the NVF masking is not suitable. Theoretically, with the proposed masking, the wPSNR value should be infinite because the watermark is always covered by the masking. It is not infinite due to the round operation because the masking values are not integers, and the final watermarked image is represented by integers.

Finally, we use wPSNR value (Table 4.1) to compare the watermarked images with the same watermark energy. The watermarked images we already showed in Section 4.2.

Table 4.1: wPSNR Value of the Quality of Different Watermarked Images

Image name	Without Masking	NVF (Non-Stationary Model)	Proposed Masking
Lena	38.9070dB	42.8331dB	55.6952dB
Work	33.0729dB	33.5407dB	57.2457dB
Discussion	33.2029dB	35.2303dB	56.2548dB

4.4 Watermark Detection

With the proposed masking method, we embed the same watermark into 25 different images, and use the linear correlation to test the image with and without watermark. The implementation result is shown in Figure 4.26:

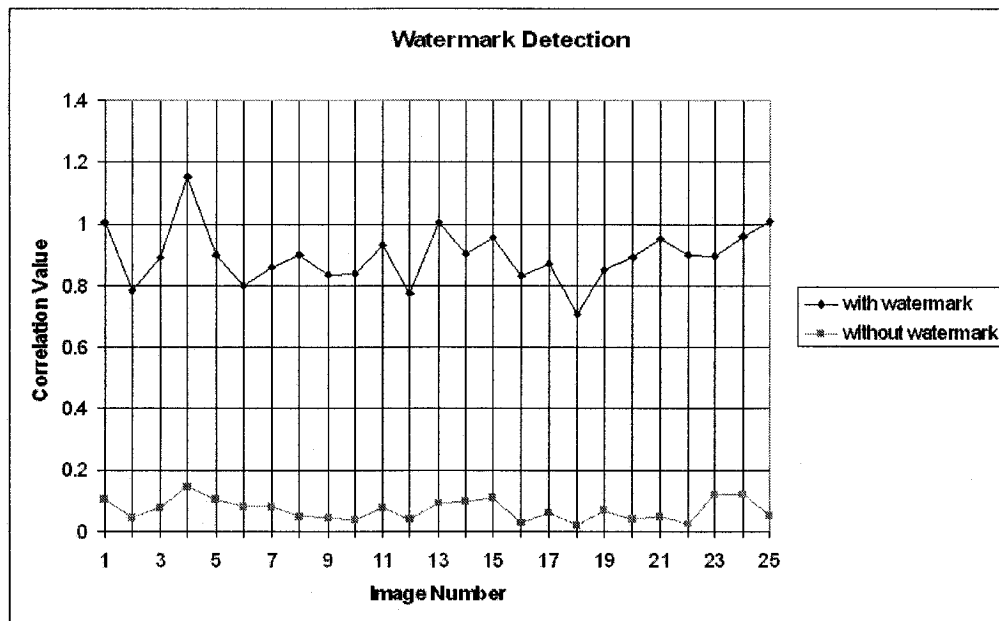


Figure 4.26: Watermarking Detection

From Figure 4.26, when the image contains the watermark, most values of the linear correlation are above 0.7. When the image has no watermark, the linear correlation results

are below 0.2. Because of the convenience of the implementation, we always use the same watermark to do the embedding in different images. we see that some values are a little bit low, this is because the original image and watermark still have some correlation. For the individual image, people can use different pseudo random seeds to create several watermarks to compute the correlation value with the original image, and select the watermark which makes the correlation value the smallest. Therefore, the final linear correlation can reach the maximum for the individual image.

When the watermarked images are processed by normal compression with different quality factors, the correlation value will be affected. Quality factor determines how much of the data will be lost and thus directly impacts the size of the compressed image. It is the amount by which quantization tables are scaled to adjust image quality. If the quality factor is 1, it specifies the lowest quality or maximum image compression. If the quality factor is 100, it specifies the highest quality or minimum image compression. Figure 4.27 shows the correlation value with different quality factors cast on the watermarked image. The quality factors are 90%, 85%, 80% and 75%.

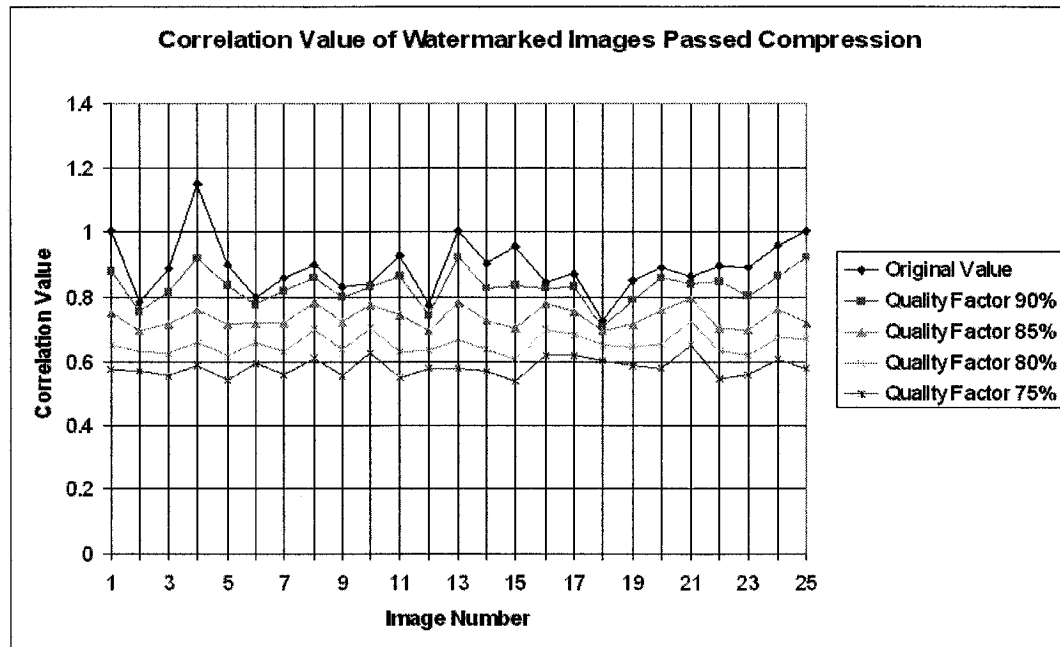


Figure 4.27: Detection Value for the Watermarked Image Passed Compression

From the results, we can see that the higher the quality factor, the higher the linear correlation value. The compression not only works on the original image, but also on the watermark which is carried with the original image. This is because when the quality factor is high, the impact of the compression on the image is small. The reduction of the watermark is small as well. As a result, the final linear correlation value is high. The shape of lines with the compression always follows the shape of the line that has no compression. When the value in the non-compression line is high, the value with the compression is also high, and vice versa. When the quality factor is 75%, most of the correlation value is above 0.5.

The experiment only tests the ability of the embedding strength to help to improve the watermarked image resist compression. If we wish the correlation value to be much higher,

there are several other methods which deal with the compression impact to the watermarked image very well [45].

Another test is the additional noise attack. In the simulation, We add different types of noise to the watermarked image. The types of the noise include the Gaussian noise, salt and pepper noise and the speckle noise. The results of the linear correlation are follows:

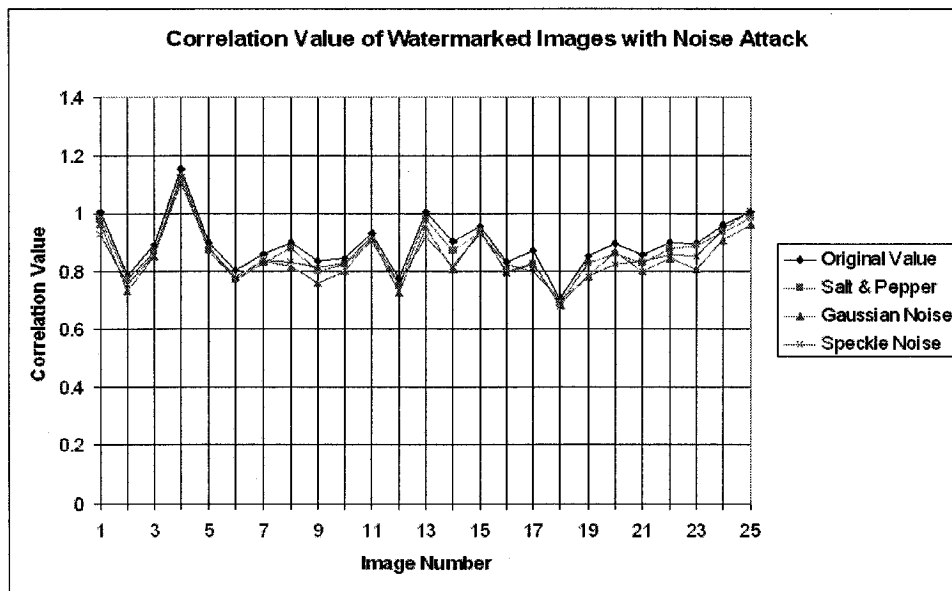


Figure 4.28: Detection Value of Watermarked Image Passed Noise Attack

From the above results, we can see that watermark is quite robust when different noise is added. This is mostly because the correlation value between the watermark and the noise is almost close to zero.

4.5 Data Hiding Application

With the proposed masking, we can use it in other watermarking algorithm such as data hiding application. In order to explain clearly, we will embed one bit 1 and -1 in one block

as an example. We extract one block with size 4×4 from image “Lena”, the value of pixels is as follows:

$$\begin{array}{cccc}
 170 & 173 & 167 & 172 \\
 173 & 172 & 175 & 180 \\
 171 & 172 & 175 & 182 \\
 177 & 175 & 180 & 178
 \end{array} \tag{4.2}$$

The minimum masking value of this block is 4. The value of watermark is as follows:

$$\begin{array}{cccc}
 1 & -1 & 1 & -1 \\
 -1 & -1 & -1 & 1 \\
 -1 & 1 & 1 & -1 \\
 1 & 1 & 1 & -1
 \end{array} \tag{4.3}$$

If we want to embed bit “1” in this block, we use the Equ.(3.17) to embed the watermark.

Then the final block will be transferred to the detection site. The value of this block is:

$$\begin{array}{cccc}
 173 & 171 & 169 & 171 \\
 171 & 171 & 175 & 181 \\
 171 & 173 & 177 & 183 \\
 177 & 177 & 181 & 179
 \end{array} \tag{4.4}$$

After the detection site receives the block, it will use the Equ.(3.19) to extract the watermark, the result of the calculation is as follows:

$$\begin{array}{cccc}
 1 & -1 & 1 & -1 \\
 -1 & -1 & -1 & 1 \\
 -1 & 1 & 1 & -1 \\
 1 & 1 & 1 & -1
 \end{array} \tag{4.5}$$

After calculating the correlation value between watermark (4.3) and extracted value (4.5), the result is 1. Therefore, the detection site will know the message bit is 1.

Similarly, we can embed the message bit “-1”. Instead of using Equ.(3.17), we use Equ.(3.18), then the final block value becomes:

$$\begin{array}{cccc}
 171 & 173 & 167 & 173 \\
 173 & 173 & 177 & 179 \\
 173 & 171 & 175 & 185 \\
 175 & 175 & 179 & 181
 \end{array}$$

After detection site receives this block, it will apply the Equ.(3.19) to recovery the watermark. The result is as follows:

$$\begin{array}{cccc}
 -1 & 1 & -1 & 1 \\
 1 & 1 & 1 & -1 \\
 1 & -1 & -1 & 1 \\
 -1 & -1 & -1 & 1
 \end{array} \tag{4.6}$$

The correlation between watermark (4.3) and extracted value (4.6) is -1. Therefore, the message bit will be -1.

Because the watermark embedding includes the plus and minus operation, we should point

out the problem of data range. If the original pixel value is lower than masking value, such as the darkness area, we use the masking value to replace the original pixel value first, and then continue the calculation. On the other hand, if the sum of the original pixel value and the masking value exceeds 255, we use the value represented by the difference of 255 and masking value to replace the original pixel value.

The implementation result shows that if there is no attack, the bit error is zero. Of course, the watermark detection site needs the HVS masking value, only one HVS masking value is required for each block. If there is some attacks or the signal processing, it will bring the bit error. In order to solve the problem, channel coding such as BCH Coding (Bose-Chaudhuri-Hocquenghem Coding) or the Reed-Solomn Coding is suggested. In addition, we are still working on how to extract the masking from the received image without the original image or keys. We also wish to determine how attacks or signal processing affect the masking.

After applying the data hiding method, the results of the watermarked images are shown. These include watermarked image of "Lena" with PSNR = 42.3204 dB (Figure 4.29), watermarked image of "Discussion" with PSNR = 34.7416 dB (Figure 4.30) and the watermarked image of "Work" with PSNR = 31.6498 dB (Figure 4.31).



Figure 4.29: Watermarked Image “Lena” in Data Hiding Application



Figure 4.30: Watermarked Image “Discussion” in Data Hiding Application



Figure 4.31: Watermarked Image “Work” in Data Hiding Application

4.6 HVS Masking Extended to the DCT Domain

Finishing the implementation of the HVS masking in the spatial domain, we wish to extend it to the DCT domain. Following the method introduced previously and using the simplified example, we only embed the watermark in the frequencies: F_{50} and F_{05} . The implementation results are as follows. The PSNR value of watermarked image “Lena” is 39.8608 dB (Figure 4.32). It also shows the normalized watermark of that image. The PSNR value of the watermarked image “Discussion” is 36.3674 dB (Figure 4.34). The watermark is also shown as follows. The PSNR value of the watermarked image “Work” is 35.2970 dB. (Figure 4.36). It also shows the watermark of that image.



Figure 4.32: Watermarked Image "Lena" with DCT Domain Masking



Figure 4.33: DCT Domain Watermark of Image "Lena"

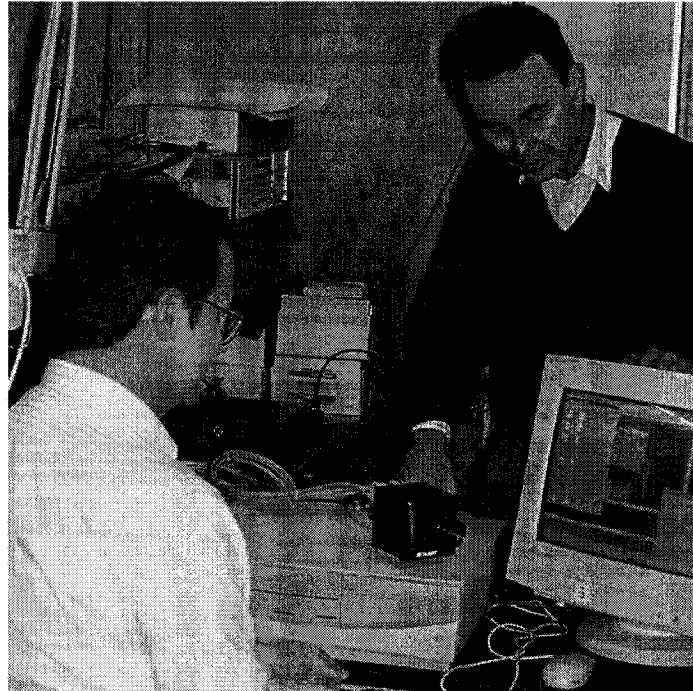


Figure 4.34: Watermarked Image “Discussion” with DCT Domain Masking

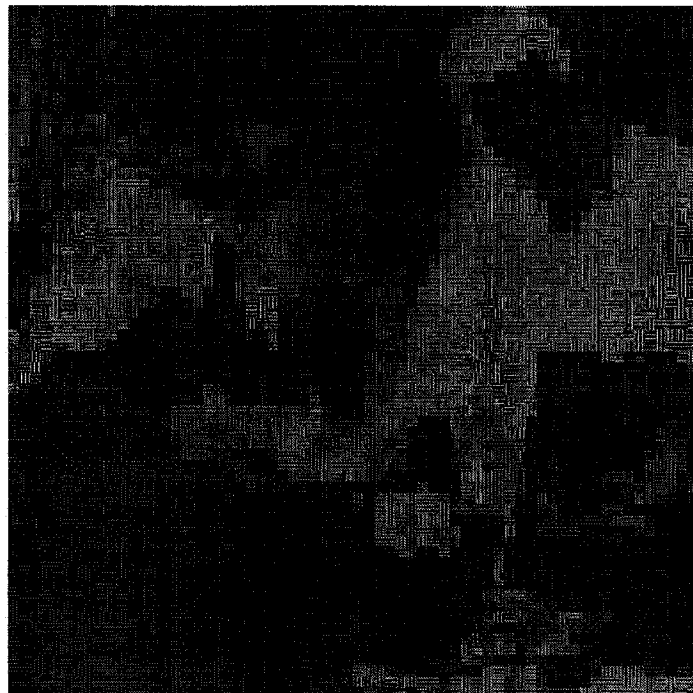


Figure 4.35: DCT Domain Watermark of Image “Discussion”



Figure 4.36: Watermarked Image “Work” with DCT Domain Masking

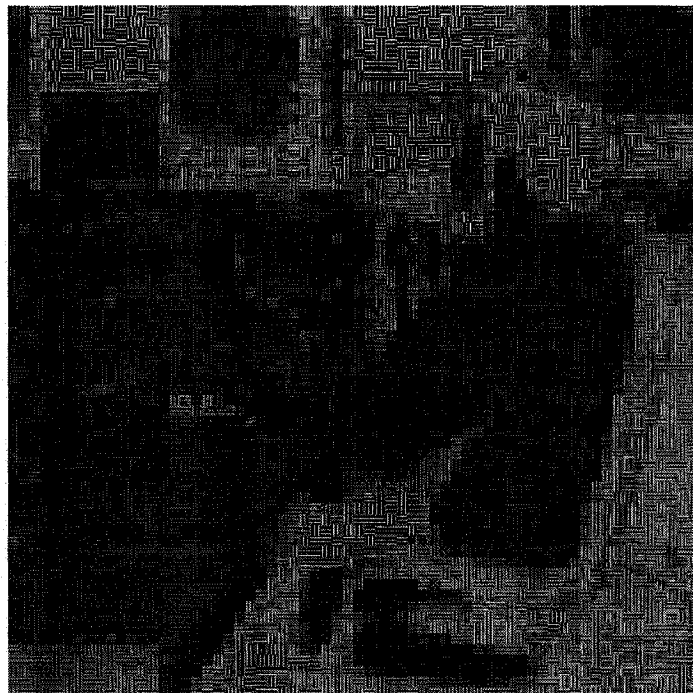


Figure 4.37: DCT Domain Watermark of Image “Work”

If we embed watermark in the frequencies F_{30} and F_{03} . The 64 expressions will be changed as follows:

$$0.25(F_{30} \cos \frac{3(2i+1)\pi}{16} + F_{03} \cos \frac{3(2j+1)\pi}{16}) \leq SM_{ij} \quad 0 \leq i \leq 7, 0 \leq j \leq 7$$

where, SM_{ij} is the same as matrix (3.25).

When we embed the watermark in the frequencies F_{30} and F_{03} , the PSNR value of watermarked image “Lena” is 39.9165 dB. The PSNR value of the watermarked image “Discussion” is 36.3756 dB. The PSNR value of the watermarked image “Work” is 35.3202 dB. We find that the value of the PSNR is similar to the the watermarked image in which the watermark was embedded in the frequencies F_{50} and F_{05} . This is due to the restricted condition represented by SM_{ij} which limits the watermark energy. No matter where the watermark will be embedded in the frequency, it will be satisfied the extreme value of the energy function and limited by the restricted condition.

It is clear that the method we introduced previously can produce the masking values for all frequencies. If we wish to embed the watermark in different frequencies, we only need to change the value of the matrix A since the calculation of the IDCT will change. After the masking is obtained, it can be used in different watermarking applications. Normally, we do not suggest using the DC component, since the DC component controls the whole block background. When the two neighbor blocks are inserted different watermark such as 1 or -1, the block artificial distortion is quite obvious. Even with the value 0.006 as the embedding strength which is suggested in [28], the distortion is still not acceptable.

Chapter 5

Conclusion and Future Work

In the previous chapters, we introduced the digital watermarking system and its properties, mechanism of the human visual system and different human visual maskings used in the digital watermarking technology. After we analyzed these maskings' advantages and disadvantages, we proposed a new spatial HVS masking which not only addresses the luminance factor, but also the texture and edges of the image. From experimental results, the values of wPSNR prove that with the proposed masking, the visual quality of the watermarked image is dramatically improved. We also suggest using noise shaping technology to deal with the watermarking detection. In addition, the proposed masking also can be merged into other watermarking algorithms, such as data hiding, which requires the masking to control the final visual quality. Finally, we extended this HVS masking into the DCT domain, and made the DCT domain masking restricted by the spatial masking.

As for future work, the estimation of the HVS masking in the detection site still needs more effort. The effects that noise attack or compression gives to the HVS masking is still complex. If the precise extraction of the HVS masking from watermarked image is possible, it will save the transmission of the key parameters. This will reduce the risk

of the information leaking and make the whole watermarking algorithm more robust. In addition, we also should extend this HVS masking to the wavelet domain. Because many watermarking algorithms happen in wavelet domain, this will help to broaden the HVS masking's applications.

Bibliography

- [1] F. Hartung and M. Kutter, "Multimedia Watermarking Techniques", Proceedings of the IEEE, Vol. 87, No.7, pp. 1079-1107, July 1999.
- [2] I. J. Cox, M. L. Miller, and J. A. Bloom, "Digital Watermarking", Morgan Kaufmann, 2001.
- [3] A. Nikolaidis and I. Pitas, "Region-Based Image Watermarking", IEEE Transactions on Image Processing, Vol. 10, No. 11, pp. 1726-1740, November 2001.
- [4] G. Voyatzis and I. Pitas, "Chaotic Watermarks for Embedding in the Spatial Digital Image Domain", Proceedings of IEEE International Conference on Image Processing, Vol. 2, pp. 432-436, October 1998.
- [5] J. Zhen, W. Xiao, and J. Zhang, "A Multiple Digital Watermarking Algorithm Based on Chaotic Sequences", Proceedings of SPIE, Vol. 4793, Mathematics of Data/Image Coding, Compression, and Encryption with Applications, pp. 287-292, January 2003.
- [6] C. Lin and S. Chang, "Watermarking Capacity of Digital Images Based on Domain-Specific Masking Effects", Proceedings of the IEEE International Conference on

- Information Technology: Coding and Computing, pp. 90-94, Las Vegas, NV, April 2001.
- [7] S. D. Servetto, C. I. Podilchuk and K. Ramchandran, "Capacity Issues in Digital Image Watermarking", Proceedings of International Conference on Image Processing, Vol. 1, pp. 445-449, Chicago, IL, October 1998.
- [8] J. F. Delaigle, C. Devleeschouwer, B. Macq, and I. Langendijk, "Human Visual System Features Enabling Watermarking", Proceedings of IEEE International Conference Multimedia Exposition, Vol. 2, pp. 489-492, 2002.
- [9] B. A. Wandell, "Foundation of the Vision", Sinuaer Associates Inc. Sunderland, Massachussetts, Stanford University edition, 1995.
- [10] M. Barni and F. Bartolini, "Watermarking Systems Engineering, Enabling Digital Assets Security and Other Applications", ISBN 0-8247-4806-9, 2004.
- [11] A. N. Netravail and B. G. Haskell, "Digital Pictures: Representation and Compression", New York, Plenum, 1988.
- [12] C. Chou and Y. Li, "A Perceptually Tuned Subband Image Coder Based on the Measure of Just-Noticeable-Distortion Profile", IEEE Transactions on Circuits and Systems for Video Technology, Vol. 5, No. 6, pp. 467-476, December 1995.
- [13] M. Kutter and S. Winkler, "A Vision-Based Masking Model for Spread Spectrum Image Watermarking", IEEE Transactions on Image Processing, Vol. 11, No. 1, pp. 16-25, January 2002.
- [14] E. Peli, "Contrast in Complex Images", Journal of the Optical Society of America A, Vol. 7, No.10, pp. 2032-2040, 1990.

- [15] S. Winkler and P. Vandergheynst, "Computing Isotropic Local Contrast from Oriented Pyramid Decompositions", Proceedings of the IEEE International Conference on Image Processing, Vol. 4, pp. 420-424, Kyoto, Japan, 1999.
- [16] D. Levicky and P. Foris, "Human Visual System Model in Digital Image Watermarking", Radio Engineering, Vol. 13, No. 4, pp. 38-43, December 2004.
- [17] P. Chen, "On the Study of Watermarking Application in WWW- Modeling, Performance Analysis, and Application of Digital Image Watermarking Systems", Master Thesis, May 1999.
- [18] G. M. Johnson and M. D. Fairchil, "On Contrast Sensitivity in an Image Difference Model", International Technical Conference on Digital Image Capture and Associated System, Reproduction and Image Quality Technologies, pp. 18-23, Portland, April 2002.
- [19] S. Voloshynovskiy, A. Herrigel, N. Baumgaertner, and T. Pun, "A Stochastic Approach to Content Adaptive Digital Image Watermarking", Third International Workshop on Information Hiding, (Dresden, Germany), pp. 212-236, September 29-October 1st, 1999.
- [20] J. F. Delaigle, C. D. Vleeschouwer, and B. Macq, "Watermarking Algorithm Based on A Human Visual Model", Signal Processing, Vol. 66, No. 3, pp. 319-335, 1998.
- [21] B. T. Hannigan, A. Reed, and B. Bradley, "Digital Watermarking Using Improved Human Visual System Model", Processings of SPIE, Electronic Image, Security and Watermarking of Multimedia Content, Vol. 4314, pp. 468-474, San Jose, CA, August 2001.

- [22] F. Bartolini, M. Barni, V. Cappellini and A. Piva, "Mask Building for Perceptually Hiding Frequency Embedded Watermarks", Proceedings 5th IEEE International Conference on Image Processing, ICIP98, Vol. 1, pp. 450-454, Chicago, IL, USA, October 1998.
- [23] S. Voloshynovskiy, S. Pereira and T. Pun, "Watermark Attacks", Erlangen Watermarking Workshop, October 5-6, 1999.
- [24] R. Liu, T. Tan, "Theoretical Framework for Watermark Capacity and Energy Estimation", Processings of SPIE, Vol. 4675, pp 433-440, Security and Watermarking of Multimedia Contents IV, April 2002.
- [25] M. Miyahara, K. Kotani, and V. R. Algazi, "Objective Picture Quality Scale for Image Coding", IEEE Transactions on Communications, Vol. 46, No. 9, pp. 1215-1226, September 1998.
- [26] D. Taskovski, "Adaptive Watermarking For Digital Images Protection", PhD Thesis. Faculty of Electrical Engineering, Skopje, June 2004.
- [27] D. Taskovski, S. Bogdanova, and M. Bogdanov, "Blind Low Frequency Watermark Method", International Journal of Signal Processing, Vol 2, No. 2, pp. 146-150, 2005.
- [28] J. Huang, Y. Q. Shi and Y. Shi, "Embedding Image Watermarks in DC Components", IEEE Transactions on Circuits and Systems for Video Technology, Vol. 10, No. 6, pp. 974-979, 2000.
- [29] C. I. Podilchuk and W. Zeng, "Image Adaptive Watermarking Using Visual Model", IEEE Journal on Selected Areas in Communications, Vol. 16, No. 4, pp. 525-539, May 1998.

- [30] A. B. Watson, "Visual Optimization of DCT Quantization Matrices for Individual Images", American Institute of Aeronautics and Astronautics, San Diego, CA, pp. 286-291, 1993.
- [31] I. J. Cox, J. Kilian, F. T. Leighton, and T. Shamoan, "Secure Spread Spectrum Watermarking for Multimedia", IEEE Transactions on Image Processing, Vol. 6, No. 12, pp. 1673-1689, December 1997.
- [32] B. Chen, "Design and Analysis of Digital Watermarking, Information Embedding, and Data Hiding Systems", PhD thesis, MIT, June 2000.
- [33] J. Wolosewicz and K. Jemili, "Apparatus and Method for Encoding and Decoding Information in Analog Signals", United States Patent #5, 828, 325, October 1998.
- [34] M. D. Swanson, B. Zhu, and A. Teefik, "Robust Data Hiding for Images", Proceedings of 7th IEEE Digital Signal Processing Workshop, pp. 37-40, (Loen, Norway), September 1996.
- [35] Z. Wang and A. C. Bovik, "A Human Visual System Based Objective Video Distortion Measurement System", Proceedings of International Conference on Multimedia Processing and Systems, August 2001. <http://www.cns.nyu.edu/~zwang/files/papers/icmps.pdf>
- [36] <http://www.unm.edu/~keithw/astrophotography/imagesharpener.html>
- [37] J. C. Candy and G. C. Temes, "Oversampling Delta-Sigma Data Converters: Theory, Design and Simulation", ISBN 0-87942-285-8, 1992.
- [38] J. Herre and J. D. Johnston, "Enhancing the Performance of Perceptual Audio Coding by Using Temporal Noise Shaping (TNS)", 101st Audio Engineering Society (AES) Convention. preprint #4384, Los Angeles, CA, November 1996.

- [39] S. Kuo, and J. D. Johnston, "Spatial Noise Shaping Based on Human Visual Sensitivity and Its Application to Image Coding", IEEE Transactions on Image Processing, Vol. 11, No. 5, pp. 509-517, May 2002.
- [40] M. Athineos and D. P.W. Ellis, "Frequency Domain Linear Prediction for Temporal Features", Proceedings of Speech Recognition and Understanding Workshop, IEEE ASRU 2003, pp. 261-266, St.Thomas, December 2003.
- [41] <http://www.mathworks.com/access/helpdesk/help/toolbox/vipblks/2ddct.html>
- [42] <http://www.mathworks.com/access/helpdesk/help/toolbox/vipblks/2didct.html>
- [43] T. F. Coleman and Y. Li, "A Reflective Newton Method for Minimizing a Quadratic Function Subject to Bounds on some of the Variables," SIAM Journal on Optimization, Vol. 6, No. 4, pp.1040-1058, 1996.
- [44] <http://www.mathworks.com/access/helpdesk/help/toolbox/images/morph.html>
- [45] C. Lu, S. Huang, C. Sze and H. M. Liao, "Cocktail Watermarking for Digital Image Protection", IEEE Transactions on Multimedia, Vol. 2, No. 4, pp. 209-223, December 2000.