GEOMETRIC DISTORTION IN IMAGE AND VIDEO WATERMARKING

Robustness and Perceptual Quality Impact

Proefschrift

ter verkrijging van de graad van doctor
aan de Technische Universiteit Delft,
op gezag van de Rector Magnificus prof. dr. ir. J. T. Fokkema,
voorzitter van het College voor Promoties
in het openbaar te verdedigen op maandag 14 juni 2004 om 10.30 uur

door

Iwan SETYAWAN

Master of Science in Electrical Engineering
geboren te Semarang, Indonesië

Dit proefschrift is goedgekeurd door de promotor:

Prof. dr. ir. R.L. Lagendijk

Samenstelling promotiecomissie:

Rector Magnificus,	voorzitter
Prof. dr. ir. R.L. Lagendijk,	Technische Universiteit Delft, promotor
Prof. dr. ir. H.J. Sips,	Technische Universiteit Delft
Prof. dr. ir. J. Biemond,	Technische Universiteit Delft
Prof. dr. H. De Ridder,	Technische Universiteit Delft
Prof. dr. ir. L.J. van Vliet,	Technische Universiteit Delft
Prof. dr. A.A.C.M. Kalker	Tecnhische Universiteit Eindhoven
Prof. dr. B. Macq,	Université Catholique de Louvain, België

Geometric Distortion in Image and Video Watermarking:
Robustness and Perceptual Quality Impact
Setyawan, Iwan
Thesis Delft University of Technology – with ref. – with summary in Dutch

Advanced School for Computing and Imaging

This work was carried out in the ASCI graduate school.
ASCI dissertation series number 100.

ISBN 90-9017826-0

To my beloved Christina
To my parents

CONTENTS

Chapter 1
INTRODUCTION

1.1. Digital watermarking and its challenges

The use of digital media as the primary means by which to distribute material such as images, audio material and video is becoming more and more common [1]. Digital media offers ease of use for both users and those involved in the creation of the material. For example, digital audio players are generally more compact than their analog counterparts. Also, such devices can offer extra capabilities, by, for example, serving as a removable storage device, which further increases their appeal. For content creators, digital media offers a more convenient way of creating and manipulating images, audio material and video. Another advantage of digital media is the higher quality they offer compared to analog media. Additionally, the quality of the material stored in a digital medium will not decay with time as is the case with analog storage. Finally, with the advent of broadband internet use for home users, digital media offer a new distribution model for content providers. Using this model, the content provider distributes its product digitally through the internet, thus reducing the overhead costs related to product duplication and distribution. Recent examples of such models are Apple's iTunes and Napster 2.0.

However, all the advantages offered by digital media can also be abused, for example by unauthorized reproduction or alteration of the digital content. The ease with which digital content can be reproduced without loss of quality forms a potential loss of revenue for content providers. The availability of broadband internet to many users also facilitates the illegal distribution of copyrighted material. This is evident in the current popularity of various peer-to-peer file-sharing networks on the internet. The ease with which digital image or video can be manipulated or altered also gives rise to another concern. For example, someone can alter or create faked images in order to damage the reputation of a person or institution. Another example is the alteration of images or video (for example, taken from a security camera) which are being used as evidence in a court of law. The alteration can influence the court's decision in favor of one or more of the parties involved in the case.

Digital watermarking techniques were born in reaction to the aforementioned misuse of digital media. For example, a content provider which wants to prevent unauthorized use of its copyrighted material can embed a watermark carrying a copyright notice in the material. When an unauthorized copy of the material is found, the content provider can assert its copyright using the embedded watermark. Alternatively, a content provider may want to prevent this unauthorized reproduction in the first place. In this scenario, the embedded watermark carries information that will disable copying operations in compliant devices. In another scenario, a content provider may want to track down the parties responsible for unauthorized copying. In this case, the content provider can embed a unique watermark associated with each user. When an unauthorized copy is found, the content provider can then track down and prosecute the user whose watermark is found in the illegal copies. Finally, watermarks can also be used to ensure the authenticity of a digital image or video. Take, for example, the image depicted in Figure 1.1. Is this a real picture or is this a tampered ("doctored") picture? For this scenario, the watermark can be designed such that it can identify the portions of the image that have been tampered with. A user receiving a tampered image will then be able to detect the tampering as well as identify the parts of the image that have been tampered with.

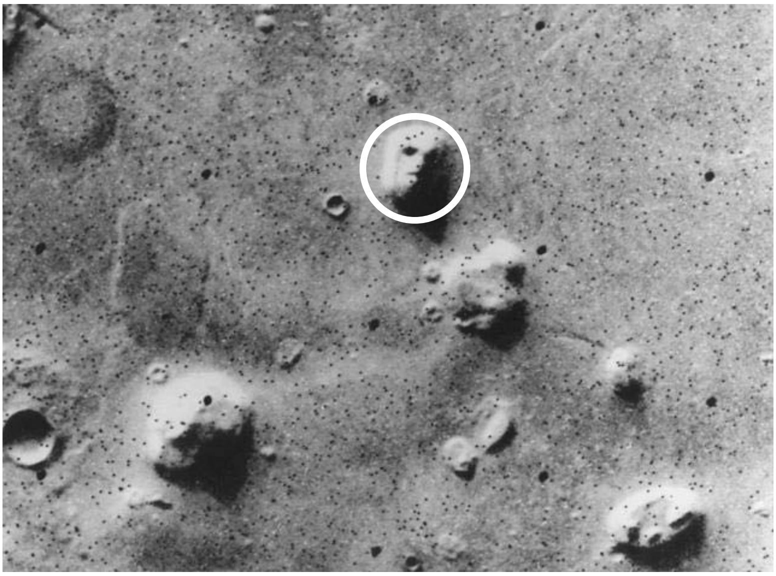


Figure 1.1. Is this an authentic image of Mars?
Or has it been "doctored"? (Picture © NASA)

Digital watermarking systems, in particular watermarked data, may encounter a lot of types of distortions collectively known as *attacks*. Such attacks may either be intentional or non-intentional. A non-intentional attack refers to common operations performed by a legitimate user without any intention to actually harm the watermark. For example, a user may want to resize an image to fit the desktop of his/her computer. Another example is a user who compresses an image to save disk-space. On the other hand, intentional attacks are expressly performed to remove the watermark or disable watermark detection. Such attacks are usually very elaborate and may employ advanced techniques. Robustness against attacks, among other things, is thus a very important factor to be taken into account in the design of a watermarking system.

1.2. Focus of the thesis

As suggested by its title, this thesis discusses mainly the challenges of dealing with geometric distortion in image and video watermarking. However, this is not the only topic explored in this thesis. Another topic that we also discuss is the challenge of embedding watermarks in video data compressed at a low bit-rate.

1.2.1. Low bit-rate compression and watermarking

Data compression schemes work by removing redundancy from the data. The part of the data that is considered to be redundant is usually the part that does not affect the perceptual quality of the data. The lower the compression bit-rate, the larger the amount of redundant data that will have to be removed. This part of the data is the logical place to embed the watermark for perceptibility reasons. Thus the removal of this part of the host data by the compression scheme may not leave enough space in which the watermark can be embedded. Therefore, the watermarking system must be carefully designed in order to maintain watermark imperceptibility, while achieving acceptable watermark capacity and robustness. One possible solution is to reduce the energy of the watermark signal while maintaining watermark capacity. The consequence of this solution is that the robustness of the watermark will suffer. Alternatively, the watermark signal can be spread more widely over the host data. The consequence of this solution is that the watermark capacity will be reduced. In some cases, both solutions may have to be implemented, as in the case of the watermarking algorithm discussed in this thesis.

1.2.2. The geometric distortion problem

Geometric distortion is one of the most challenging problems in watermarking. Geometric distortion can happen due to the deliberate application of a geometric transformation or operation to a (digital) image or video. Such an operation can be simple, for example, rotating an image by a few degrees. It can also be very sophisticated, for example, by applying complex combinations of several geometric transformations. Figure 1.2 shows an example of geometric distortion caused by applying geometric transformations to an image. Geometric distortion can also happen as a by-product of other operations performed on the image or video. For example, the process of scanning or printing an image can introduce geometric distortions due to the imperfections of the scanner or printer. Another example is the Digital Cinema Attack [7]. In this scenario, an attacker uses a hand-held video camera to record a movie being shown in a digital cinema. The overhead view of this scenario is shown in Figure 1.3. In this example, we see that the attacker is recording the movie from an angle α relative to the center of the cinema screen.

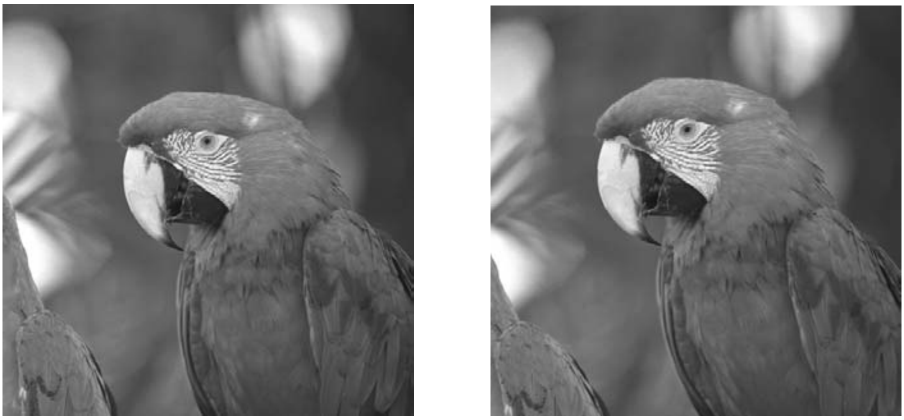


Figure 1.2. Example of geometric distortion:
(a) Original image and (b) Image distorted using random bending

Due to the relative position of the camera to the movie screen, the recorded video will suffer geometric distortion. This is shown in Figure 1.4. In this example, we assume that the attacker is making the recording from a position to the lower left side of the cinema screen. Figure 1.4(a) shows the original image shown on the cinema screen while Figure 1.4(b) shows the image recorded by the attacker. If the attacker wants to sell this recording, he can remove the annoying black portions of Figure 1.4(b) by cropping them, giving him the image shown in Figure 1.4(c).

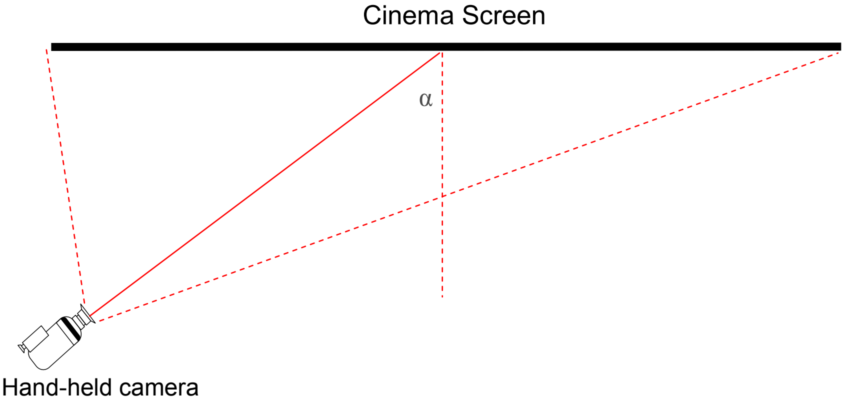
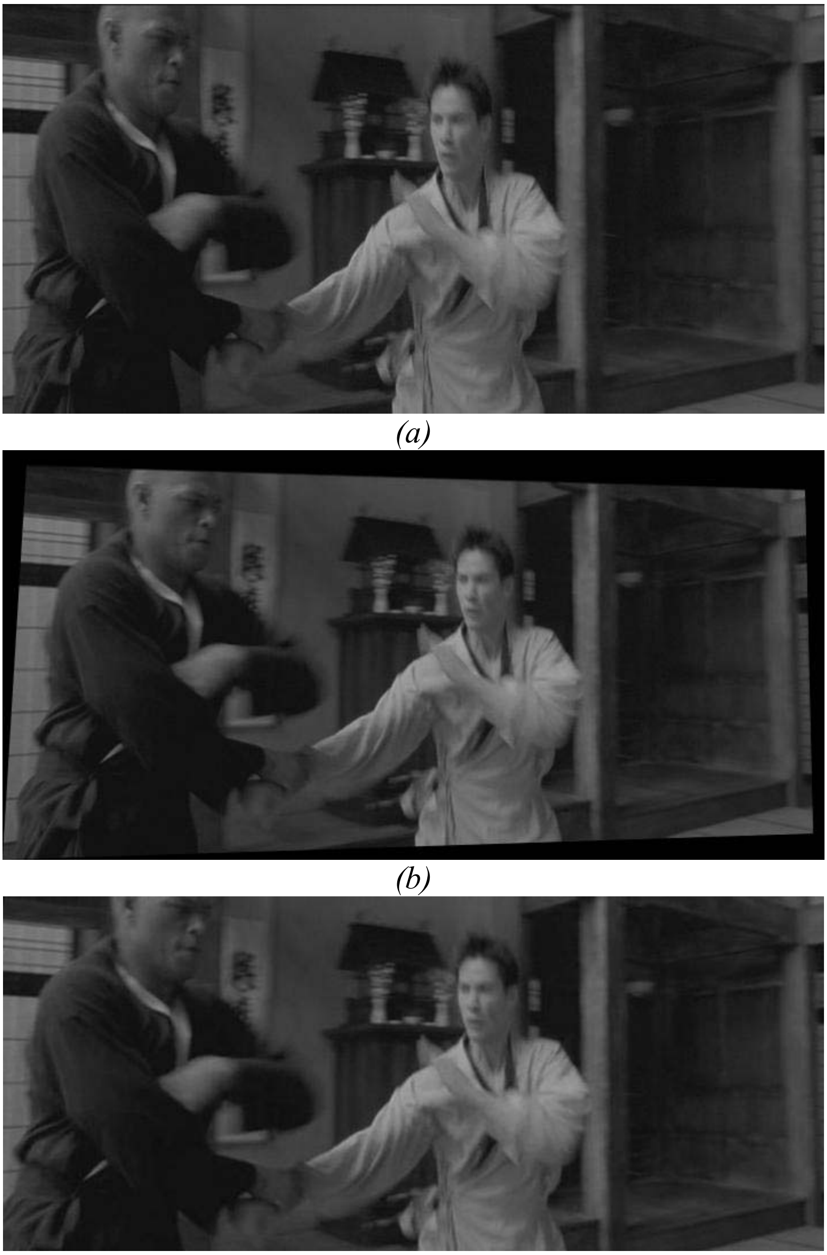


Figure 1.3. Digital Cinema Attack



(a)

(b)

(c)

Figure 1.4. Results of the Digital Cinema Attack:
(a) Original image, (b) Recorded image and (c) Cropped recorded image
© 1999 Warner Brothers

Geometric distortion forms a problem for the designers of watermarking systems because it is relatively easy to perform while it is difficult to combat.

Geometric distortion can prevent the proper detection of the watermark while preserving the perceptual quality of the attacked data. For example, the geometric distortion applied to the image shown in Figure 1.2(a) can prevent watermark detection in most watermarking systems. However, the attacked image, shown in Figure 1.2(b), still has very high perceptual quality. In other words, most people will not find the distortion objectionable. Many people may not even notice the distortion at all.

Geometric distortion in watermarking systems has two aspects, namely the *watermark desynchronization aspect* and the *perceptual quality aspect*. These two aspects are briefly discussed as follows:

- ***Watermark desynchronization aspect.*** Geometric distortion does not actually remove the embedded watermark. It prevents the detection of the watermark by disturbing the *synchronization* between the watermark and the watermark detector. Applying geometric operations to the watermarked data is equivalent to changing the sampling grid of the watermark and thus making it different from the sampling grid of the watermark detector. Upon detection, the detector will fail to detect the watermark properly, not because the watermark has been removed, but because the detector can no longer find the location of the watermark. This aspect of geometric distortion has been widely studied in the literature. Research in this area has resulted in watermarking schemes that are invariant to geometric distortions or schemes that can resynchronize the watermark after a geometric distortion. Watermark resynchronization can be done either by using the host data as a reference or by inserting synchronization patterns into the watermark. The first approach is called a *non-blind* approach while the second approach is referred to as a *blind* approach. In this thesis, we discuss the design of a *non-blind* approach to resynchronize a watermark after a geometric distortion. We also propose another approach to the synchronization problem by designing a watermarking system for image and video that does not rely on strict spatial synchronization between the watermark and the watermark detector. This system is based on structured noise patterns and offers a better robustness to geometric distortions compared to conventional noise-based watermarking systems.
- ***Perceptual quality aspect.*** This aspect of the geometric distortion problem is a challenging issue that has not been widely studied in the literature. The main consequence of this fact is that we do not have any suitable objective system to measure the perceptual impact of geometric distortion on a human observer. Existing objective quality measurement schemes, for example the widely used PSNR measurement, are not suitable for measuring the impact of geometric distortion on the perceptual quality of the image or video. For example, the PSNR value between the original image shown in Figure

1.3(a) and the distorted image shown in Figure 1.3(b) is very low, but most people will not find the distortion disturbing. The lack of objective measurement systems gives us two problems. In the first place, we cannot determine the level of distortion that humans can still tolerate. If we could determine this level, we would be able to optimize the performance of the watermarking system so that the watermark would survive this level of distortion. The second problem is that without an objective quality measurement system, it is very difficult to compare the robustness of various watermarking systems against geometric distortions. With an objective quality measurement system, we would be able to set a common standard with which to measure the performance of the watermarking systems. In this thesis, we propose a new system that enables us to perform objective perceptual quality measurement on geometrically distorted images.

1.3. Overview of Chapters

Most chapters in this thesis have been previously published as conference papers. Consequently, a slight overlap between chapters, especially in the introductory sections of the chapters, is inevitable.

In Chapter 2, we present a more detailed discussion of the basic principles of watermarking techniques. In this chapter, we will also discuss the various attacks typically encountered by watermarking systems.

In Chapter 3, we discuss the challenge of embedding watermarks into low bit-rate video data by presenting our Extended Differential Energy Watermarking (XDEW) algorithm [2].

In Chapter 4, we discuss the first aspect of the geometric distortion problem, namely the watermarking desynchronization aspect. In this chapter, we propose two approaches to deal with watermark desynchronization. The first approach is a watermarking algorithm for image and video that does not require strict spatial synchronization [3]. The second approach we present in this chapter is a watermarking system that allows us to re-synchronize the embedded watermark after a geometric distortion is applied [4].

In Chapters 5 and 6, we discuss the second aspect of the geometric distortion problem, namely the perceptual quality measurement of geometrically distorted images. In Chapter 5, we propose a numerical measurement system to characterize geometric distortions applied to images [5]. In this chapter, we discuss the hypothesis underlying the measurement algorithm and the details of its implementation. In Chapter 6, we describe the design and implementation of a user test to obtain subjective perceptual quality scores of geometrically distorted images [6]. The results of this test are further used to validate the measurement scheme presented in Chapter 5.

Finally, in Chapter 7, we summarize our results and provide our conclusions. Furthermore, we also provide an outlook for future research in this area.

1.4. Main contributions

The main contributions of this thesis can be summarized as follows:

- A watermarking scheme, the XDEW, suitable for MPEG1 and MPEG2 video encoded in low bit-rate (128 – 768 kbps) has been proposed and evaluated [2].
- A new approach to the watermark synchronization aspect of the geometric distortion problem has been presented. This new approach removes the need for strict spatial synchronization between the watermark and the detector by using colored noise patterns [3]. This scheme has higher robustness to geometric distortion compared to classic noise-based watermarking systems.
- A new complexity-scalable strategy to re-synchronize the watermark after a geometric attack has been presented [4]. Implementation of this strategy on top of an existing watermarking scheme can increase its robustness to geometric distortion.
- A new algorithm to provide a numerical measure to characterize the geometric distortion applied to an image has been presented [5]. To validate the algorithm, a user test to study human perception of geometric distortion in images has been implemented and analyzed [6]. The results show that the new algorithm has a much better correspondence to human perception of geometric distortion in images compared to the commonly used PSNR measurement.

1.5. References

1. G.C. Langelaar, I. Setyawan, R.L. Lagendijk, *Watermarking image and video data: A state-of-the-art overview*, IEEE Signal Processing Magazine, September 2000, Vol. 17, No. 5, ISSN 1053-5888, pp. 20 – 46
2. I. Setyawan, R.L. Lagendijk, *Low bit-rate video watermarking using temporally extended Differential Energy Watermarking (DEW) algorithm*, in Proceedings of SPIE, Security and Watermarking of Multimedia Contents III, Vol. 4314, pp. 73-84, San Jose, CA, 2001
3. I. Setyawan, G. Kakes, R.L. Lagendijk, *Synchronization-insensitive video watermarking using structured noise pattern*, in Proceedings of SPIE, Security and Watermarking of Multimedia Contents IV, Vol. 4675, pp. 520 – 529, San Jose, CA, 2002
4. P.J.O. Doets, I. Setyawan, R.L. Lagendijk, *Complexity-scalable compensation of geometrical distortions in image watermarking,* in Proceedings of IEEE, ICIP 2003, Vol. I, Barcelona, 2003
5. I. Setyawan, D. Delannay, B.M. Macq, R.L. Lagendijk, *Perceptual quality evaluation of of geometrically distorted images using relevant geometric transformation modeling*, in Proceedings of SPIE, Security and Watermarking of Multimedia Contents V, Vol. 5020, pp. 85 – 94, Santa Clara, CA, 2003
6. I. Setyawan, R.L. Lagendijk, *Human perception of geometric distortions in images*, to appear in Proceedings of SPIE, Security, Steganography and Watermarking of Multimedia Contents VI, Vol. 5306, San Jose, CA, 2004
7. D. Delannay, J.-F. Delaigle, B. Macq, *Compensation of Geometrical Deformations for Watermark Extraction in the Digital Cinema Application*, in Proceedings of SPIE, Security and Watermarking of Multimedia Contents III, Vol. 4314, pp. 149 – 157, San Jose, CA, 2001

Chapter 2
DIGITAL WATERMARKING: PRINCIPLES AND ATTACKS

2.1. Introduction

This chapter provides a short overview of the basic principles of digital watermarking, its applications and a summary of the basic requirements that a watermarking scheme should fulfill. To illustrate the implementation of the basic principles of digital watermarking, this chapter also provides a simple image watermarking scheme as an example. Finally, attacks are an ever-present concern for the designers and users of digital watermarking schemes. These attacks can take various forms and are targeted at various components of a digital watermarking system. In this chapter, we present a classification of attacks commonly encountered by current digital watermarking systems.

This chapter is organized as follows. In Section 2.2, we present the basic principles of digital watermarking. In Section 2.3, we present some applications of digital watermarking. The requirements of a digital watermarking scheme is presented in Section 2.4. A simple example of a digital image watermarking technique is presented in Section 2.5. In Section 2.6, we present common digital watermarking attacks. In Section 2.7, we present the information-theoretical approach to digital watermarking. Finally, in Section 2.8, we present our concluding remarks.

2.2. Digital watermarking: Basic principles

Digital watermarking is a method of embedding information into digital data, for example, digital images, audio or video data. The data into which the watermark is to be embedded is usually referred to as the *host data*. The information is embedded into the host data by performing alterations to the content of the host data. A generic watermarking system is shown in Figure 2.1.

This chapter is based in part on *Watermarking image and video data: A state-of-the-art overview*, by G.C. Langelaar, I. Setyawan and R.L. Lagendijk, in IEEE Signal Processing Magazine, September 2000, Vol. 17, No. 5, ISSN 1053-5888

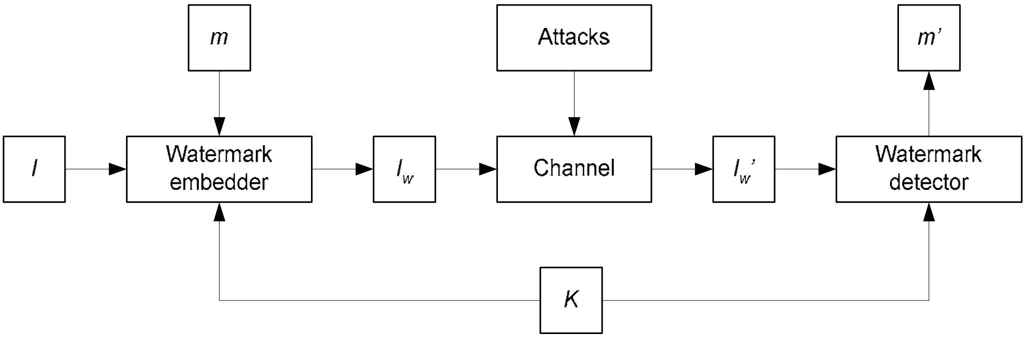


Figure 2.1. Generic watermarking system

The information to be embedded, *m*, is encoded and embedded into the host data *I* by the watermark embedder. A secret key *K* can be used if necessary, so that unauthorized parties cannot read the embedded message. After the embedding process, we have the watermarked data I_w. For visual data (images, video or documents) the embedded information can be embedded such that humans can see it without requiring any special processing of the watermarked data I_w. Such a watermark is called a *visible* watermark. Alternatively, when the watermark is designed so that humans cannot see it, we have an *invisible* watermark. In most cases, an invisible watermark is more preferable than a visible one since a visible watermark is considered to interfere with the content of the host data.

The watermarked data then goes through a transmission channel that may introduce distortions due to attacks, producing the received data I_w'. The watermark can be designed to be able to withstand these distortions; such a watermark is called a *robust* watermark. Alternatively, some scenarios may require that any distortions applied to the watermarked data should destroy the watermark. In this case we have a *fragile* watermark. The watermarked data is then passed to the watermark detector. The detector declares the presence or absence of the watermark or extracts the (probably distorted) embedded message *m'*. If the detector requires the presence of the original host data *I* in the watermark detection process, we call the watermarking system a *non-blind* watermarking system. Alternatively, if the original host data is not needed for watermark detection we have a *blind* watermarking system. Generally, the watermark will not be removed from the watermarked data after detection. However, an emerging class of watermarking techniques called *reversible* watermarking [1] is designed to enable removal of the watermark. In effect, this technique allows retrieval of the original data *I*.

In this thesis, we restrict our discussion to a particular class of watermarking systems in which the embedded data is designed to be imperceptible, the watermark is required to be robust to attacks and the watermark is not removed after detection. We call this a *robust invisible watermark.*

2.3. Digital watermarking: Applications

In the past decade, there has been an explosion in the use and distribution of digital multimedia data. Personal computers with (broadband) internet connections have become more and more common, and have made the distribution of multimedia data and applications much easier and faster. Electronic commerce applications and on-line services are rapidly being developed. Even the analog home audio and video equipment are rapidly being replaced by digital successors. As a result, digital mass recording devices for multimedia data are entering today's consumer market. Digital data has many advantages over analog data. However, it also opens the possibility of unrestricted duplication and manipulation of copyrighted material.

To prevent the unauthorized access or manipulation of digital multimedia data, two complementary techniques can be used, namely encryption and watermarking [2]. Encryption techniques can be used to protect digital data during the transmission from the sender to the receiver [3]. However, after the receiver has received and decrypted the data, the data is identical to the original data and no longer protected. Watermarking techniques can complement encryption by embedding a secret imperceptible signal, a watermark, directly into the original data in such a way that it always remains present. Such a watermark can, for instance, be used for the following purposes [7]:

- **Copyright protection:** A watermark is used to carry copyright information as a proof in case of a copyright or ownership dispute.
- **Fingerprinting:** Unique information, directly coupled to user identification, is embedded in the data as a watermark. In case of copyright violation, this watermark can be used to trace the source of illegal copies.
- **Copy protection:** A watermark is used to carry information prohibiting copying of protected data on compliant hardware.
- **Broadcast monitoring:** A watermark is embedded into data, for example, commercials or copyrighted materials [4], to allow automatic monitoring of the data in the broadcasting channels. The results of this monitoring can be used for royalty or copyright protection purposes.

Digital watermarking can also be used in other applications not dealing with copy or copyright protection:

- **Indexing:** Indexing of video mail, where comments can be embedded in the video content; indexing of movies and news items, where markers and comments can be inserted that can be used by search engines.
- **Medical safety:** Embedding the date and the patient's name in medical images could be a useful safety measure.
- **Data embedding:** Watermarking techniques can be used to embed messages in the data. The data can be secret or private, but it can also be public. An example of the latter is Digimarc's Smart Images [5].
- **Error detection:** In [8], the authors presented an error detection scheme in video coding using a fragile watermark. The authors show that this proposed scheme performs significantly better than a syntax-based error detection scheme. Similar approaches are also presented in [23, 24].
- **Compression:** The authors in [9] use watermarking techniques to improve the compression rate of color images. In this scheme, the color information of the image is embedded as a watermark into the luminance data to reduce the data storage requirements.

2.4. Digital watermarking: Requirements

The exact requirements of a watermarking system strongly depend on the particular applications in which it will be deployed. However, the general requirements for a robust, invisible watermark can be summarized as follows [7]:

- **Imperceptibility:** In most applications, the watermarking algorithm must embed the watermark such that this does not affect the quality of the underlying host data. The watermark is truly imperceptible if humans cannot distinguish the host data from the watermarked data. However, since users of watermarked data normally do not have access to the host data, they cannot perform this comparison. Therefore, it is sufficient that the modifications in the watermarked data go unnoticed as long as the data are not compared with the original data.
- **Capacity:** The term watermark capacity (payload) refers to the amount of information that can be stored in a watermark. In other words, the capacity refers to the amount of information carried by the message *m* (see Figure 2.1). The payload requirements for a watermarking system depend on the specific application. For copy protection purposes, a payload of one bit is usually sufficient. For other applications, up to 70 bits [6] of information may have to be embedded in the host data, the image, video-frame or audio fragment. Another important concept regarding watermark payload for digital audio and video data is *watermark granularity*. Watermark

granularity represents how much data is needed to embed one unit of message *m*. Using the example above, the message *m* contains 70 bits. The watermarking system could be designed such that *m* is embedded in a single frame of video. Alternatively, it can also be spread over 100 frames of video (or similarly for audio, *m* could be embedded in a one-second fragment or spread over five seconds of audio data). Spreading the message in this way may not be desirable because when someone takes just 80 frames from the watermarked video, the message *m* can no longer be completely retrieved. For digital videos, one second of video is considered to be the smallest copyrighted entity. Therefore, the watermark information has to be embedded in a less than one one-second fragment of the video stream (approximately 25 frames). Again using the example above, the watermark bit rate should then be more than 70 bits/s.

- **Robustness:** A robust watermark should remain in the host data, even if the quality of the host data is degraded (i.e., attacked) either intentionally or unintentionally. A more detailed discussion of attacks on watermarking systems is provided in Section 2.6.

These requirements are not independent of each other and in the implementation of watermarking system trade-offs have to be made between the requirements. For example, increasing the payload of the watermark usually means that the robustness or the imperceptibility of the watermark will have to be reduced.

2.5. Digital watermarking: An example

To illustrate how a watermarking system works, we present in this section an image watermarking technique as an example. More examples of state-of-the-art watermarking techniques are presented in [7] and [10].

The watermarking scheme presented in this section is one of the oldest and most straightforward ways to add a watermark to an image. In this method, the watermark is embedded spatially as a pseudo-random noise pattern to the luminance values of the host image pixels (see Figure 2.2). Many methods are based on this principle [7, 10]. In general, the pseudo-random noise pattern consists of the integers {-1,0,1}; however, floating-point numbers can also be used. The pattern is generated based on a key using, for instance, seeds, linear shift registers or randomly shuffled binary images. The only constraints are that the energy in the pattern is more or less uniformly distributed and that the pattern is not correlated with the host image content. To create the watermarked image $I_W(x,y)$ the pseudo-random pattern $W(x,y)$ is multiplied by a small gain factor k and added to the host image $I(x,y)$. In other words, we have

$$I_W(x,y)= I(x,y)+ k \cdot W(x,y) \tag{2.1}$$

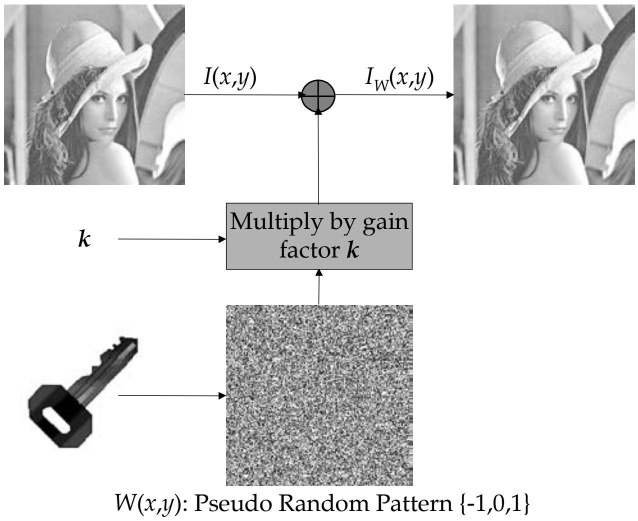

I(x,y)
I_W(x,y)
k
Multiply by gain factor k
W(x,y): Pseudo Random Pattern {-1,0,1}

Figure 2.2. Watermark embedding procedure.

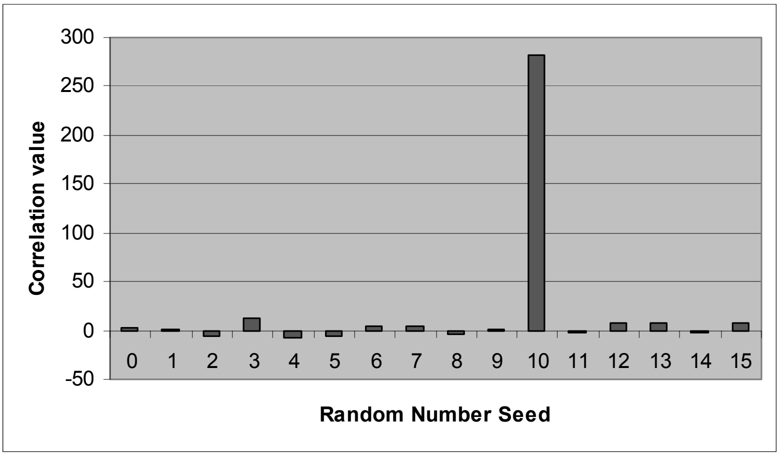


Figure 2.3 Correlation values for a pseudo-random pattern generated with seed=10 correlated with pseudo-random patterns generated with other seeds

To detect a watermark in a possibly watermarked image $I'_W(x,y)$ we calculate the correlation between the image $I'_W(x,y)$ and the pseudorandom noise pattern $W(x,y)$. Pseudo-random patterns generated using different keys have very low correlation with each other. Therefore, during the detection process the correlation value will be very high for a pseudo random pattern generated with the correct key, and would be very low otherwise. As an example, we have watermarked the Lena image by adding a pseudo-random pattern generated using seed = 10 to the image. Figure 2.3 shows the correlation values between some pseudo-random patterns generated using seeds varying between 0 and 15 and the watermarked image. It can be seen that the correlation when the correct seed (10) is used is very high, while the correlation when the wrong seeds are used is very low.

During the detection process, it is common to set a threshold *T* to decide whether the watermark is detected or not. If the correlation exceeds a certain threshold *T* the watermark detector determines that image $I'_W(x,y)$ contains watermark $W(x,y)$. Otherwise, the watermark detector determines that the image is not watermarked.

The watermarking method described above carries a payload of only 1 bit. This method can be extended to increase the capacity of the watermark. The most straightforward way to do this is by dividing the image into multiple blocks. A pseudo-random pattern is then added to each image block, each representing one bit of the watermark data. This extension is presented for example in [11].

The techniques described above can also be applied to *transformed* image data. Each transform domain has it own advantages and disadvantages. For example, the author in [12] used the phase of the Discrete Fourier Transform (DFT) to embed a watermark, because the phase is more important than the amplitude of the DFT values for the intelligibility of an image. Putting a watermark in the most important components of an image improves the robustness of the watermark, since tampering with these important image components to remove the watermark will severely degrade the quality of the image. The second reason to use the phase of the DFT values is that it is well known from communication theory that phase modulation often possesses superior noise immunity in comparison with amplitude modulation.

The Discrete Cosine Transform (DCT) domain can also be used to embed watermarks [7,10]. Using the DCT, an image can easily be split up into pseudo frequency bands, so that the watermark can conveniently be embedded in the most important middle band frequencies. Furthermore, the sensitivity of the human visual system (HVS) to the DCT basis images has been extensively studied, which resulted in the recommended JPEG quantization table. These results can be used for predicting and minimizing the visual impact of the distortion caused by the watermark. Finally, the block-based DCT is widely used for image and video compression. By embedding a watermark in the same domain as the compression scheme used to process the image (in this case, in the DCT domain), we can anticipate lossy compression because we are able to anticipate which DCT coefficients will be discarded by the compression scheme. Furthermore, we can exploit the DCT decomposition to make real-time watermark applications.

Another interesting domain that can be used to embed watermarks in images is the Discrete Wavelet Transform (DWT). This transform is very attractive, because it can be used as a computationally efficient version of the frequency models for the Human Visual System (HVS) [13] to improve watermark imperceptibility. For instance, it appears that the human eye is less sensitive to noise in high resolution DWT bands and in the DWT bands having an orientation of 45° (i.e. *HH* bands). Furthermore, DWT image and video coding, such as embedded zero-tree wavelet (EZW) coding, is included in the state-of-the-art image and video compression standards, such as JPEG2000 [14]. By embedding a watermark in the same domain (DWT domain), we can anticipate lossy EZW compression because we can anticipate which DWT bands are going to be affected by the compression scheme. Furthermore, we can exploit the DWT decomposition to make real-time watermark applications.

2.6. Attacks on digital watermarking systems

Watermarking systems are susceptible to many kinds of attack. These attacks could be performed intentionally or unintentionally. Watermarking systems utilized in copy protection or data authentication schemes are especially susceptible to intentional attacks. Unintentional attacks usually come from common signal processing operations done by legitimate users of the watermarked materials, for example a user might want to compress a bitmap image using JPEG compression simply to conserve disk space. Intentional attacks are usually done by more competent people with more knowledge of watermarking systems and more resources to make the attack. The discussion in this section is limited to the watermarking system applied to digital images and video data.

The general classification of attacks on watermarking systems is shown in Figure 2.4. The distinction between Type I and II attacks is in the target which each attack class focuses and is shown in Figure 2.5. Since “Type I” attacks operate on the watermarked data, these attacks usually involve some signal processing operations. As illustrated in Figure 2.4, this type of attack is further divided into two categories. The first category attacks the embedded watermark and aims to make a corresponding watermark detector unable to detect the embedded watermark. The second category tries to modify or otherwise tamper with the data in which the watermark is embedded, without destroying the watermark itself. We will call the first category “Type I-A” attacks and the second category “Type I-B” attacks.

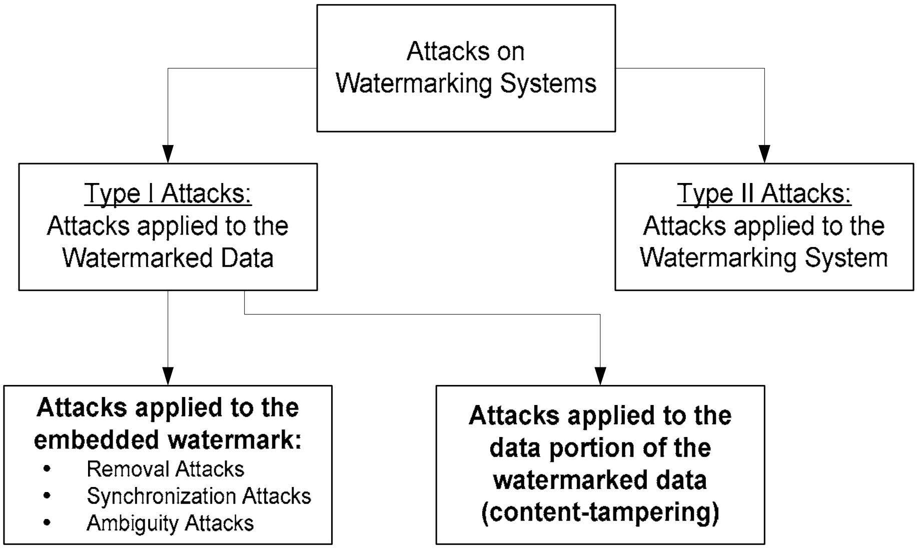

Attacks on Watermarking Systems
Type I Attacks: Attacks applied to the Watermarked Data
Type II Attacks: Attacks applied to the Watermarking System
Attacks applied to the embedded watermark:
Removal Attacks
Synchronization Attacks
Ambiguity Attacks
Attacks applied to the data portion of the watermarked data (content-tampering)

Figure 2.4. General Classification of Watermark Attacks

"Type II" attacks may be performed without regard of the watermarked data or the original (unwatermarked) data. Therefore, a signal processing operation might not be needed. Instead, intimate knowledge of programming languages, operating systems or hardware is usually needed. This attack is usually referred to as "hacking" when it deals with software or "hardware tampering" if it deals with hardware.

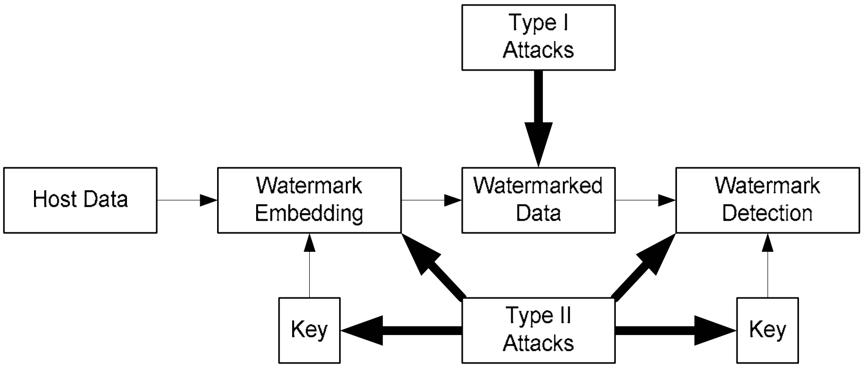


Figure 2.5. Distinction between Type I and Type II attacks

2.6.1. "Type I-A" attacks

The Type I-A attack category is further divided into 3 sub-categories, with their own distinctive characteristics. The authors in [17] divided this category into four sub-categories, differentiating between simple and removal attacks. However, this author thinks that these two attacks have a nearly identical *aim*, but a different *strategy*. Therefore, they are classified as different members of the "Removal Attack" sub-category.

2.6.1.1. Removal attacks

The main distinctive characteristic of the Removal Attacks sub-category is that *they aim to remove or severely reduce the energy of the watermark embedded in the host data* so that a detector can no longer positively detect it. It is further divided into "Simple" and "Analysis" attacks to show the different strategies adopted to reach this common goal.

Simple attacks do not involve analysis of the watermarked data in order to remove the watermark. A simple attack operates directly on the watermarked data and tries to reduce the energy of the watermark signal until it disappears from the host data or until it is no longer detectable. Because these attacks operate on the watermarked data, both the data and the watermark are purposefully degraded during the attack. These attacks rely on the fact that the watermark signal is of much lower energy than the host data signal, and therefore an attacker hopes that the watermark energy can be reduced beyond detection before the quality of the host data is severely degraded. It should be noted that the term "simple" in simple attacks stems from the fact that an attacker does not try to analyze the watermark embedded in the image/video material. It should not be inferred that simple attacks are in fact simple or trivial to execute. For example, a decoding and re-encoding process of MPEG video material might be an enormous task, demanding high disk and computational capacity. Examples of simple attacks include:

- ***Lossy Compression:*** Lossy compressions, for example JPEG and MPEG, purposefully discard some portions of the image/video data that are deemed unimportant. The amount of data removed depends on the quality factor/compression factor used. The watermark is usually embedded in this unimportant portion of the data in order to give the smallest impact on the quality of the watermarked material. Therefore, it could be removed or severely impaired during the process.
- ***Digital-to-Analog and Analog-to-Digital conversion:*** Certain watermarking techniques, for example LSB manipulation of the digital data [21], will not be able to survive this attack. When the data is converted into an analog signal, for example when viewing an MPEG movie, the watermark is lost. An attacker could record the movie into an analog video tape, and he will get an unwatermarked video. If needed, he could always re-encode the unwatermarked video back into a digital format.
- ***Transcoding:*** Watermarks applied to digital video data, for example an MPEG stream, might also be removed when the video is re-encoded with a

lower bit-rate [21]. The process is similar to re-compressing a JPEG compressed image using JPEG compression with a lower quality factor.

- ***General filtering:*** General filtering operations could be used to attack watermarked data. Low pass filtering, for example, can be used to remove a pseudo-random noise watermark, since the watermark is essentially a high frequency noise.

In the other category of removal attacks, namely the "analysis" removal attacks, an attacker tries to analyze (using some statistical analysis) the watermarked data in order to find (or estimate) the watermark or the host data. This information is then used to remove the watermark. These attacks are usually quite elaborate and are usually done intentionally. Unlike simple attacks, watermark removal using analysis attacks usually does not severely affect the quality of the data. Attacks belonging to this category include:

- ***Non-linear filtering:*** Using a non-linear filter, an attacker could estimate the watermark embedded in an image. This estimate is then used to remove the watermark. An example of this attack is the WRS attack [20].
- ***Statistical averaging:*** In this scenario, an attacker possesses N different images (or frames of a video sequence) all embedded with the same watermark. By statistically averaging these images/frames, the attacker would be able to estimate the watermark applied to them. This information is then used to remove the watermark embedded in each individual image or video frame. This attack will be particularly successful if the watermark is not significantly dependent on each image.
- ***Collusion attack:*** A collusion attack could be seen as the complement of the statistical averaging attack mentioned above. In this attack, each member of a group of attackers possesses one copy of an image I. However, each individual copy is watermarked with different watermarks (or fingerprints). By averaging these copies, these attackers would be able to estimate and produce a copy of the original, non-watermarked, host data.
- ***Embedder/Detector Observation:*** This approach is different from the hacking attack belonging to the "Type II" attacks. In this scenario, the attacker possesses the watermark detector device. He then proceeds to modify the properties of the watermarked data (changing pixel luminance, etc.) and observes how the detector/embedder responds. His objective is to find the smallest possible modification to the watermarked data such that the watermark detector will fail to detect the presence of the watermark. This modification is then applied to all materials watermarked with a similar scheme. If the attacker possessed the watermark embedder device, for example a DVD player/recorder capable of changing the watermark from "copy-once" to "no more copies", he would be able to observe the data before and after the watermark embedding process. He could then compute the difference image, which is equal to the watermark embedded. All he

need do is pre-distort the unwatermarked material by subtracting this difference. If this pre-distorted material is then run through the watermark embedder, the result should be approximately identical to the original unwatermarked material.

2.6.1.2. Synchronization attacks

The main characteristic of this class of attacks is that an attacker does not attempt to remove the watermark from the watermarked data, but to remove the synchronization of the watermark so that it cannot be detected properly by a watermark detector. The watermark itself (or a major part thereof) is still physically present in the data. As in simple removal attacks, the attacker does not have to analyze the watermarked data to identify the watermark. However, unlike in a simple removal attack where the energy of the watermarked signal is reduced, in synchronization attacks the watermark loses only its synchronization with the detector. A synchronization attack is done by performing geometric operations to the watermarked data. When these operations are performed on the watermarked data spatially, we have spatial synchronization attacks. We encounter spatial synchronization attacks in image and video watermarking. Examples of spatial synchronization attacks include:

- ***Geometric transformation.*** Performing geometrical transformations, for example rotation, translation, scaling or slight bending can disturb the synchronization of the watermark and the watermark detector.
- ***Pixel deletion/substitution:*** An example of this attack is the removal of a row/column of pixels from an image. If the image size is to be preserved, another row/column could be duplicated and inserted. This operation usually does not give perceptible degradation of the watermarked data.
- ***Mosaic attack:*** A mosaic attack is performed by dividing an image into smaller portions. When used in web pages, a browser will reconstruct the image with no apparent quality loss or time delay (sometimes loading a complete image is slower than reconstructing the pieces). This attack is primarily done to prevent web-crawlers designed to check watermarks from completing their job because the smaller pieces contain no recognizable watermark. It is possible, of course, to embed individual copies of the watermark into smaller blocks of the original picture. However, many watermarking methods are generally unable to embed watermarks into small pieces of image (smaller than 100 × 100 pixels) [22]. Therefore, by dividing the original image into blocks smaller than 100 × 100 pixels an attacker can prevent the web-crawlers from detecting the watermark.

On the other hand, when the geometric operations are performed on the temporal axis we have a temporal synchronization attack. These types of synchronization attacks are encountered in audio and video watermarking. Examples of temporal synchronization attacks include:

- ***Temporal scaling.*** Increasing or decreasing the playback speed of audio or video data is equivalent to performing a scaling operation on the temporal axis. Like spatial scaling, this operation can also disturb the temporal synchronization of the watermark and the detector.
- ***Sample deletion or duplication.*** This attack is the temporal equivalent of spatially removing rows or columns from an image. In the case of audio watermarking, this attack is performed by removing or duplicating audio samples. In the case of video watermarking, this attack is performed by removing or duplicating video frames.

2.6.1.3. Ambiguity attacks

One form of ambiguity attack is a scenario in which an attacker tries to embed another watermark into watermarked data, thus making it difficult (or impossible) to determine the first embedded watermark (and thus the real legitimate watermark). One way to do this is simply to insert another watermark into already watermarked data. This could be countered by embedding a time-stamp, or by keeping the original watermarked data as a reference in the event of a dispute. A more sophisticated variant of this attack is to claim part of the original watermarked data as counterfeit host data and insert a second watermark derived from the legitimately watermarked data. For example, assume that I is the original, unwatermarked image, W is the legitimate watermark, $E(I, W)$ is a function to embed the watermark W into I and I_W is the watermarked version of I. It has been demonstrated [15] that for some watermarking algorithms, an attacker could compute a pattern W', a counterfeit original I' and a function $E'(I', W')$ such that $E'(I', W') = I_W$ and claim that I' is his original, unwatermarked data and W' is his watermark, thus creating an ownership dispute over I_W. This attack will work only on so-called *invertible* watermarking algorithms. A more sophisticated attack that does not impose such limitations is discussed in [16].

Another form of ambiguity attack is the copy attack, described in [33]. In this attack, an attacker copies a valid watermark from watermarked data I_w (containing a valid watermark W) and embeds it into another host data, X, producing X_w. The attacker can do this without any knowledge of the original embedding algorithm or of the key used to embed the original watermark. The watermark detector will declare that both I_w and X_w contain the watermark W. This attack can lead to an ambiguous situation. For example, the attacker may claim that the original owner of the watermark has stolen his data, X, and use the copied watermark as proof.

2.6.2. “Type I-B” attacks

This category of attacks is aimed to modify or tamper with the data in which the watermark is embedded, without destroying the watermark itself. Such an attack is performed, for example, to discredit an institution by tampering with material bearing its watermark (for instance, by blurring part of the image or changing the color of some area of the image). This might prove very disturbing if, for example, the image is going to be used as evidence before a court of law. This attack is only effective against robust watermarks and not against fragile watermarks or watermarks that are specifically designed to detect tampering.

2.6.3. “Type II” attacks

In “Type II” attacks, an attacker attempts to defeat the watermarking system not by attacking the watermarked data. Instead, the attacker performs his attacks on the software components or the hardware components of the watermarking system. Examples of “Type II” attacks are:

- ***Software tampering (hacking).*** If the watermark embedder and detector are implemented in software and are widely available, they are especially susceptible to these attacks. Attacks of this kind are usually performed as follows: an attacker (hacker) obtains the watermark embedder/detector software and proceeds to either decompile the software or use debugging software to dig deep into the code. The attacker might then be able to find the specific portion of the code that generates or detects the watermark. Once this is accomplished, this information is used to accomplish the attacker’s goal. For example, if the aforementioned attacker found the portion of the code used to generate the watermark, he could use this information to generate counterfeit watermarks. Alternatively, if he found the portion of the code used to check for the presence of a watermark in the detector code, he could modify the code to bypass the security scheme routine implemented in the detector.
- ***Hardware tampering.*** These attacks are performed on the hardware components of the watermarking system, for example a DVD player. Here an attacker will actually disassemble the hardware, study the inner workings of the hardware and modify it to suit his needs. For example, an attacker can alter the circuitry of a DVD player to disable its watermark detection capability.

2.6.4. Counter measures against watermark attacks

So far we have discussed some possible forms of attacks against the watermarking system. There are some counter measures that could be taken to deal with these attacks. Attempts to defeat an attacker should always consider that an attacker also has a certain set of criteria when performing the attack. The main criteria are the cost-effectiveness of the attack and the distortion caused by the attack. Therefore the main goal to defeat watermark attacks is to make the attack as difficult as possible or to make the attack degrade the quality of the watermarked data as much as possible.

To deal with simple removal attacks, basically we have to come up with a watermarking algorithm that could put a watermark with a higher power into the image/video. By properly exploiting HVS properties, a higher watermark signal power could be embedded without affecting the visual quality of the image/video. A stronger watermark signal means that the watermarked data must be degraded more in order to render the watermark undetectable. This might not be possible since the resulting image quality might not be acceptable. This measure might not work against analysis removal attacks though, especially if the attacker can observe the non-watermarked data as well as the watermarked data. To combat a collusion attack, a proposed method is to embed a watermark that has dynamic and static components [17]. The dynamic component varies for each user, and might average to zero when attacked using a collusion attack. The static component will not average to zero, and therefore will remain present in the attacked image. Furthermore, in [25], a mathematical framework of collusion attacks in video watermarking and another robust watermarking scheme are described.

By their nature, synchronization attacks do not actually remove the watermark from the watermarked data. One approach to counter this attack is to design the watermarking system such that it is invariant to synchronization attacks. An example of such a scheme is given in [26]. Another, more popular approach is to make the watermark re-synchronizable after an attack. This can be achieved by embedding a synchronization pattern into the watermark. Examples of this approach are given in [27]. Invariant features of the host data can also be used for this purpose [28]. Alternatively, the watermark can also be re-synchronized by inverting the geometric distortion using image registration techniques. Examples of this approach are presented in [29, 30]. The examples discussed above are developed to combat spatial synchronization attacks. Similar approaches can also be used to combat temporal synchronization attacks. For example, the authors in [31] present a watermarking scheme that can recover temporal synchronization by comparing the attacked data to the original host data. An alternative approach is presented in [32], in which the temporal synchronization can be recovered blindly, i.e., without relying on the presence of the original host data.

Using secure time-stamping is one possible solution to counter ambiguity attacks, and this may be enough against simple re-watermarking attacks. A more sophisticated counter measure for more advanced attacks is presented in [17, 36]. The best counter measure against copy attacks is to make the watermark highly dependent on the unique characteristic of the host data. For example, the watermark could carry the hash value of the host data. Comparing the hash values computed from the received watermarked data and the hash value carried by the watermark is a way of verifying the validity of the watermark.

Content-tampering could be defeated with watermarking algorithms designed to detect not only whether the image/video had been tampered with (e.g., by employing a fragile watermark), but also to show where the attacker had tampered with the material. Examples of watermarking schemes designed to protect data against content tampering are presented in [18, 19].

“Type II” attacks could be defeated by carefully designing the software or hardware components of the watermarking systems. Although there are no real guarantees that carefully designed software or hardware will be able to defeat a highly skilled and determined attacker, at least a designer should try to make the attack harder, and therefore more costly.

2.7. Theoretical approaches to digital watermarking

The capacity of the watermarking system discussed in the example presented in Section 2.5 is very low, namely 1 bit of information per picture. This scheme can still be extended to carry more information bits and there are also more advanced watermarking techniques that have larger capacities. However, in general, the capacities of current watermarking systems are still far below the information-theoretical capacity limit [36].

To deal with this problem, new watermarking approaches based on information theory have been proposed. In these approaches, the host data is considered as side information, while in most other blind watermarking approaches, it is considered as an interfering noise. For example, the scheme proposed by Costa gives optimal capacity of a watermarking scheme facing Additive White Gaussian Noise (AWGN) attacks [34]. However, this scheme involves a random and very large code book. Therefore, this approach cannot be implemented as a practical watermarking system. The authors in [34] propose a practical implementation of Costa’s idea, called Scalar Costa Scheme (SCS), where the random code book is replaced by a structured code book. The capacity of this watermarking scheme is lower than, but closely approaches, the theoretical capacity limit. Another approach is based on Quantization Index Modulation (QIM) methods which are given in [35].

Research on watermark capacity from a game theory point of view has also been performed [36, 37, 38]. From a game theory point of view, the watermarking problem is seen as a game between the watermark embedder and the watermark attacker. The goal of the embedder is to maximize the amount of information embedded into the host data, while the goal of the attacker is to minimize this amount. The optimal watermark capacity for such a scenario is derived in [36]. The authors in [37] also propose another watermarking scheme based on the game-theoretical approach.

One limitation of the watermarking approaches discussed in this section is that they are only designed to be robust against a limited class of attacks. The class of attacks considered includes only attacks where the distortion incurred to the watermarked data can be measured using Mean Squared Error (MSE) metric, for example AWGN attacks. Therefore, the schemes are still very vulnerable to synchronization attacks.

2.8. Concluding remarks

In this chapter, we have discussed the basic principles of digital watermarking techniques, their application and the requirements that have to be fulfilled by a digital watermarking scheme. We have also discussed the information-theoretical approaches to digital watermarking. Finally, we discussed the attacks that can be encountered by digital watermarking systems. These attacks can remove the watermark or render the watermark detector unable to detect the watermark. The most challenging attack is the synchronization attack, especially the spatial geometric distortion of images and video frames. To completely solve this problem, more research still has to be performed, not only to increase watermark robustness against desynchronization, but also to be able to quantify the effect of this attack on the perceptual quality of the attacked data.

2.9. References

1. J. Fridrich, M. Goljan and R. Du, *Invertible authentication*, in the Proceedings of SPIE: Security and Watermarking of Multimedia Contents III, San Jose, CA, 2001
2. I.J. Cox and M.L. Miller, *A review of watermarking and the importance of perceptual modeling*, in the Proceedings of SPIE: Storage and Retrieval for Image and Video Databases V, San Jose, CA, 1997
3. G.C. Langelaar, *Conditional access to television service*, Wireless Communication, the interactive multimedia CD-ROM, 3rd edition 1999, Baltzer Science Publishers, Amsterdam
4. T. Kalker, G. Depovere, J. Haitsma and M. Maes, *A video watermarking system for broadcast monitoring*, in the Proceedings of SPIE: Security and Watermarking of Multimedia Contents, San Jose, CA, 1999
5. A.M. Alattar, *Smart images using Digimarc's watermarking technology*, in the Proceedings of SPIE, Security and Watermarking of Multimedia Contents II, San Jose, CA, 2000
6. M. Kutter and F. A. P. Petitcolas, *A fair benchmark for image watermarking systems*, in the Proceedings of SPIE: Security and Watermarking of Multimedia Contents, San Jose, CA, 1999
7. G.C. Langelaar, I. Setyawan and R.L. Lagendijk, *Watermarking image and video data: A state-of-the-art overview*, IEEE Signal Proc. Magazine, Vol. 17, No. 5, pp. 20-46, September 2000
8. M. Chen, Y. He and R.L. Lagendijk, *Error Detection by Fragile Watermarking*, in the Proceedings of the 22nd Picture Coding Symposium PCS 2001, Seoul, Korea, April 2001
9. P. Campisi, D. Kundur, D. Hatzinakos and A. Neri, *Hiding-based Compression for Improved Color Image Coding*, in the Proc. of SPIE: Security and Watermarking of Multimedia Contents IV, Vol. 4675, pp. 230 – 239, San Jose, CA, 2002
10. I.J. Cox, M.L. Miller and J.A. Bloom, *Digital Watermarking*, Academic Press, London, 2002
11. A. Hanjalic, G.C. Langelaar, P.M.B. van Roosmalen, J. Biemond and R.L. Lagendijk, *Image and Video Databases: Restoration, Watermarking and Retrieval*, Volume 8 of the series Advances in Image Communications, Elsevier Science, 2000
12. J.J.K. Ó Ruanaidh, W.J. Dowling and F.M. Boland, *Phase watermarking of digital images for copyright protection*, in the Proc. of IEEE, ICIP '96, Vol. III, pp. 239-242, Lausanne, 1996
13. M. Barni, F. Bartolini, V. Cappellini and A. Piva, *Mask building for perceptually hiding frequency embedded watermarks*, in Proc. of SPIE: Security and Watermarking of Multimedia Contents, San Jose, CA, 1999

14. X.-G. Xia, C.G. Boncelet and G.R. Arce, *A multiresolution watermark for digital images*, in the Proc. of IEEE ICIP ‘97, Santa Barbara, CA, 1997
15. S. Craver, N. Memon, B.L. Yeo and M. M. Yeung, *Can Invisible Watermarks Resolve Rightful Ownerships?*, IBM Technical Report RC 20509, 1996.
16. S. Craver, N. Memon, B.L. Yeo and M. M. Yeung, *On the Invertibility of Invisible Watermarking Techniques*, in the Proc. IEEE ICIP ’97, Vol. I, Santa Barbara, CA, pp. 540-543, 1997.
17. F. Hartung, J.K. Su and B. Girod, *Spread Spectrum Watermarking: Malicious Attacks and Counter-attacks*, Proceedings of SPIE: Security and Watermarking of Multimedia Contents, San Jose, CA, 1999.
18. D. Kundur and D. Hatzinakos, *Digital Watermarking for Telltale Tamper Proofing and Authentication*, Proceedings of the IEEE, Vol. 87, No. 7, July 1999.
19. M.U. Celik, G. Sharma, A. Murat Tekalp, E. Saber, *Localized Lossless Authentication Watermark (LAW)*, in Proceedings of SPIE, Security and Watermarking of Multimedia Contents V, Vol. 5020, pp. 689 – 698, Santa Clara, CA, 2003
20. G.C. Langelaar, R.L. Lagendijk and J. Biemond, *Watermark Removal based on Non-linear Filtering*, ASCI ’98 Conference, Lommel, Belgium, 1998.
21. G.C. Langelaar, R.L. Lagendijk and J. Biemond, *Real-Time Labeling of MPEG-2 Compressed Video*, Journal of Visual Communication and Image Representation, Vol. 9, No. 4, pp. 256-270, 1998.
22. F.A.P. Petitcolas, R. Anderson and M.G. Kuhn, *Attacks on Copyright Marking Systems*, in Lecture Notes in Computer Science, vol. 1525, D. Aucsmith, Ed., Second Workshop on Information Hiding, Portland, OR, 1998, pp. 218-238.
23. Y. Hwang, B. Jeon, *Error detection in a compressed video using fragile watermarking*, in Proceedings IEEE, ICME 2002, Vol. I, pp. 129 – 132, August 2002
24. P. Campisi, G. Giunta, A. Neri, *Object-based Quality of Service Assessment using Semi-fragile Tracing Watermarking in MPEG4 Video Cellular Devices*, in Proceedings of IEEE, ICIP 2002, Vol. II, pp. 881 – 884, Rochester, NY, 2002
25. K. Su, D. Kundur, D. Hatzinakos, *A novel approach to collusion-resistant video watermarking*, in Proceedings of SPIE, Security and Watermarking of Multimedia Contents IV, Vol. 4675, pp. 491 – 502, San Jose, CA, 2002
26. J.J.K. Ó Ruanaidh, T. Pun, *Rotation, scale and translation invariant digital image watermarking*, in Proceedings of IEEE, ICIP 1997, Vol. I, pp. 536 – 539, Santa Barbara, CA, 1997
27. F. Deguillaume, S. Voloshynovskiy, T. Pun, *A method for the estimation and recovering from general affine transforms in digital watermarking applications*, in Proceedings of SPIE, Security and Watermarking of Multimedia Contents IV, Vol. 4675, pp. 313 – 322, San Jose, CA, 2002

28. P. Bas, J.-M. Chassery and B. Macq, *Geometrically invariant watermarking using feature points*, IEEE Trans. on Image Proc., Vol. 11, No. 9, pp. 1014 – 1028, September 2002.
29. P. Loo, and N. Kingsbury, *Motion estimation based registration of geometrically distorted images for watermark recovery*, Proceedings of SPIE, Security and Watermarking of Multimedia Contents III*,* Vol. 4314, pp. 606 – 617, San Jose, CA, 2001
30. D. Delannay, J.-F. Delaigle, B. Macq, *Compensation of Geometrical Deformations for Watermark Extraction in the Digital Cinema Application*, in Proceedings of SPIE, Security and Watermarking of Multimedia Contents III, Vol. 4314, pp. 149-157, San Jose, CA, 2001
31. D. Delannay, C. de Roover, B. Macq, *Temporal alignment of video sequences for watermarking systems*, in Proceedings of SPIE, Security and Watermarking of Multimedia Contents V, Vol. 5020, pp. 481 – 492, Santa Clara, CA, 2003
32. E.T. Lin, E.J. Delp, *Temporal synchronization in video watermarking: further studies*, in Proceedings of SPIE, Security and Watermarking of Multimedia Contents V, Vol. 5020, pp. 493 – 504, Santa Clara, CA, 2003
33. S. Craver, *The return of ambiguity attacks*, in Proceedings of SPIE, Security and Watermarking of Multimedia Contents IV, Vol. 4675, pp. 252 – 259, San Jose, CA, 2002
34. J.J. Eggers, J.K. Su, *Performance of a practical blind watermarking scheme*, Proceedings of SPIE, Security and Watermarking of Multimedia Contents III*,* Vol. 4314, pp. 594 – 605, San Jose, CA, 2001
35. B.Chen, G.W.Wornell, *Preprocessed and postprocessed quantization index modulation methods for digital watermarking*, Proceedings of SPIE, Security and Watermarking of Multimedia Contents II*,* Vol. 3971, pp. 48 – 59, San Jose, CA, 2000
36. P. Moulin and J. A. O'Sullivan, *Information-Theoretic Analysis of Information Hiding*, in IEEE Trans. on Inf. Theory, Vol. 49, No. 3, pp. 563 – 593, March 2003.
37. J.J. Eggers, B. Girod, *Informed Watermarking*, Kluwer Academic Publishers, Dordrecht, 2002
38. S. Voloshynovskiy, O. Koval, F. Deguillaume, T. Pun, *Data hiding capacity analysis for real images based on stochastic non-stationary geometrical models*, in Proceedings of SPIE, Security and Watermarking of Multimedia Contents V, Vol. 5020, pp. 580 – 593, Santa Clara, CA, 2003

Chapter 3
WATERMARKING LOW BIT-RATE VIDEO

3.1. Introduction

Digital video data distribution through the internet is becoming more common [1]. The rapid growth of the internet and the increasing number of internet users make it a very strong marketing medium with which to reach potential customers for various products. When Hollywood studios release new movies, it is now common for them to set up an official website for the movies in which they put multimedia materials, such as the movie trailers, interviews with the cast, etc. Most recording artists nowadays have their own official websites, where they can also put their video clips to promote their new albums. The same goes for big music publishing companies, because such promotion can also boost the sales of the albums of the artists under their label. The interactive entertainment industry, i.e., video and computer games industry, also sees the internet as a medium not only to distribute demos or preview versions of their games for potential customers to download, but also as a medium to distribute video materials of their games, such as in-game video sequences, the opening cinematics of their games or dedicated "game trailers" in which they show off the exciting parts of their game in a similar manner as that used in movie trailers. All these marketing efforts, especially for the last case, may make or break the sales of the products.

These multimedia materials share an important feature, namely they must be compressed at low bit rates to facilitate distribution through the internet. Furthermore, these materials need to be protected in order to prevent copyright infringement issues. Digital watermarking is one of the possible solutions for this copyright protection problem [2,5,7]. However, most of the existing video watermarking algorithms are more geared towards high bit rate environments suitable for DVD or television broadcasting [3,6]. Low bit rate (below 1 kbps) video watermarking utilizing MPEG-4 facial animation parameters has been investigated [4], and is suitable for video telephony application. However, low bit-rate watermarking for other applications, such as

The work presented in this chapter is an expanded version of *Low bit-rate video watermarking using temporally extended Differential Energy Watermarking (DEW) algorithm*, by I. Setyawan and R.L. Lagendijk, first published in Proc. SPIE, Security and Watermarking of Multimedia Contents III, Vol. 4314, San Jose, CA, 2001 the one mentioned in the previous paragraph, has received little attention in the literature.

Low bit-rate environments present new challenges to the watermarking operation which are not found in watermarking operations at high bit-rate environments. Video encoded at low bit-rates inherently possesses low redundancy and small visual degradation tolerance. This brings forward three important issues. The first issue is the watermark capacity, i.e. the number of watermark bits we can embed into the data. The second issue is the visual impact. The low visual degradation tolerance of the original video sequence/stream means that we must take special care before embedding the watermark, which essentially adds more distortion into the data. On the other hand, the coding artefacts are also more visible in streams encoded at low bitrates. Therefore, it is possible for the watermarking artefact to be dominated by the coding artefact. The third issue is the robustness of the watermark. These three issues are closely interrelated and adjusting one of these performance aspects will affect the performance of the others.

In our previous work, we developed a video watermarking scheme suitable for MPEG-1/-2 video streams encoded at high bit-rates (1.4 to 8 Mbps), called the Differential Energy Watermarking (DEW) algorithm [7,9]. This method has been shown to have relatively low complexity, high capacity and low visual impact. We consider this technique to have the potential to be extended for use in low bit-rate environments, and in this paper we present the extension scheme and an evaluation of its performance in order to investigate the behavior of this technique in low bit rate environments.

This chapter is organized as follows. In Section 3.2, the DEW algorithm is briefly described and the extension scheme is explained in detail. In Section 3.3, the experiment setup and results are presented. In Section 3.4 we present the conclusions of our experiments. Finally, in Section 3.5 we present brief discussion on recent developments in low bit-rate video watermarking techniques.

3.2. The Extended DEW algorithm

3.2.1. The DEW algorithm

The DEW algorithm embeds watermark bits into an MPEG stream (or any other block DCT based video compression system) by enforcing energy difference between certain groups of 8 × 8 DCT blocks of the I-frames to represent either a ‘1’ or a ‘0’ watermark bit. The energy difference is enforced by selectively removing high frequency components from the DCT blocks. The 8 × 8 DCT blocks of an I-frame are first randomly shuffled using a secret seed. This process serves two purposes. In the first place, the seed serves as a secret key without which one cannot extract the watermark properly. In the second place, the process is performed to avoid having a group of blocks in which there is an unbalanced energy content. If this happens, then the watermarking artefact may become visible. As mentioned above, the energy difference is enforced by removing high frequency DCT components. If too many high-frequency components are removed in order to enforce this difference, then the watermarking artefacts, in the form of blurred edges, will be visible. This may happen, for example, when one group of blocks has no high frequency component (i.e., contains only flat areas) while the other group of blocks contains edges. If the energy content of the second group of blocks has to be reduced to enforce the energy difference, then too many high frequency components would have to be removed and the edges will be blurred as a consequence. The fundamental terms of the DEW algorithm are illustrated in Figure 3.1.

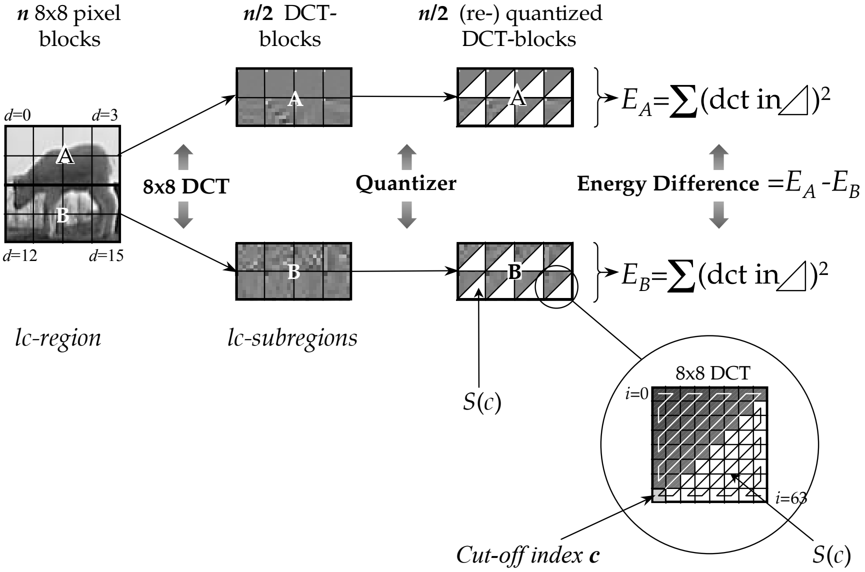

n 8x8 pixel blocks
n/2 DCT-blocks
n/2 (re-) quantized DCT-blocks
$E_A = \sum(\text{dct in} \triangle)^2$
d=0
d=3
A
B
d=12
d=15
8x8 DCT
Quantizer
Energy Difference $= E_A - E_B$
$E_B = \sum(\text{dct in} \triangle)^2$
lc-region
lc-subregions
S(c)
8x8 DCT
i=0
i=63
Cut-off index **c**
S(c)

Figure 3.1. The Differential Energy Watermarking

The DEW algorithm has several adjustable parameters. By adjusting these parameters, we can adjust the watermarker to optimise it either for capacity, robustness or visual impact [9]. The parameters are as follows:

- *Number of 8 × 8 blocks per watermark bit*: This parameter is represented by *n* in Figure 3.1. It influences the capacity and the robustness of the watermark. The more blocks are used to embed a single watermark bit, the less capacity is achieved, but the more robust the watermark would be and the less degradation would be introduced because the required energy difference is “spread” among the blocks, and the more blocks we use the less energy in the region *S(c)* has to be removed from each DCT block.

- *Enforced energy difference*: This is the minimum allowed value of E_A-E_B in Figure 3.1. This parameter influences the robustness and visual impact of the watermark. The larger the energy difference enforced, the more robust the watermark would be. The disadvantage could be worse visual quality because more DCT coefficients have to be discarded. Furthermore, due to the limitation imposed by the next parameter, it is possible that some watermark bits cannot be correctly embedded.
- *Minimal cut-off point*: This parameter is represented by c in Figure 3.1, and denotes the index of the particular DCT coefficient number in an 8 × 8 block (zigzag scanned). Any coefficient with index $i < c$ may not be removed to enforce the energy difference. Thus it can be seen as a limiter to the previous parameter because this parameter determines how many DCT coefficients may be removed to enforce the energy difference. If this parameter is set too high, then the watermark robustness would suffer and there is a possibility that some watermark bits cannot be properly embedded because the proper energy difference could not be enforced. However, the visual quality degradation introduced by the watermarking would be lower than the degradation introduced when a lower minimal cut-off is set because fewer DCT coefficients are removed.

The DEW algorithm also has several interesting properties. It is relatively uncomplicated because it embeds the watermark at the DCT coefficient level and thus only VLC decoding is needed for the watermark embedding and detection process, and no full decoding and re-encoding of the stream is needed. This scheme also has sufficient robustness because a full decoding and re-encoding is needed to completely remove the watermark from the stream. It has been shown that even transcoding a watermarked 8 Mbps MPEG stream down to 6 Mbps only introduce a 7% Bit Error Rate (BER) [7]. The capacity of the watermarking scheme is also sufficiently high, up to 0.42 kbps for a stream encoded at 8 Mbps. The visual impact of the watermarking process is also negligible.

3.2.2. Extending the DEW algorithm

The primary motivation of extending the DEW algorithm is to “spread” the watermark bits more in the temporal dimension. Spreading the watermark data in the temporal dimension offers potential improvements to the original DEW algorithm, especially for implementation in low bit-rate environments. The potential improvements are discussed below:

- ***Improved capacity:*** One of the main issues when we move to a lower bit-rate environment is the watermark capacity. The spatial resolution of the frames plays a very important role for the watermark capacity of the DEW algorithm. Videos in the application scenarios mentioned in the introduction are usually of CIF or lower spatial resolution, while the video used in the applications for which the DEW algorithm was originally designed usually has as much as four times the spatial resolution (4CIF). Thus, one of the main problems here is to find a way to compensate this capacity limitation imposed by the video resolution. One possible solution to this problem is to reduce the number of 8 × 8 DCT blocks that are used to embed each watermark bit. However, reducing the number of blocks also reduces the robustness of the embedded watermark. The other solution we investigate here is to use more frames to embed the watermark, thus spreading the watermark in the temporal dimension. To achieve this, we will use not only the I-frames of the stream to embed the watermark, but also the P-frames. The challenges here are:

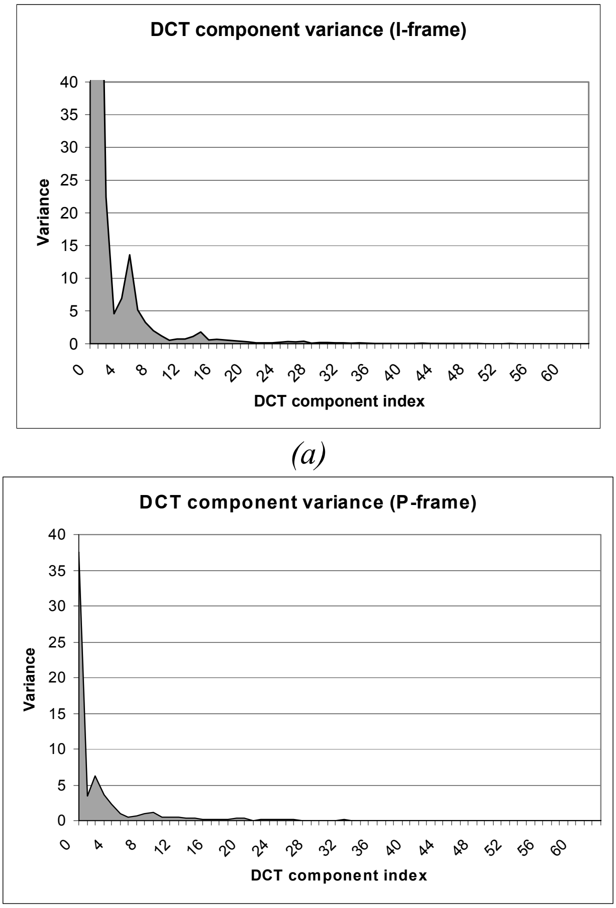
  - The much lower energy content of the P-frames compared to the I-frames means that this solution will require a more delicate approach in order to balance the capacity, robustness and visual impact requirements.
  - The drift effects [3]. This is the result of error accumulation because the watermarked frames are used to reconstruct the frames and they are also being used as a prediction reference for other frames. Over time, this error accumulation may become visible. Even worse, the error may spatially spread.

- ***Improved robustness:*** The next issue concerns the robustness of the watermark. In this respect, the extension offers two possible advantages. The first advantage is derived directly by the extra space we can use when we use the temporal dimension. By using this extra space, we can embed fewer bits in each frame (thus increasing the watermark robustness) but still achieve the same watermark payload. Another possible advantage is that if an attacker wants to remove the watermark in a video stream watermarked using the original DEW algorithm, he has only to deal with the I-frames, which are relatively few in a sequence. If the algorithm is extended so that the watermark data does not reside only in the I-frames, then the attacker would have to deal with more frames. This does not eliminate the possibility of an attacker successfully removing the watermark, but this will make the attack more cumbersome.
- ***Improved visual quality:*** The next possible improvement concerns the visual quality of the watermarked stream. This issue is related to the previous issues, because if we embed fewer watermark bits per frame then the degradation to the data is reduced. The extra space provided by the extension would compensate the decrease of watermark payload per frame.

The extension to the original DEW algorithm is achieved by modifying the watermark embedder so that it embeds the watermark not only in the I-frames, but also the P-frames. As noted above, the energy content of an I-frame and a P-frame is very much different. In an I-frame, the whole image data from that frame is intra-encoded. A P-frame, on the other hand, is predicted from a previous I-frame. Only the *prediction error* is encoded into a P-frame. This prediction error carries much less energy than an intra-coded frame. As an example, one I-frame of the Claire MPEG stream (CIF resolution, encoded at 700 kbps) has a total signal variance of 2057.7, while a P-frame predicted from this I-frame has a total signal variance of only 59.8. This value varies widely from P-frame to P-frame, depending on the amount of activity (movements) in the sequence. When there are a lot of movements in the sequence, the P-frame will contain more energy than when the amount of movements is small. For this particular sequence, the average variance of the P-frames is 40.223. The variance of the I-frames also varies from I-frame to I-frame, but the variation is less significant. The average variance of the I-frames in this particular sequence is 2034.6. In Figure 3.2, we show the variance of the DCT components of one I- and one P-frame predicted from the I-frame.

Figure 3.2(a) is clipped at variance values of 40 in order to show the variance of the higher DCT coefficients, and also to allow a somewhat easier comparison between the two figures. Other than the obvious difference in scale, the two figures show very similar behavior. Although the variance of each individual P-frame is different, depending on the level of activity found in the sequence, the behavior is similar to that observed in Figure 3.2(b). This behavior suggests that, with proper parameter adjustments, the DEW algorithm can be applied directly to the P-frames. The parameters that need to be adjusted are either the *enforced energy difference*, which should be lower due to the lower energy content of the P-frame, the *minimal cut-off point*, which should also be lower due to how the energy is distributed in the frame, or both.

On the average, the variance of the I-frames is larger by a factor of approximately 50 compared to the variance of the P-frames. Therefore, we will need approximately 50 times as many 8 × 8 DCT blocks to embed one watermark bit in one P-frame to be able to achieve the same level of performance as when we embed the watermark in an I-frame, all other parameters being equal. If we can properly embed watermarks into an I-frame using 32 DCT blocks per watermark bit, we would need approximately 1600 DCT blocks in a P-frame. Since a sequence with CIF resolution only has 1584 DCT blocks per frame, this means that there is a maximum of 1 watermark bit that can be reliably embedded in a P-frame on average. Therefore, instead of only increasing the number of blocks, we also reduce the level of energy difference that has to be enforced. In this way, we should not need 50 times as many blocks to be able to properly embed the watermark bits, which means that we should be able to embed more than one watermark bit per P-frame. The price we have to pay here is the lower robustness of the watermark.



(a)

(b)

Figure 3.2. DCT component variance of (a) an I-frame and (b) a P-frame taken from the Claire sequence, MPEG encoded at 700 kbps

It is also possible to extend the algorithm by embedding the watermark into the B-frames. Most of the frames in an MPEG sequence are B-frames. This large number of frames offers even more capacity increase than if we extend the algorithm to use the P-frames. Furthermore, from a robustness point of view, this will make the attack even more cumbersome. The B-frames are not used to predict other frames, which means we do not have to deal with drift effects. However, the B-frames contain even less energy than the P-frames. We have discussed above that even the P-frames may not contain enough energy to properly accommodate the watermark, and thus we choose not to extend the algorithm to embed the watermark in the B-frames.

The watermark bits are embedded in the P-frames in the same manner as they are embedded in the I-frames. First, the 8×8 DCT blocks of the P-frames are shuffled pseudo-randomly using a random key. The same key is used for both the I-frames and the P-frames in our experiments. However, technically there is no problem if a different key (independent of the key used to shuffle the I-frames blocks) is used. After the blocks are shuffled, the DEW algorithm is performed to embed the watermark bits. Each watermark bit is embedded into a certain number of 8 × 8 DCT blocks. From the watermarking point of view, the operations on the I-frame and the P-frame are independent, i.e. the watermark in the I-frame can be detected independently from the watermark in the P-frame (and vice versa) and the BER of the watermark embedded in the I-frame is not affected by the BER of the watermark embedded in the P-frame (and vice versa).

The BER of the watermark embedded in either the I- or P-frames can be introduced either due to:

- Insufficient energy content in the I- or P-frames which means that certain energy differences cannot be enforced. This happens during embedding, and we will call this the *e*BER.
- Distortion of the watermarked frame due to attacks, for example due to re-encoding. We will call this the *a*BER.

Both BERs are calculated as follows:

$$BER = \frac{bit\,errors}{total\,embedded\,bits} \times 100\% \qquad (3.1)$$

In Equation (3.1), the *total embedded bits* refers to the total amount of bits the watermarker software attempts to embed in the sequence. Due to the parameter settings chosen, it may not be able to properly embed some of these bits.

Furthermore, the watermark embedded in the P-frames also has the same adjustable parameters as the watermark embedded in the I-frames. These parameters are adjusted independently from the parameters of the I-frame watermark. This allows us to use either identical settings or different settings that are more appropriate for the P-frames due to their different characteristics.

3.3. Experiment setup and results

3.3.1. Experiment setup

We test the extended DEW algorithm to see its performance in terms of watermark capacity, watermark robustness and visual quality impact. We use MPEG-2 sequences encoded at 256, 384, 512 and 700 kbps with a frame rate of 25 fps. All sequences are encoded using the same *group of pictures* (GOP) structure. We use the commonly used GOP structure, i.e., IBBPBBPBBPBB.

The sequences are encoded as a *progressive* sequence and therefore the use of MPEG-1 coded sequences would also have been possible. The extended DEW algorithm itself is compatible with MPEG-1 stream. The spatial resolution of the sequences is 352 × 288 pixels (CIF). The sequences used in our experiments are: "Claire" (78 frames), "Trevor" (150 frames) and "Akiyo" (250 frames). The sequences are watermarked using the new version of our DEW watermarker software that can operate in both "default" and "extended" mode. In default mode, the software essentially operates identically to the original watermarking software. In extended mode, the watermarker embeds the watermark using the extended DEW algorithm.

Two parameters are fixed during the experiments, i.e. the *enforced energy difference* and the *minimal cut-off point*. For the I-frames (both in "default" and in "extended" modes) the *enforced energy difference* is set at 20 and the *minimal cut-off point* is set at 6. For the P-frames, the *enforced energy difference* is set at 4. This much lower value is chosen due to the lower energy content of the P-frames compared to the I-frames. We have pointed out in Section 3.2.1 that this parameter (for a fixed number of DCT blocks per watermark bit and a certain *minimal cut-off point*) influences the probability that a watermark bit can be properly embedded. If this parameter is set at the same value as the one used for the I-frames, there is a high possibility that the watermark bits cannot be properly embedded because the P-frames contain much lower energy. Furthermore, as discussed in Section 3.2.2, choosing lower value for this parameter will allow us to use fewer DCT blocks per watermark bit and thus enable us to embed more watermark bits in the P-frame. Since the average variance of the P-frames in our test sequences is lower by a factor of approximately 50, by choosing an *enforced energy difference* of 4 (which is 20% of the value chosen for the I-frames), we expect that we would need as few as around ten times as many DCT blocks to embed one watermark bit in a P-frame, with similar performance as embedding the watermark in an I-frame, instead of 50. The *minimal cut-off point* is set at 6 for both cases, in order to avoid too much image degradation due to the watermarking process.

3.3.2. Watermark capacity

As noted in Section 3.2, the watermark capacity is determined by the number of 8 × 8 DCT blocks that are used to embed one watermark bit. The number of watermark bits that can be embedded in one frame can be computed as follows:

$$W_b = \left\lfloor \frac{F_p}{B \times 64} \right\rfloor \text{ bits} \qquad (3.2)$$

In the equation above, W_b is the number of watermark bits in the frame, F_p is the total number of pixels in the frame and B is the number of 8 × 8 DCT blocks per watermark bit. The watermark bit-rate is computed simply as the total number of embedded watermark bits divided by the length of the sequence in seconds.

When 4CIF-sized sequences and the default parameter (64 blocks per bit) are used, a label rate of 0.21 kbps is achieved [7]. By reducing this number to 32 blocks per bit, a capacity of 0.42 kbps is achieved. The sequences used in our experiment are in CIF format, which is a quarter as large as the 4CIF sequence. Therefore, using the default parameter a label rate of 50 bps can be achieved. We call this watermark bit-rate the *base capacity*. In the following sub-sections, we investigate the performance of the original and the extended DEW algorithms in two areas, namely *below* and *above* this base capacity.

3.3.2.1. Performance below base capacity

In order to investigate the behavior of both algorithms below base capacity, we use various settings yielding watermark bit-rates ranging from 2 bps to 50 bps. For the I-frames, we use various numbers of DCT blocks per watermark bit, ranging from 128 to 1024 blocks. For the P-frames, we use 512 or 1024 DCT blocks per watermark bit. The settings we use, the achieved watermark bit-rates and the achieved *e*BER are presented in Table 3.1.

Table 3.1. Settings, Watermark Bit-rate and eBER of the original and extended DEW algorithm below base capacity (Claire, 256 kbps)

DCT blocks/water mark bit (I-frame)	P-frame settings					
	No watermark bits in P-frames (DEW)		1024 DCT blocks/ watermark bit (XDEW)		512 DCT blocks/ watermark bit (XDEW)	
	Watermark bitrate (bps)	*e*BER (%)	Watermark bitrate (bps)	*e*BER (%)	Watermark bitrate (bps)	*e*BER (%)
1024	2.2	0	8.6	0	21.5	0
512	6.7	0	13.1	0	26	0
256	13.5	0	19.9	0	32.7	0
128	27	0	33.3	0	46.2	0

Table 3.1 only shows the *e*BER of a sequence encoded at 256 kbps, but the results for sequences encoded at other bit-rates are identical. We can observe from the results that both the original and the extended DEW algorithm performs well for watermark bit-rates below the base capacity. Furthermore, the results show that the P-frames contain enough energy to accommodate the watermark if we use 512 or 1024 DCT blocks to embed one watermark bit.

3.3.2.2. Performance above base capacity

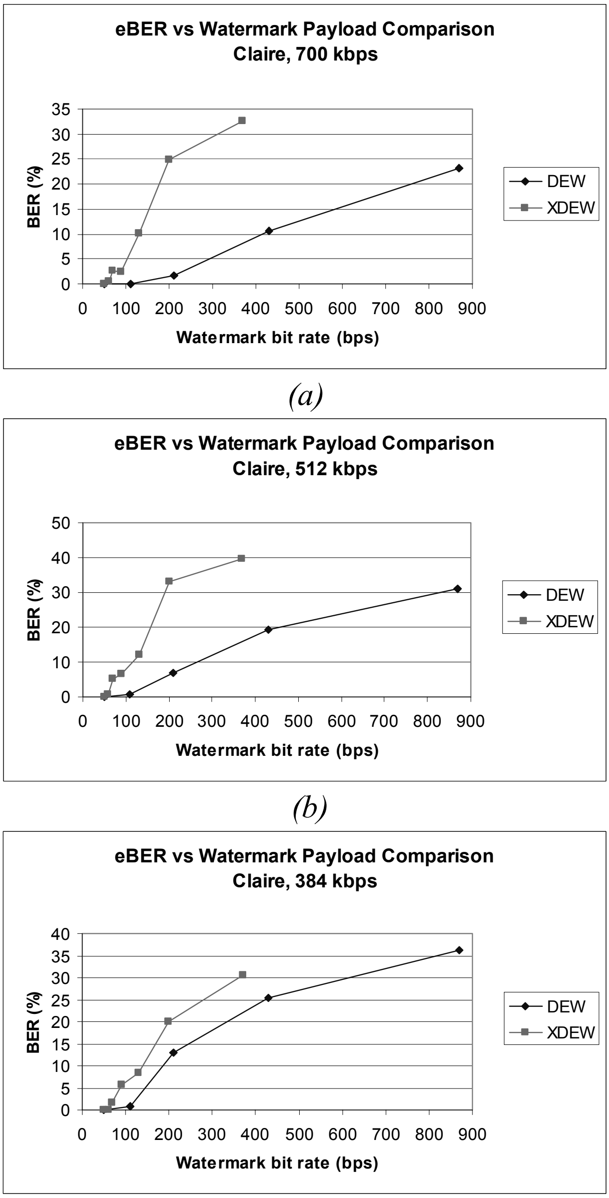
The base capacity of 50 bps may not be sufficient for all applications, as some applications may require that the watermark bit-rate is at least 70 bps [8]. We compare two approaches to increase the capacity. The first approach is to use the original DEW algorithm but reduce the number of blocks used to encode each watermark bit and the second approach uses the extended DEW algorithm (with various numbers of blocks to encode each watermark bit in the P-frame) using the default parameter for the I-frames (64 blocks/bit). In the first approach, we can achieve various bit-rates of 50 to 870 bps. In the second approach, we achieve bit-rates of 50 to 370 bps. The relation of the number of blocks per watermark bit and the achieved watermark bit-rate is presented in Table 3.2.

Table 3.2. Relation between number of blocks/watermark bit and watermark bit-rate

Original DEW		Extended DEW (64 DCT blocks/I-frame)	
DCT Blocks/bit (I-frame)	Watermark Bitrate (bps)	DCT Blocks/bit (P-frame)	Watermark Bitrate (bps)
64	50	1024	60
32	110	512	70
16	210	256	90
8	430	128	130
4	870	64	200
		32	370

We compare the two approaches by evaluating the watermark *e*BER produced by each approach as we increase the number of bits embedded in the streams. The results of this experiment are presented in Figure 3.3.

From Figures 3.3(a) to 3.3(d), we can see that for all encoded bit-rates, except 256 kbps, it is much more attractive to use the original DEW algorithm and reduce the number of blocks used to encode each watermark bit rather than using the P-frames to gain extra space to embed more watermark bits. For a 256 kbps encoded bit-rate, both approaches seem to yield the same performance. This is because the I-frames in such a stream have already lost much of the high-frequency DCT components due to the nature of MPEG quantization. As an example, a comparison of the variance of the DCT coefficients between two matching I-frames, one taken from a sequence encoded at 700 kbps and the other from a sequence encoded at 256 kbps, is shown in Figure 3.4.

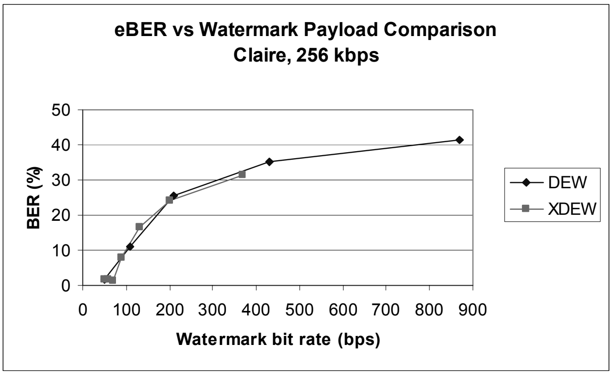


(a)

(b)

(c)

Figure 3.3. Watermark eBER for sequences encoded at various bit rates, watermarked using the DEW algorithm and the Extended DEW (XDEW) algorithm with various payloads



(d)
Figure 3.3 (continued)

Figure 3.4 shows only the variances of the DCT coefficients with indices 6 up to 63, because the coefficients with lower index (indices lower than 5) have relatively similar variance and also these coefficients do not play an important role in the enforcement of the energy difference. As we can see, the energy content of the I-frame of the sequence encoded at 256 kbps is lower than the one of the sequence encoded at 700 kbps. As the discussion in Section 3.2.2 points out, this means that in order to enforce the same energy difference with the same number of DCT blocks per watermark bit, a lower *minimal cut-off point* should be chosen. And since in our experiments this parameter is fixed, there are some watermark bits that cannot be properly embedded. It should also be noted that the variances plotted in Figure 3.4 are from the collection of blocks and the condition of individual DCT blocks might be worse (i.e., there is less energy available to enforce the energy difference).

From Figure 3.3, we can also see that for the original DEW algorithm relatively low *e*BER (2.5% bit error or less, which is roughly equal to a BER of 10^{-3}) is achieved only for watermark payloads below 110 bps, which means that the number of blocks used to embed each watermark bit is halved (from 64 blocks/watermark bit to 32 blocks/watermark bit). For a sequence encoded at 700 kbps, this number can still be achieved at a payload of 210 bps. Meanwhile, for the sequence encoded at a bit-rate of 256 kbps, even a watermark payload of 110 bps produces more than 10% *e*BER. For the Extended DEW algorithm, the numbers are even lower. For the sequence encoded at 700 kbps, a payload of up to 90 bps can be achieved, while for other bit-rates, this number drops to around 70 bps.

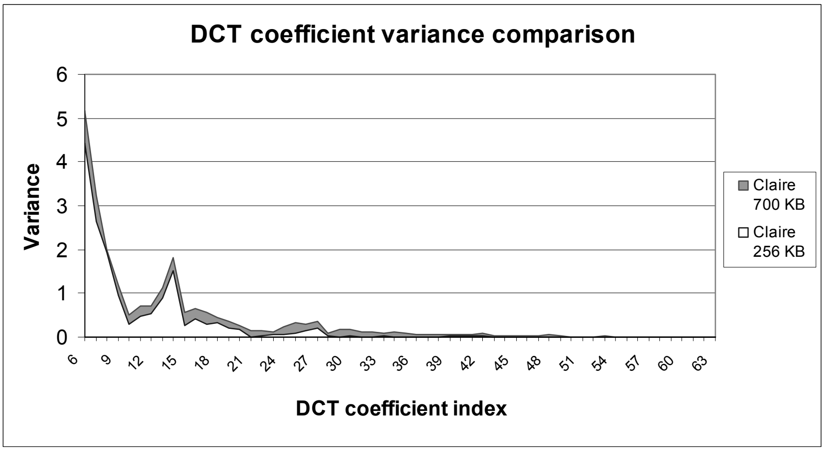


Figure 3.4. Comparison of the variance of theDCT coefficients from 2 I-frames Taken from sequences encoded at different bit-rates

These numbers show that for the I-frames a proper energy difference can no longer be enforced when less than 32 DCT blocks are used to encode one watermark bit (16 DCT blocks, in the case of sequences encoded at 700 kbps), while for the P-frames the minimum number of DCT blocks that should be used to embed one watermark bit is 512 blocks (256 DCT blocks for sequences encoded at 700 kbps).

3.3.3. Watermark robustness

We test the watermark robustness by re-encoding the watermarked stream at a lower bit-rate and then seeing whether the watermark survives the operation. Watermark survival is measured by the *a*BER of the watermark. The reencoding operation we performed is illustrated in Figure 3.5.

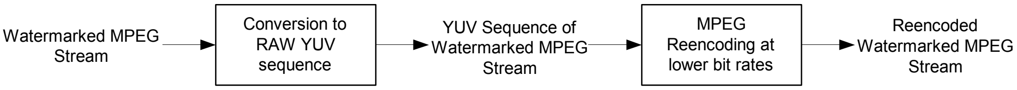
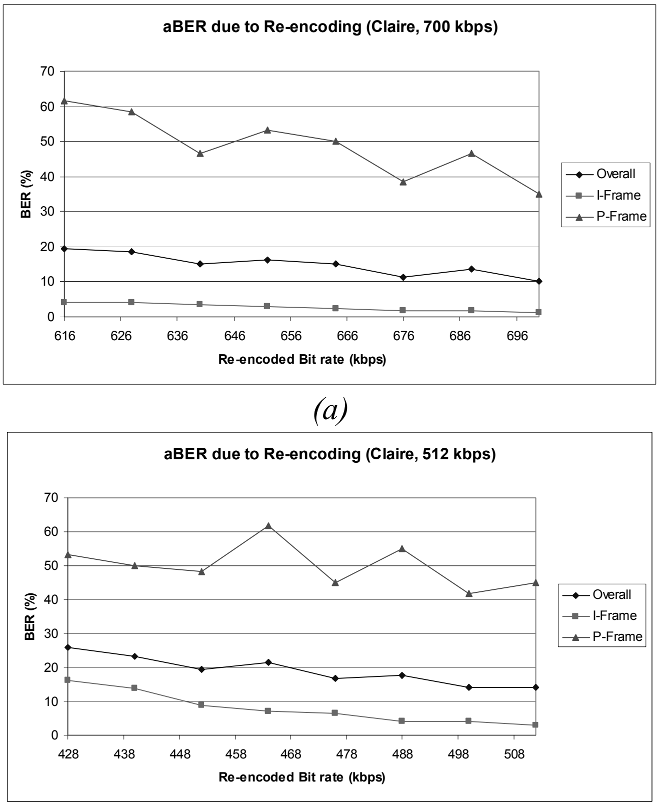

Watermarked MPEG Stream
Conversion to RAW YUV sequence
YUV Sequence of Watermarked MPEG Stream
MPEG Reencoding at lower bit rates
Reencoded Watermarked MPEG Stream

Figure 3.5. Re-encoding procedure

The *a*BER is computed from a re-encoded sequence previously watermarked using the Extended DEW algorithm, and is presented in Figure 3.6 for the I-frames and the P-frames separately. Since the watermarks embedded in the I-frames and the P-frames are detected independently, the watermark *a*BER for the I-frames can be interpreted as the watermark *a*BER of the original DEW algorithm after re-encoding. The behavior observed in Figure 3.6 is typical for all sequences in our experiments.

As can be observed in Figure 3.6, the watermark embedded in the I-frames performs quite well after re-encoding to a lower bit-rate. However, the watermark embedded in the P-frames is severely damaged by the re-encoding operation. The reason for this phenomenon is twofold. In the first place, the P-frames are predicted from a *different version* of the I-frame (i.e., an already watermarked I-frame) during MPEG compression which yields a *different* error signal from the one originally contained in the watermarked MPEG stream. Since this error signal is where the watermark is embedded, this difference will introduce errors in the watermark detection. Furthermore, the re-quantization process introduces further differences in the P-frames, which in turn introduce errors in the watermark detection process. We measure the difference between the original and the re-encoded watermarked P-frames by computing the average correlation value of the matching P-frames of the original watermarked sequence (Claire, encoded at 512 kbps) and the sequence re-encoded at 428 kbps. The average correlation value is 0.57, which is quite low and shows that there are indeed significant differences between the original and the re-encoded P-frames. The effects of re-encoding to the I-frames are caused by the re-quantization process, but the effects are not very significant due to the high energy content of the I-frames.



(a)

(b)

Figure 3.6. Watermark aBER due to reencoding at lower bit rates for sequences encoded at 700 and 512 kbps. The sequences are watermarked using the Extended DEW algorithm with a watermark payload of 70 bps

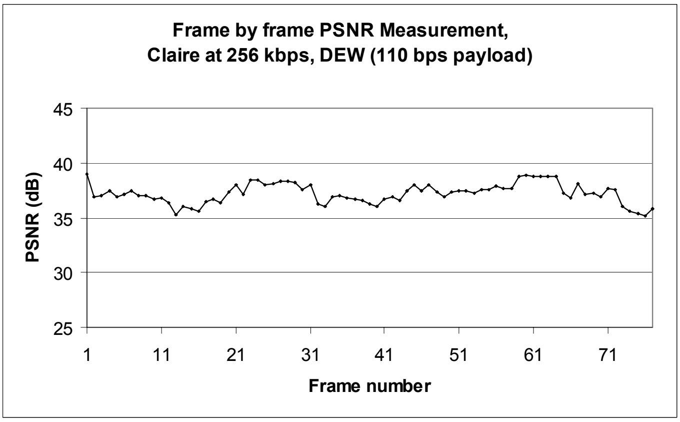
Applying pre-quantization to the data prior to watermarking has been shown to increase the watermark robustness against re-encoding [7]. Pre-quantization is done using a standard MPEG quantization procedure with a certain Q factor. The watermark embedded in the I-frames does indeed have higher robustness when pre-quantization is used. But pre-quantization does not seem to have any positive effect on the robustness of the watermark embedded in the P-frames. Actually, pre-quantization with low Q value seems to have a negative effect on the watermark embedded in the P-frames because in many cases the DCT blocks do not have enough energy to enforce the energy difference determined by the selected pre-quantization Q factor. The end result is an actually higher watermark *e*BER.

3.3.4. Visual quality impact

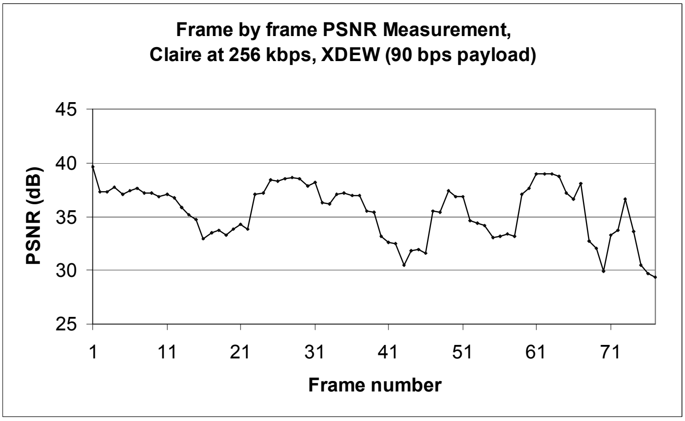
The visual quality is assessed objectively and subjectively. The objective assessment is done by measuring the PSNR value of the video stream watermarked using both the original and the extended DEW algorithm compared to the unwatermarked stream. The subjective assessment is done by visually judging the quality of the watermarked material. The behavior of the PSNR curves of the extended and the original DEW algorithm is different, because in the original algorithm we only process the I-frames directly while in the extended algorithm we process the I- and the P-frames. This is apparent at higher watermark bit-rates (i.e., when 256 DCT blocks or fewer are used to encode one watermark bit in the P-frames). An example of such behaviour of the PSNR curves is shown in Figure 3.7.

The different behavior may seem to indicate that it is unfair to simply compare the time-averaged PSNR values produced by the algorithms, and that it is more appropriate to compare only the quality of the processed (watermarked) frames. However, despite the difference in this curve behavior, we decide to use the time-averaged PSNR value to compare the performance of the two algorithms because we consider this to be a better representation of the overall quality of the watermarked material, and this will be the main concern of somebody who is viewing the watermarked material.

The results of the objective visual quality assessment are presented separately for watermark bit-rates below and above the base capacity. The results for watermark bit-rates below the base capacity are shown in Table 3.3. In this table, the values in the brackets below the PSNR values are the difference between the PSNR values of the watermarked sequence and the unwatermarked sequence, which is 37.57 dB. The results for watermark bit-rates above the base capacity are presented in Figure 3.8. The results in both cases are presented only for sequences encoded at 256 kbps, but the results for other encoded bit-rates show similar behavior, except as noted below.



(a)



(b)

Figure 3.7. Frame by frame PSNR measurements of a sequence encoded at 256 kbps, watermarked using:
(a) the original algorithm (DEW, watermark payload=110 bps)
and (b)the extended algorithm (XDEW, watermark payload=90 bps)

Table 3.3 shows that the performances of both the original and extended DEW algorithms are very similar when the watermark bit-rate is below the base capacity. Furthermore, we can see from this table that both algorithms incur virtually no visual degradation to the sequence being watermarked. This means that for these watermark bit-rates, the required energy difference can be enforced without removing too many DCT coefficients.

However, Figure 3.8 shows the much sharper decrease in visual quality of the sequences watermarked using the extended algorithm compared to the one watermarked using the original algorithm as the watermark payload is increased. We can also observe that at 256 kbps and low payload (up to 70 bps), the performance of both algorithms is similar, but the visual quality of the sequence watermarked using the extended DEW algorithm rapidly deteriorates as the payload is increased above this level. This is not observed at the other encoded bit-rates, where the performance of the two algorithms is very different for all watermark bit-rates. In both cases, the visual degradation introduced by both the original and the extended DEW algorithms is much higher than the one introduced when the watermark bit rates are below base capacity. This means that more DCT coefficients have to be removed to enforce the energy difference for watermark bit-rates above the base capacity.

Table 3.3. Visual quality impact assessment for the original and extended DEW algorithm, below base capacity

DCT blocks/watermark bit (I-frame)	P-frame settings					
	No watermark bits in P-frames (DEW)		1024 DCT blocks/ watermark bit (XDEW)		512 DCT blocks/ watermark bit (XDEW)	
	Watermark bitrate (bps)	PSNR (dB)	Watermark bitrate (bps)	PSNR (dB)	Watermark bitrate (bps)	PSNR (dB)
1024	2.2	37.57 (0)	8.6	37.56 (-0.01)	21.5	37.56 (-0.01)
512	6.7	37.56 (-0.01)	13.1	37.56 (-0.01)	26	37.55 (-0.02)
256	13.5	37.54 (-0.03)	19.9	37.54 (-0.03)	32.7	37.54 (-0.03)
128	27	37.52 (-0.05)	33.3	37.52 (-0.05)	46.2	37.52 (-0.05)

Our experiments also show that the visual quality degradation is sharper at higher bit-rates. This is observed in sequences watermarked using both algorithms. This means that at lower bit-rates, the MPEG coding artefacts start to play a bigger role in the overall visual quality of the sequences, which is not the case in higher bit-rates. In other words, the watermarking artefacts are dominated by the coding artefacts at lower bit-rates.

The subjective quality assessment reveals that no watermarking artefacts are visible when the watermark bit-rate is below the base capacity. However, some artefacts become visible at high payloads, albeit only in some frames. The artefacts are visible as blurred edges in some blocks and blotches. The blotches appear due to the drift effect, and are sometimes visible in the sequences watermarked using the extended DEW algorithm, especially at higher watermark payloads and lower bit-rates. These blotches appear (and disappear again) gradually over time, except when the blotches appear in a frame directly preceding an I-frame, in which case the blotches disappear completely when the I-frame is displayed. These blotches are the reason why for some frames the PSNR of the sequence watermarked using the extended algorithm drops to a low value, as can be seen in Figure 3.7 (b).

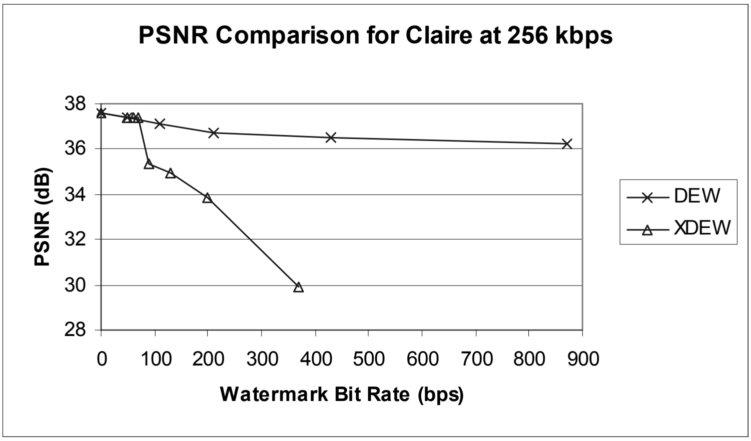


Figure 3.8. Visual quality impact comparison between the DEW algorithm and the extended DEW (XDEW) algorithm for watermark bit rates above base capacity.

3.4. Conclusions

We have presented in this chapter the results of our investigation on the performance of the DEW and Extended DEW algorithm in a low bit-rate environment according to three performance criteria, namely payload, robustness and visual quality. The results can be summarised as follow:

1. From the payload point of view, both the original DEW and the extended DEW algorithms perform similarly when the watermark bit-rate is low (below base capacity) for all encoded video bit-rates. However, the extended algorithm generally performs worse than the original DEW as the watermark capacity is increased beyond the base capacity. This is observed in all encoded video bit-rates, except for the bit-rate of 256 kbps, where the performance of the two algorithms is comparable.
2. From the robustness point of view, the watermark embedded in the P-frames is very vulnerable to re-encoding. This is true for all sequences used in this experiment.
3. From the visual quality point of view, both the original and the extended DEW algorithms perform similarly when the watermark bit-rate is below base capacity, and both incur virtually no visual degradation to the data. However, the extended DEW algorithm shows rapid visual quality degradation as the watermark payload is increased. Furthermore, drift effects are visible in the sequences watermarked using the extended algorithm. We also observe that the visual quality curves for the DEW algorithm become less steep as the encoded bit-rates become lower. This is apparent from the fact that, as the encoded bit-rate decreases, encoding artefacts play a larger role to determine the overall quality of the sequence.

In other words, the watermarking artefacts become less significant or dominated by the coding artefacts. For the Extended DEW algorithm, the same can be said, but only for lower watermark payloads (less than 70 bps). For higher payloads, the watermarking artefacts are still more significant than the coding artefacts.

Based on these results we draw the following conclusions:

1. The original DEW algorithm scales well to lower bit-rates/smaller spatial resolution for watermark payloads of up to 110 bps, except for the sequence encoded at 256 kbps.
2. The extension scheme to the DEW algorithm we have presented works reasonably for low payloads (up to watermark payload of 70 bps). This is especially true for watermark bit-rates below base capacity. For this payload level, the *e*BER is still relatively low while the introduced watermarking artefacts are either negligible or dominated by the coding artefacts.
3. From these two preceding statements, we can conclude that the Extended DEW algorithm should not be used to pursue higher payloads. To achieve higher watermark payload it is better and easier to adjust the number of blocks used to encode each watermark bit. Even then, the watermark capacity can not be pushed beyond 110 bps without incurring severe *e*BER.
4. Finally, at low bit-rates, the limitations of the MPEG-1/-2 encoder become more obvious. The coding artefacts become visible and at very low bit-rates the specified encoding bit-rate cannot be achieved due to the overhead associated with MPEG-1/-2 stream. Therefore, further developments in low bit-rate video watermarking should be focused on formats more suitable to such bit-rates, like MPEG-4 or H.263.

3.5. Final remarks

Further research on watermarking techniques of low bit-rate video has been performed since the work presented in this chapter was originally published in 2001. Two examples of recent works on low bit-rate video watermarking found in the literature are briefly discussed below.

The first example, presented in [10], uses semi-fragile watermarks to assess the Quality of Service (QoS) of the communication link between cellular video communication devices. The QoS of the communication link is evaluated by measuring the distortion suffered by the embedded watermarks due to a noisy communication channel. Due to the inherent limitations of cellular devices, the size and bit-rate of the video are limited. In their experiments, the authors used QCIF sequences encoded using an MPEG-4 encoder at bit rates of 200 – 1000 kbps. The proposed scheme uses spread spectrum watermarking technique. The watermark message *w* is embedded in each Video Object (VO) of the MPEG4-coded video. The VO is first transformed into DCT domain, then the watermark pattern is added to the mid-band frequency DCT coefficients of the VO. On the detection side, w_i' is estimated from each VO, where $i = 1,\ldots, n$ is the number of VO's in one video frame. An estimate of the embedded watermark, w', is then computed by averaging all w_i' in one frame. By calculating the Mean Square Error (MSE) between the original watermark message w and the estimated watermark message w', the quality of the communication link can be estimated. In [10], the authors introduce random bit errors (with adjustable BER's) to simulate a noisy channel. Their experiments show that the increase of the measured MSE corresponds well to the increase of the introduced BER. Therefore, the proposed technique demonstrates that semi-fragile watermarks can be used to estimate the quality of the communication link in an additive noise environment.

The second example is a spread-spectrum based watermarking scheme for low bit-rate (128 – 768 kbps) MPEG-4 video proposed in [11]. The proposed scheme is an extension of the scheme proposed in [5]. The watermark is embedded in DCT domain, so that full decoding of the compressed MPEG-4 bit stream is not necessary. The detection process, however, is performed in spatial domain. This gives the advantage of enabling watermark detection even when the watermarked MPEG-4 video is re-encoded using another compression algorithm. However, the disadvantage is that the proposed scheme cannot use MPEG-4-specific properties in the detection process.

Robustness against synchronization attacks (rotation, translation and scaling (RTS) transform of the video frame) is provided by using synchronization templates. The first template is constructed in an approach similar to the one used in [6], namely by tiling the message-carrying watermark pattern periodically over the video frame. In the autocorrelation domain these tiles will produce periodic peaks that can be used to recover watermark synchronization. The second template is a purely synchronization signal. The signal is constructed in a similar manner to that proposed in [12]. In the frequency domain, this signal is composed of peaks in the mid-frequency band. Each peak occupies one frequency coefficient with pseudo-random phase. This synchronization signal provides additional synchronization capability especially in low bit-rate environment where a large part of the watermark is lost due to the compression process [11]. Estimation of the RTS transform parameters is performed in log-polar domain. After the parameters are estimated, the RTS transform is reversed, thus re-synchronizing the watermark. The approach used to increase watermark robustness against geometric distortion limits the amount of watermark bits that can be embedded. In [11], the total number of information embedded (excluding error-correcting code bits) is 31 bits.

To improve watermark imperceptibility, the authors use an adaptive gain control mechanism that adjusts watermark signal strength based on the local “activity” of the original video. Small gain is applied to low-activity, i.e., smooth areas and larger gain is applied to high-activity (textured) areas. Furthermore, since the proposed scheme is similar to the one described in [5], drift compensation is also implemented to prevent error propagation when a watermarked frame is used to predict another frame. Finally, the authors also implement a heuristic-based optimization approach to control bit-rate. In this approach, the bit-rate control tries to balance the increase in bit-rate due to the watermarking process and bit allocation based on the local gain factor. The details of this optimization approach are provided in [11]. Bit-rate control is needed to prevent the size of the watermarked video from consuming substantially more bits compared to the original video stream. This type of bit-rate control mechanism is not needed for the DEW or XDEW algorithms since the watermark is embedded by discarding DCT coefficients, thus guaranteeing that the resulting watermarked video data will be smaller than the original video data. If the size of the watermarked video data is to be maintained, dummy bits can be added to replace the discarded DCT coefficients.

The results of the experiments show that the watermark detection rate is quite high even when the watermarked video is attacked using filtering, scaling, rotation and transcoding. The performance of the adaptive gain control is not yet optimal for video segments with a lot of movements since it has not taken the temporal properties of the watermarked video into account. The bit-rate control implemented is shown to be able to limit the increase of the video bitstream size to under than 5%. The complete description of the test setup and results are provided in [11].

3.6. References

1. H. Brynhi, H. Lovett, E. Maarmann-Moe, D. Solvoll, T. Sorensen, *On-demand Regional Television over the Internet*, ACM Multimedia '96, Boston, MA, 1996
2. J. Dittmann, T. Fiebig, R. Steinmetz, S. Fischer, I. Rimac, *Combined Video and Audio Watermarking: Embedding Content Information in Multimedia Data*, in Proceedings of SPIE, Security and Watermarking of Multimedia Content II, Vol. 3971, pp. 455 – 464, San Jose, CA, 2000
3. F. Hartung, B. Girod, *Watermarking of uncompressed and compressed video*, Signal Processing, Vol. 66, No. 3, pp. 283 – 301, May 1998
4. F. Hartung, P. Eisert, and B.Girod, *Digital Watermarking of MPEG-4 facial animation parameters*, Computer Graphics, Vol. 22, No. 3, pp. 425 – 435, 1998
5. F. Hartung, M. Kutter, *Multimedia Watermarking Techniques*, Proceedings of the IEEE, Vol. 87, No. 7, Special Issue: Identification & Protection of Multimedia Information, pp. 1079 – 1107, July 1999
6. T. Kalker, G. Depovere, J. Haitsma, M. Maes, *A video watermarking system for broadcast monitoring*, in Proceedings of SPIE, Security and Watermarking of Multimedia Contents, vol. 3657, pp. 103 – 112, San Jose, CA, January 1999
7. G.C. Langelaar, *Real-time watermarking techniques for compressed video data*, Ph.D. dissertation, Delft University of Technology, The Netherlands, January 2000
8. G.C. Langelaar, I. Setyawan, R.L. Lagendijk, *Watermarking Digital Image and Video Data*, IEEE Signal Processing Magazine, Vol. 17, No. 5, ISSN 1053-5888, pp. 20 – 46, September 2000
9. G.C. Langelaar, R. L. Lagendijk, *Optimal Differential Energy Watermarking of DCT Encoded Images and Video*, IEEE Transactions on Image Processing, Vol. 10, No. 1, January 2001
10. P. Campisi, G. Giunta, A. Neri, *Object-based Quality of Service Assessment using Semi-fragile Tracing Watermarking in MPEG4 Video Cellular Devices*, in Proceedings of IEEE, ICIP 2002, Vol. II, pp. 881 – 884, Rochester, NY, 2002
11. A.M. Alattar, E.T. Lin, M.U. Celik, *Digital Watermarking of Low Bit-Rate Advanced Simple Profile MPEG4 Compressed Video*, IEEE Trans. On Circuits and Systems for Video Technology, vol. 13, no. 8, pp. 787 – 800, August 2003
12. J.J.K. Ó Ruanaidh, T. Pun, *Rotation, scale and translation invariant digital image watermarking*, in Proceedings of IEEE, ICIP 1997, Vol. I, pp. 536 – 539, Santa Barbara, CA, 1997

Chapter 4
GEOMETRIC DISTORTION AND WATERMARK SYNCHRONIZATION

4.1. Introduction

One of the most difficult tasks faced by image and video digital watermarking algorithm developers is resistance to geometrical attacks. Geometrical transformations may take simple forms such as rotation, translation and scaling. They could also take much more complex forms, for example rubber-sheet stretching. A famous example of the latter is the StirMark software package that can perform a wide range of minor, unnoticeable geometrical transformations on an image, including slight stretching, bending or shifting [1]. Another example of geometrical attack is the Digital Cinema Attack [2]. In this scenario, the geometrical attack is not actually performed on the watermarked data directly. Instead, the attacker records a (watermarked) movie being shown on the cinema screen using a camera. The result of this recording is then illegally distributed on video CD's or put on the internet for downloading. The quality of the recording is influenced by numerous factors, e.g., the position of the camera with respect to the cinema screen, the (usually low) quality of the lenses in the camera and the fact that the cinema screen itself is not perfectly flat. All these factors contribute to the complex combination of geometrical transformations applied to the recorded video.

Geometrical transformation attacks do not actually remove the watermark from the data. Instead, they work by exploiting the fact that most watermarking techniques rely on the synchronization between the watermark and the watermark detector. If this synchronization is destroyed, the detector can no longer correctly detect the presence of the watermark in the data, although the watermark itself (or a major part thereof) might still remain in the data. As a simple example, let us take a simple image watermarking algorithm that embeds a pseudo-random noise pattern all over the image and detects the

The work presented in this chapter has been published in two separate papers:

- *Synchronization-insensitive video watermarking using structured noise pattern*, by I. Setyawan, G. Kakes and R.L. Lagendijk, in the Proc. of SPIE, Security & Watermarking of Multimedia Contents IV, Vol. 4675, San Jose, CA, 2002
- *Complexity-scalable compensation of geometrical distortions in image watermarking*, by P.J.O. Doets, I. Setyawan and R.L. Lagendijk, in Proc. ICIP 2003, Vol. I, Barcelona, 2003.

embedded watermark by correlating the watermarked image with an identical pseudo-random noise pattern [3]. A high correlation value signifies the presence of the watermark. If this watermarked image is now cropped by removing 5% of the pixels, the correlation value produced by the detector would be very low although most of the embedded noise pattern itself is still present in the image.

Some approaches have been developed in the literature to deal with attacks that destroy the synchronization between the watermark and the detector. In this chapter, we propose two approaches designed to deal with this synchronization problem. The first approach we present is a new watermarking approach that is less sensitive to synchronization. The basic idea underlying our approach is to remove the requirement for strict synchronization between the watermark and the detector. The second approach we present is an approach that allows the synchronization of the watermark to be recovered by inverting the geometric distortion.

This chapter is organized as follows. In Section 4.2, an overview of the existing approaches to combat attacks that destroy the synchronization between the watermark and the detector are discussed. In Section 4.3, we present a watermarking scheme that does not rely on the rigid spatial synchronization between the watermark and the watermark detector. We present the basic ideas, design considerations and our basic implementation of this idea. We also present the evaluation of this basic implementation. In Section 4.4, we present the second proposed approach to the synchronization problem. In this approach, we invert the geometric distortion to recover the spatial synchronization. We present the details of this approach and the evaluation of its performance. Finally, in Section 4.5, we present our conclusions.

4.2. Existing techniques to combat geometrical transformation

Existing techniques found in the literature to combat geometrical transformation generally belong to the following categories of approaches, each with its own strengths and weaknesses:

1. **Performing the watermarking operation in a domain that is invariant to geometric transformation.** In this category, watermark embedding and detection is performed in a domain that is resistant to geometric transformation. Thus, the original data is first transformed into this domain, the watermark is embedded in this domain and then the data is transformed back to its original domain. On the detection side, the received data is again transformed into the domain invariant to geometric transformation. A watermark detection operation is then performed in this domain. An example of this approach is given in [4]. In this scheme, the log-polar map domain is used. This domain is invariant to rotation, translation and scaling operations. The advantage of this approach is that if the correct invariant

domain could be found, then the watermark would be resistant to the geometric transformations for which this domain is invariant. In the previous example, the watermark would be highly resistant to rotation, translation and scaling operations. However, finding a domain that would be resistant to all possible geometrical transformations is very difficult, if at all possible.

2. **Using re-synchronizeable watermarks.** The second category consists of approaches that embed a watermark that can be re-synchronized after a geometrical transformation. One implementation of this approach is to embed identifiable marker signals in the watermark with certain relative positions [5]. After an attack, the absolute positions of the markers are likely to change. However, since the correct relative position is known, it is possible to recover the original positions and thus re-synchronize the watermark. Another example is to use watermark synchronization points based on the invariant features of the watermarked data. Using these points, the synchronization can be recovered after an attack [6]. There is a security issue with this approach, namely that an attacker can also detect the marker signal. It is therefore possible that he can remove or jam it. To prevent this, measures must be taken to hide or secure this signal [17, 18].

3. **Reversing or compensating the distortions caused by the geometrical transformation.** The third category consists of approaches that attempt to reverse or compensate the effects of the geometrical transformation and restore the data to its original state. One way to do this is to compare the attacked data to the original undistorted data. Based on this comparison, a restoration to the data's original state is then attempted. If the restoration is successful, then the probability of correctly detecting the watermark would be high since the data processed by the detector would be very similar to the original watermarked data. An example of this technique is given in [7]. The advantage of this approach, when it works, is obvious. The watermark detection problem becomes virtually identical to the problem of detecting a watermark in non-attacked data. The second approach we propose in this chapter belongs to this category.

Another possible approach that is not widely used in the literature, is exhaustive search. In this approach, the watermark detector exhaustively searches the space of all possible geometric transformations. This approach is not widely used primarily due to the fact that the search space is very large, and thus the computational cost of searching the correct geometric distortion will be unfeasibly high. Another, and more important, reason is that this approach can produce an unacceptable rate of false positive in the watermark detection as shown in [8].

All of the approaches discussed above have one common objective, namely to resist or recover from attacks that disturb the synchronization between the watermark and the detector. These approaches aim to achieve this objective by either making the synchronization of the watermark robust (i.e., hard to destroy) or making the synchronization retrievable after an attack. However, all of these methods are still dependent on the synchronization (i.e., dependent on the proper phase information) in order to function properly. The first approach we propose in this chapter deals with this problem from a different point of view, i.e., by removing the synchronization requirement. In order to achieve this objective, we must remove the dependence of the watermark on phase information. This approach is discussed further in the next section.

4.3. Structured noise pattern watermarking

4.3.1. Basic idea

Our new watermarking approach is based on two new central ideas. First, the watermark payload is embedded in the geometrical structure of the embedded (invisible) *watermark patch*. An example of such a structure is the presence of a hole in the patch (representing watermark bit “0”) or the absence of such a hole (representing watermark bit “1”). The choice of the geometrical structure of the watermark patch to embed depends on which watermark bit is to be embedded. Second, the watermark patch is embedded as a *colored noise* pattern. This colored noise can be discriminated on the detection side by a properly designed filter sensitive to such noise. Both the filter used to color the noise and the filter used to detect it depends on a secret key. Using this technique, we embed the information not in the phase of the watermark signal, but in its frequency distribution. Figure 4.1 presents an overview of our approach.

The watermark detector is designed so that it does not need to know exactly the spatial position of the watermark patch. This is needed to ensure that the system does not have to rely on rigid spatial synchronization. This will also limit the complexity of the detector, because the detector does not need to perform complex operations to search for the position of the patch. Furthermore, this will also ensure that the system can still recover the watermark even when a portion of the patch is missing. Another advantage of this design is that the watermark embedder could embed the patch virtually anywhere within the frame. Thus, sophisticated embedding strategies can be developed so that the embedder can choose the optimal embedding location in order to reduce its visibility (i.e., choosing areas with more masking capability) or to increase its robustness (i.e., choosing areas that are important to the legibility of the frame, thus forcing an attacker to preserve them). The system is also designed to work without requiring the presence of the original, unwatermarked frame. This requirement is based on the fact that the original frame may not be available in every practical application.

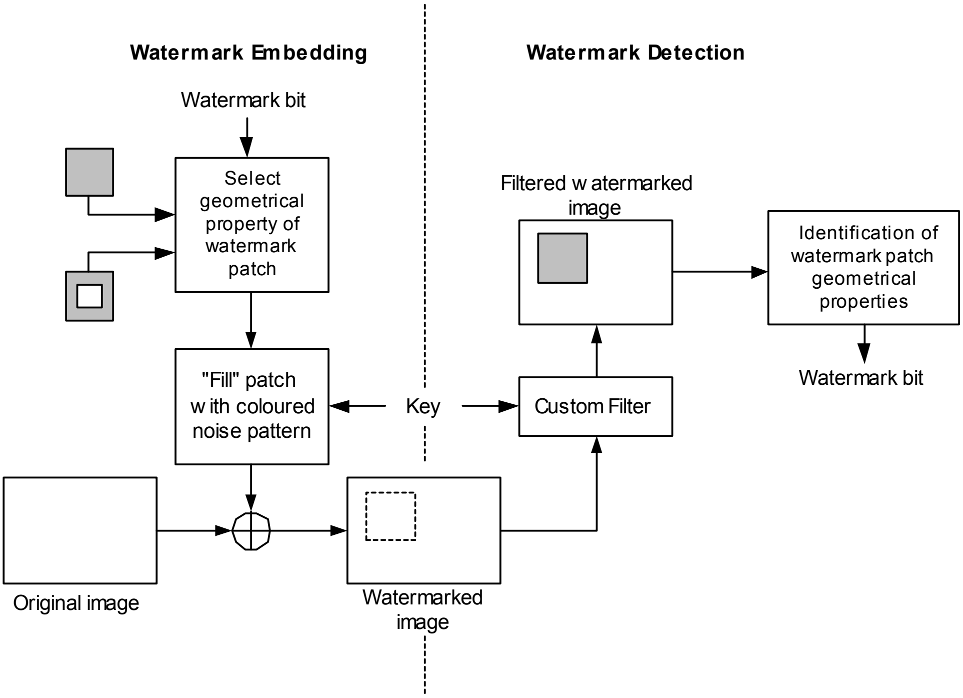

Watermark Embedding
Watermark Detection
Watermark bit
Select geometrical property of watermark patch
"Fill" patch with coloured noise pattern
Key
Custom Filter
Filtered watermarked image
Identification of watermark patch geometrical properties
Watermark bit
Original image
Watermarked image

Figure 4.1. Basic block diagram of the proposed approach.

From the previous discussion, the design challenges for our approach can be classified into two different areas. The first one is directly related to the process of patch embedding and detection. The basic objectives of this part of the design process are as follows. In the first place, we have to design an identifiable pattern, i.e., pattern that could later be detected by the watermark detector. Thus, the pattern must possess certain properties that can be picked up by the filter in the watermark detector. Secondly, we have to keep the security of the pattern, i.e., the generation and detection of the patch should depend on a secret key. Without knowledge of this key one should not be able to see the pattern (either completely or partially). Otherwise, a potential attacker could generate a random key and then modify the pixel values of the frame while observing the response of the detector to determine what level of modification is sufficient to disable watermark detection. These two requirements may be contradictive, i.e., a highly structured pattern is easy to detect but also easy to guess, while a random pattern is much more difficult to guess but is also hard to detect. The second area of the design challenge has more to do with the actual classification and identification of the properties of the embedded pattern in order to recover the watermark bit. This problem is very similar to the problems of image segmentation and pattern recognition. In this chapter, we concentrate on the first design challenge, namely implementing the basic idea of being able to embed and detect a watermark patch constructed using a colored noise pattern.

The watermark embedder performs the following operation on the original frame to embed a watermark patch

$$\begin{aligned} X_w(i,j) &= X(i,j) + \gamma \cdot \left(P_0(x,y \mid K) * n(u,v)\right), && X(i,j) \in \text{patch} \\ &= X(i,j), && \text{otherwise} \end{aligned} \tag{4.1}$$

In Equation (4.1), $X_w(i,j)$ represents the value of a pixel in the watermarked frame, $X(i,j)$ represents the value of a pixel in the same position of the non-watermarked frame, $P_0(x,y|K)$ represents the filter (that depends on the key K) used to color the pseudo-random noise $n(u,v)$ and γ is a constant used to control the gain (and thus the visibility) of the watermark patch. Without loss of generality, we assume that the variance of the colored noise $P_0(x,y|K)*n(u,v)$ is equal to 1.

On the detection side, the received watermarked frame is filtered using a custom filter $P_1(i,j|K)$, thus

$$X_{wf}(i,j) = P_1(x,y \mid K) * X_w(i,j) \tag{4.2}$$

Substituting $X_w(i,j)$ with the expression from Equation (4.1), we can rewrite Equation (4.2) as follows:

$$\begin{aligned} X_{wf}(i,j) &= P_1(x,y \mid K) * X(i,j) \\ &\quad + \gamma \cdot \left(P_1(x,y \mid K) * P_0(x,y \mid K) * n(u,v)\right), && X_{wf}(i,j) \in \text{patch} \\ &= P_1(x,y \mid K) * X(i,j), && \text{otherwise} \end{aligned} \tag{4.3}$$

In Equations (4.2) and (4.3), $X_{wf}(i,j)$ represents the value of a pixel of the filtered watermarked image and $P_1(x,y|K)$ represents the filter in the watermark detector, again depending on the same key, K. From Equation (4.3) we can observe that the response of the filtering operation for areas within the watermark patch is different from the response of areas outside the watermark patch.

To detect the watermark patch, we calculate the local variances of the filtered watermarked frame, expressed here in the Fourier domain. For simplicity, we drop the K notation of the filters. The local variances of the watermarked frame within the watermark patch can be expressed as follows:

$$\begin{aligned}\sigma_{wf}{}^{2} &= \frac{1}{4\pi}\int_{-\pi}^{\pi}\int_{-\pi}^{\pi}\left|P_1(\omega_1,\omega_2)\right|^2 S_x(\omega_1,\omega_2)d\omega_1 d\omega_2 \\ &+\frac{\gamma^2\sigma_n^2}{4\pi}\int_{-\pi}^{\pi}\int_{-\pi}^{\pi}\left|P_0(\omega_1,\omega_2)\right|^2\left|P_1(\omega_1,\omega_2)\right|^2 d\omega_1 d\omega_2\end{aligned} \tag{4.4}$$

On the other hand, in areas outside the watermark patch, the local variances can be expressed as follows:

$$\sigma_{wf}{}^{2} = \frac{1}{4\pi}\int_{-\pi}^{\pi}\int_{-\pi}^{\pi}\left|P_1(\omega_1,\omega_2)\right|^2 S_x(\omega_1,\omega_2)d\omega_1 d\omega_2 \tag{4.5}$$

The detection of the embedded patch is based on the shift of the local variance in areas within the watermark patch, as shown by the presence of the second term in Equation (4.4). In order to get the best performance of the watermark detector, we would have to do some optimization. From Equation (4.4) we can see that optimization of the detection problem could be done in several ways. We will discuss here two optimization of approaches. The first one, $C_1[P_1(\omega_1,\omega_2)]$, is to try to minimize the first term of Equation (4.4) while maximizing the second term, i.e.,

$$\begin{aligned}&\max_{P_1(\omega_1,\omega_2)}\left(\frac{\gamma^2\sigma_n^2}{4\pi}\int_{-\pi}^{\pi}\int_{-\pi}^{\pi}\left|P_0(\omega_1,\omega_2)\right|^2\left|P_1(\omega_1,\omega_2)\right|^2 d\omega_1 d\omega_2\right) \\ &\min_{P_1(\omega_1,\omega_2)}\left(\frac{1}{4\pi}\int_{-\pi}^{\pi}\int_{-\pi}^{\pi}\left|P_1(\omega_1,\omega_2)\right|^2 S_x(\omega_1,\omega_2)d\omega_1 d\omega_2\right)\end{aligned} \tag{4.6}$$

The second optimization approach is to minimize the ratio between the first and the second terms of Equation (4.4), i.e.,

$$\begin{aligned}C_2[P_1(\omega_1,\omega_2)] &= \min_{P_1(\omega_1,\omega_2)}\left(\frac{\frac{1}{4\pi}\int_{-\pi}^{\pi}\int_{-\pi}^{\pi}\left|P_1(\omega_1,\omega_2)\right|^2 S_x(\omega_1,\omega_2)d\omega_1 d\omega_2}{\frac{\gamma^2\sigma_n^2}{4\pi}\int_{-\pi}^{\pi}\int_{-\pi}^{\pi}\left|P_0(\omega_1,\omega_2)\right|^2\left|P_1(\omega_1,\omega_2)\right|^2 d\omega_1 d\omega_2}\right) \\ &= \min_{P_1(\omega_1,\omega_2)}\left(\frac{\int_{-\pi}^{\pi}\int_{-\pi}^{\pi}\left|P_1(\omega_1,\omega_2)\right|^2 S_x(\omega_1,\omega_2)d\omega_1 d\omega_2}{\gamma^2\sigma_n^2\int_{-\pi}^{\pi}\int_{-\pi}^{\pi}\left|P_0(\omega_1,\omega_2)\right|^2\left|P_1(\omega_1,\omega_2)\right|^2 d\omega_1 d\omega_2}\right)\end{aligned} \tag{4.7}$$

To normalize the optimization problems, we assume (in addition to the aforementioned variance of the colored noise) that the variance of P_1 is also equal to 1. These constraints are repeated below for convenience.

$$\frac{1}{4\pi}\int_{-\pi}^{\pi}\int_{-\pi}^{\pi}\left|P_1(\omega_1,\omega_2)\right|^2 d\omega_1 d\omega_2 = 1$$

$$\frac{1}{4\pi}\int_{-\pi}^{\pi}\int_{-\pi}^{\pi}\left|P_0(\omega_1,\omega_2)\right|^2 d\omega_1 d\omega_2 = 1 \tag{4.8}$$

$$\sigma_n^2 = 1$$

Let us first consider the first optimization approach, presented in Equation (4.6). We can rewrite this equation and try to minimize the following:

$$C_1[P_1(\omega_1,\omega_2)] = \min_{P_1(\omega_1,\omega_2)}\left(\begin{array}{l}\frac{1}{4\pi}\int_{-\pi}^{\pi}\int_{-\pi}^{\pi}\left|P_1(\omega_1,\omega_2)\right|^2 S_x(\omega_1,\omega_2)d\omega_1 d\omega_2 - \\ \qquad \frac{\gamma^2\sigma_n^2}{4\pi}\int_{-\pi}^{\pi}\int_{-\pi}^{\pi}\left|P_0(\omega_1,\omega_2)\right|^2\left|P_1(\omega_1,\omega_2)\right|^2 d\omega_1 d\omega_2\end{array}\right) \tag{4.9}$$

Let us assume that *P₁(ω₁,ω₂)* = *P₁(-ω₁,-ω₂)*. We can therefore rewrite Equation (4.9) as follows:

$$C_1[P_1(\omega_1,\omega_2)] = \min_{P_1(\omega_1,\omega_2)}\left(\frac{1}{4\pi}\int_{-\pi}^{\pi}\int_{-\pi}^{\pi}\left|P_1(\omega_1,\omega_2)\right|^2\left\{S_x(\omega_1,\omega_2)-\gamma^2\sigma_n^2\left|P_0(\omega_1,\omega_2)\right|^2\right\}d\omega_1 d\omega_2\right) \tag{4.10}$$

The solution of Equation (4.10) is therefore:

$$\begin{aligned} P_1(\omega_1,\omega_2) &= \delta(\omega_1,\omega_2), && \text{when } S_x(\omega_1,\omega_2)-\gamma^2\sigma_n^2\left|P_0(\omega_1,\omega_2)\right|^2 \text{ is minimum} \\ &= 0, && \text{otherwise} \end{aligned} \tag{4.11}$$

Equation (4.11) means that the optimal P_1 must be able to pick one single frequency pair (ω_1,ω_2) where the original image data has relatively low energy compared to the energy of the colored noise.

Let us now evaluate the second optimization approach, presented in Equation (4.7). First, let us rewrite this Equation, dropping the factors γ^2 (since it is obvious that larger γ will give a better detection) and σ_n^2 since it is assumed to be equal to 1. Furthermore, assuming that we have $P_0(\omega_1,\omega_2), P_1(\omega_1,\omega_2) \in \Re$, we will proceed to minimize $P_1^2(\omega_1,\omega_2)$ instead of $P_1(\omega_1,\omega_2)$. For notational simplicity, we will drop the squares notation. Thus we have

$$C_2[P_1(\omega_1,\omega_2)] = \min_{P_1(\omega_1,\omega_2)} \left(\frac{\int\limits_{-\pi}^{\pi}\int\limits_{-\pi}^{\pi} P_1(\omega_1,\omega_2) S_x(\omega_1,\omega_2) d\omega_1 d\omega_2}{\int\limits_{-\pi}^{\pi}\int\limits_{-\pi}^{\pi} P_0(\omega_1,\omega_2) P_1(\omega_1,\omega_2) d\omega_1 d\omega_2} \right) \tag{4.12}$$

To evaluate Equation (4.12), we will use its 1-D discrete implementation for purposes of simplicity, namely

$$C_2[P_1(l)] = \min_{P_1(l)} \left(\frac{\sum\limits_{l} P_1(l) S_x(l)}{\sum\limits_{l} P_0(l) P_1(l)} \right), \qquad 0 \le l \le N \tag{4.13}$$

In Equation (4.13), N represents the length of the DFT.

We have performed experiments with various $P_0(l)$ and $S_x(l)$, and in all of those experiments, the $P_1(l)$ solution that minimizes Equation (4.13) satisfies the following condition:

$$\begin{aligned} p_1^n &= \delta, && \text{for } n = l \text{ where } \frac{S_x(l)}{P_0(l)} \text{ is minimum} \\ &= 0, && \text{otherwise} \end{aligned} \tag{4.14}$$

In Equation (4.14), p_1^n represents the n^{th} element of $P_1(l)$ ($0 \le n \le N$). In other words, P_1 only has a response at a single frequency, namely the frequency in which the original data has relatively low energy compared to the colored noise energy.

We can thus draw a conclusion that both optimization approaches lead to the same conclusion, namely that the optimal P_1 must be able to pick one single spatial frequency, namely the frequency where the energy of the original image data is relatively lower compared to the energy of the colored noise. However, if P_1 is implemented in this way, it would be very sensitive to attack and small variations to the data. Furthermore, we are interested in solutions that are spatially localized. Therefore, our choice of frequency for P_1 would depend heavily on the spatial location we are looking at. Since image data typically has low-pass behavior, we can conclude that the optimal frequencies to choose for P_1 would be the high frequencies. Furthermore, to make the system more robust, we will design P_1 so that it detects a range of frequencies (i.e., the frequency range of P_0) instead of just selecting one frequency. In other words, P_1 should be constructed as a bandpass filter that can suppress the signal with frequencies outside the frequency range of P_0, while passing a signal within this frequency range.

4.3.2. Basic practical implementation and evaluation

In this section, we present a basic practical implementation of the proposed watermarking approach and the evaluation of this basic implementation. We will first describe the implementation of the watermark embedder and detector, then we will present the results of our experiments. We perform our test by watermarking raw video sequences with CIF resolution. In our experiments, we use P_0 and P_1 of size 8 × 8. Furthermore, the size of the pseudo-random noise pattern (and hence, the size of the watermark patch) is chosen to be a disc with a diameter of 120 pixels.

4.3.2.1. Watermark embedding

The watermark embedder follows the block diagram of the watermark embedding process presented in Figure 4.1. Since the video frames are treated as individual frames, the scheme could also be applied to still images. For the sake of simplicity, both still images and video frames will be referred to collectively as images. The watermark is embedded as a circular patch of structured noise pattern by modifying the luminance of the pixels of a chosen area of the image. In our experiment, these areas are either chosen at random or determined specifically by the user. In other words, we have not used any of the sophisticated embedding strategies mentioned in the previous section. The choice of a disc as the shape of the patch is made for simplicity, although obviously other shapes can also be used. The modification of the value of the luminance is done according to Equation (4.1a). We use γ equals to 7, because in our experiments we found that γ value of 7 or less gives an acceptable distortion to the watermarked image.

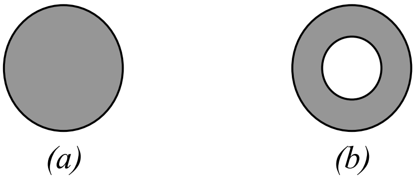

(a) *(b)*

Figure 4.2. The watermark patches:
(a) Patch representing O_A, (b) Patch representing O_B

Depending on the watermark bit to be embedded, either patch O_A or O_B is embedded. If watermark bit “1” is to be embedded, then patch O_A is chosen. Otherwise, patch O_B is chosen. In our experiments, we choose a solid disc to represent O_A and a disc with a “hole” in it to represent O_B. These patches are shown in Figure 4.2. For patch O_A, the entirety of image pixels within the boundaries of the disc is modified according to Equation (4.1). For patch O_B, only the areas between the two discs (the gray areas in Figure 4.2(b)) are modified. The sizes (determined by diameter of the disc) of O_A and O_B are user adjustable (within the constraints of the actual size of the image to be watermarked). This adjustment can be made independently for each patch. For the sake of simplicity we choose to make the sizes of O_A and O_B identical. Choosing the size of the patterns is a trade-off between watermark visibility and reliability of detection. Smaller sizes may be less conspicuous (for an equal γ), but will make the patch harder to detect reliably. The “hole” (inner disc) of patch O_B is put exactly in the middle of the outer disc in our experiments. Its size is user adjustable, and is expressed as a fraction of the area of the outer disc. The size of the inner disc will also determine the reliability of detection for patch O_B. If the ratio is too big, then the “walls” would be too thin and the patch would be harder to detect reliably. On the other hand, if the ratio is too small, the detection performance would also suffer because the two patches may become not easily distinguishable.

4.3.2.2. Watermark detection

The watermark detector follows the block diagram of the watermark detection process presented in Figure 4.1. As discussed in the previous section, the watermark detection procedure consists of two stages and we focus only on the first stage of this procedure. The first stage is a processing applied to the image that may contain a watermark. This processing is done to “reveal” the watermark patch embedded in it. This stage is illustrated in Figure 4.3. In this Figure, X_w represents the received watermarked image. The custom filter P_1 is a filter which depends on the same key as the one used to generate the filter during the watermark embedding procedure. The size of the kernel also has to be identical. X_{wf} is the filtered version of the received watermarked image.

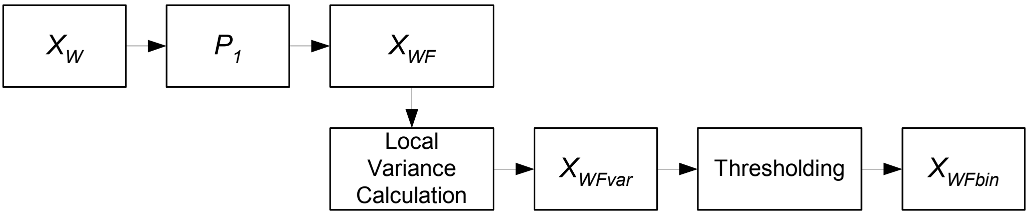
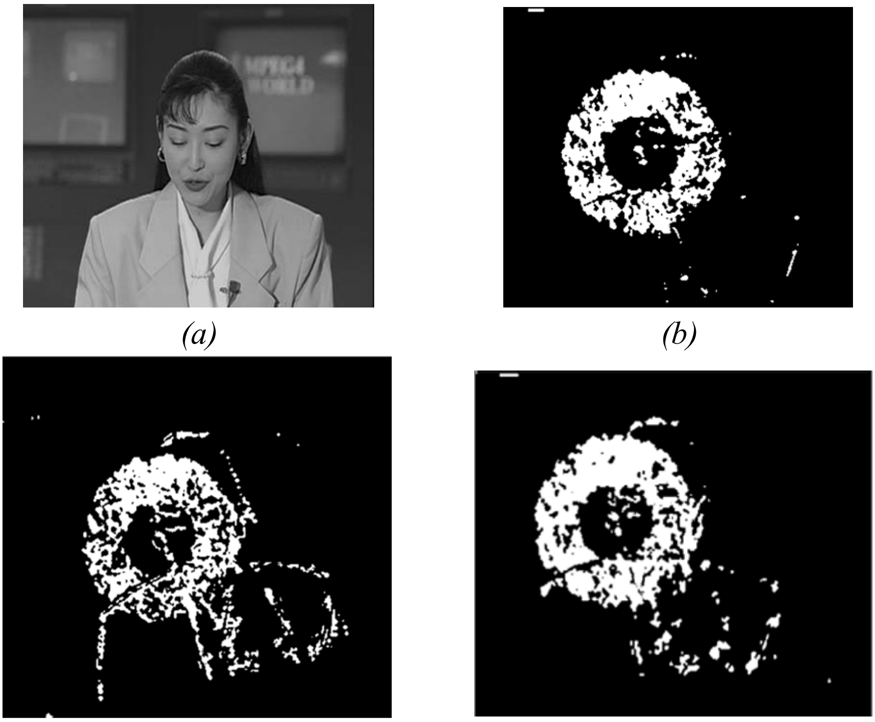


Figure 4.3. Process to reveal the watermark patch

The processing of the watermarked image proceeds as follows. First, a copy of the received watermarked image is filtered using P_1, resulting in X_{wf}. Then the local variances of this filtered image are calculated. The local variances calculation is done using a sliding window, i.e., variances are calculated within overlapping blocks of $n \times n$ pixels. In our experiments, we choose $n = 3$. The result of this processing is saved into X_{wfvar}. By examining X_{wfvar}, we observe that the values of the variances of the watermarked areas fall within a certain range. We then proceed to threshold X_{wfvar} and produce a binary image X_{wfbin} which is the final output of this procedure and would become the input of the second stage of the watermark bit detection process (identification of the geometrical properties of the watermark patch). The thresholding is done as follows:

$$\begin{aligned} X_{wfbin}(i,j) &= 255, && T_1 < X_{wfvar}(i,j) < T_2 \\ &= 0, && \text{otherwise} \end{aligned} \tag{4.15}$$

In other words, we designate the areas containing the watermark patch by assigning them a value of 255 (white) and the other areas are designated as non-watermarked by assigning them the value 0 (black). The choice of the thresholds is a trade-off between the probability of missed detection and probability of false alarm (i.e., designating an area outside the watermark area as watermarked). Choosing a narrow gap between T_1 and T_2 will increase the probability of missed detection (i.e., designating an area within the watermark patch as not-watermarked) but reduce the probability of false alarm and vice versa.



(a) (b)

(c) (d)

Figure 4.4. Experiment results:
(a) Original watermarked image,
(b) X_{wfbin}, with P_1 constructed by convoluting P_0 with a high-pass filter kernel,
(c) X_{wfbin}, when X_{wf} is rotated by 10°,
(d) X_{wfbin} when X_{wf} is reduced in size by 10%.

In our experiments, we observe that using P_1 that is identical to P_0 does not give a satisfactory result. In other words, the shift in the variances is too small to be detected. Therefore, we have to perform some optimization on the detector. We will follow the optimization approach discussed in Section 4.3.1. We therefore construct P_1 as a convolution of a high-pass filter and P_0. The high-pass filter is used to suppress the influence of the original image data, since it has primarily low-pass behavior. This filter has the following kernel.

$$\begin{bmatrix} -22 & -35 & -22 \\ -35 & 288 & -35 \\ -22 & -35 & -22 \end{bmatrix}$$

Optimizing the detector using this high-pass filter gives an improved detection result. The result of our experiment is presented in Figure 4.4. Figure 4.4(a) shows the watermarked image before processing. Figure 4.4(b) shows X_{wfbin} with P_1 constructed as a convolution of a high-pass filter and P_0. Figure 4.4(c) shows X_{wfbin} when the watermarked image is rotated by 10° and then cropped, and Figure 4.4(d) shows X_{wfbin} when the watermarked image is resized by a factor of 10%. In all of these experiments, patch O_B is embedded.

4.4. Complexity-scalable compensation of geometric distortions

This section describes the second proposed approach to deal with the watermark synchronization problem. This approach is based on a strategy for inverting the geometrical distortion with the use of the original image. The term strategy refers to the choice of the transformation class, the degree of locality of the transformation and the method of estimating the parameters, given a set of correspondence vectors. These correspondence vectors can be generated by comparing the detected template and a reference template or, alternatively, by comparing the distorted image and a reference image (by using a motion vector field or by applying feature point detection in combination with point matching). An example of such a strategy is the application of an affine transform, first on a global scale, later on a more local scale [9]. Another example is the application of a translation (e.g. resulting from block matching) in a coarse to fine approach [10]. While in previous approaches the choice of the spatial transformation is fixed, the emphasis in this chapter is on a strategy that is *scalable* in terms of transformation complexity. The strategy is to choose a transform that is as complex as needed to enable watermark detection, but still as simple as possible. In the end, the goal of the strategy is not the perfect registration of the image, but the recovery of the spatial synchronization of the watermark.

4.4.1. Proposed strategy

The image registration problem can be formalized as follows. The spatial transform is determined using corresponding points in a distorted image (i_k, j_k) and a reference image (i_k', j_k'). The corresponding points can be the result of point matching, block matching, etc. The proposed method uses an approximation method, i.e., given a set of N corresponding points, the function F is determined such that the corresponding points from the distorted and the reference images map as closely as possible. In other words,

$$(i_k', j_k') \approx F(i_k, j_k) \quad k = 1, \ldots, N \tag{4.16}$$

Function F can then be used to register the distorted image.

The strategy consists of the consecutive estimation of a transform that is more complex than the previous one. The one that fits best, according to some criterion, is chosen to be applied on the distorted image, prior to watermark detection.

The scheme is shown schematically in Figure 4.5. Basically, the transformation complexity is progressively increased, minimising the Mean Square Error (MSE) between the reference and registered image coordinates, hereafter to be called the *point error*. Minimising the point error does not necessarily minimise the registration error, i.e., the MSE between the pixel values of the registered image and reference image. The transformation does not control what happens to points that lie in between the N corresponding points used to compute the transform. Therefore, we base our choice of the optimally registered image on the registration error. We assume that minimisation of the registration error yields an approximation of an optimal detection probability:

$$\begin{aligned} \min_F BER &\approx \min_F d(X_{we}, X) \\ &\approx \min_F d(F(i_k, j_k), (i_k', j_k')) \\ X_{we} &= estim[X_w] \end{aligned} \tag{4.17}$$

where X and X_w represent the unwatermarked and watermarked reference images, respectively. X_{we} is the estimation of the watermarked reference image X_w. The function $d(.)$ denotes the MSE function. If the registration error has decreased, the complexity of the transformation is increased and the procedure is repeated. Otherwise, the previous attempt is kept. The optimally registered image is used to detect a watermark.

4.4.2. Orthogonal polynomial mapping

We choose to use a polynomial mapping to implement the complexity-scalable strategy presented in the previous section. The order of the polynomial mapping determines the complexity of the spatial transformation and provides a logical ranking in the degree of complexity in subsequent transformations. Commonly used transformation types such as rotation, scaling and affine transforms, are subclasses of polynomial transformations.

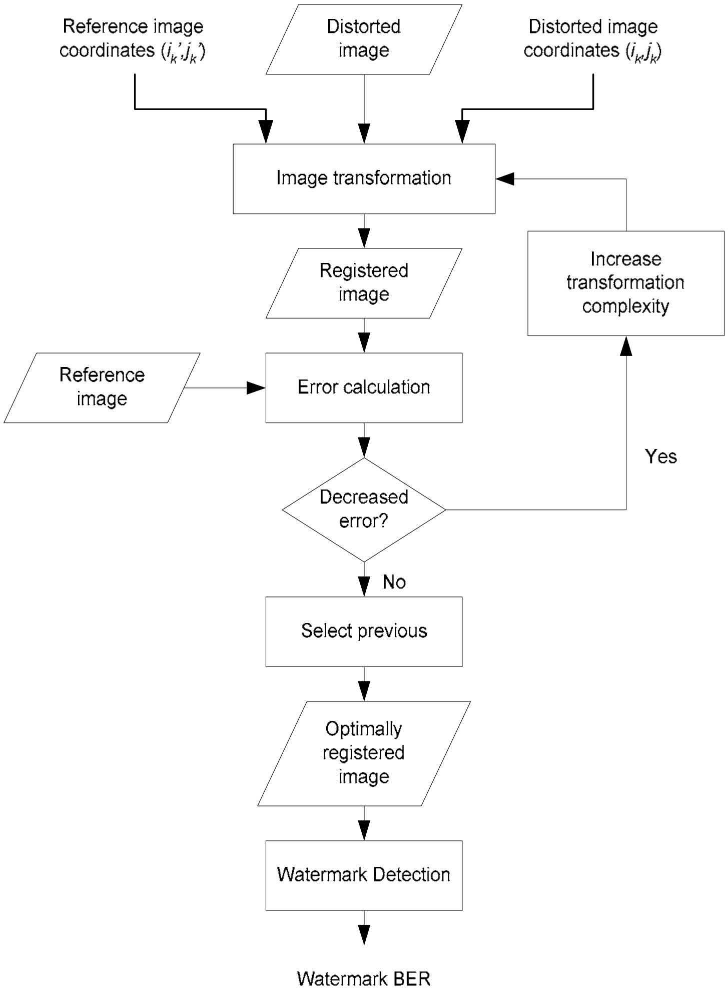


Figure 4.5. Flowchart of the proposed strategy.

For the polynomial mapping, orthogonal polynomials have been used. This type of polynomial mapping has been used previously for image registration purposes [11]. Due to the orthogonality of these polynomials, the complexity of the transformation can be increased without the need for recalculation of the parameters of lower order polynomials.

Equation (4.16) is split into two scalar functions, which are more convenient to implement [12]:

$$\begin{aligned} i_k' &\approx f(i_k, j_k) \\ j_k' &\approx g(i_k, j_k) \quad k = 1, \dots, N \end{aligned} \tag{4.18}$$

In the following, only the transformation *f* mentioned in Equation (4.18) will be determined; *g* follows in the same manner.

A set of *M* polynomials

$$P_1(i, j), P_2(i, j), \dots, P_M(i, j) \tag{4.19}$$

is orthogonal over points (i_k, j_k), k = 1, …, N if the following relation holds between them:

$$\sum_{k=1}^{N} P_x(i_k, j_k) P_y(i_k, j_k) = 0 \quad x, y = 1,2,3,\dots,M \quad x \neq y \tag{4.20}$$

The function *f* using polynomials becomes:

$$f(i, j; a_1, a_2, \dots, a_M) = \sum_{x=1}^{M} a_x P_x(i, j) \tag{4.21}$$

The number of matched points, *N*, should be much larger than the number of parameters to be estimated, *M*. If *M* is equal to the number of non-collinear points, these points will map exactly on top of each other. If $M << N$, there is some robustness to overcome mismatches and to increase spatial accuracy. The exact number of non-collinear points that is needed to yield a good registration result depends on the spatial accuracy of the used points and the presence of mismatches.

Using a set of *M* linearly independent functions

$$h_1(i, j), h_2(i, j), \dots, h_M(i, j) \tag{4.22}$$

a set of orthogonal polynomials can be determined by the Gram-Schmidt orthogonalisation process. This process uses the following notation to represent orthogonal functions

$$\begin{aligned} P_1(i,j) &= w_{11}h_1(i,j) \\ P_2(i,j) &= w_{21}P_1(i,j) + w_{22}h_2(i,j) \\ &\cdots \\ P_M(i,j) &= \sum_{y=1}^{M-1} w_{My}P_y(i,j) + w_{MM}h_M(i,j) \end{aligned} \tag{4.23}$$

To calculate the mapping *f*, w_{xy} and a_x have to be determined. The parameters w_{xy} ($x = 1, .., M$; $y = 1, .., x$) are computed by setting the w_{x1} values to $w_{x1} = 1$ for all *x*, and applying the least-squares criterion and the orthogonalization property to the polynomials [12]. This results in:

$$w_{xx} = -\frac{\sum_{k=1}^{N} [P_1(i_k, j_k)]^2}{\sum_{k=1}^{N} P_1(i_k, j_k) h_i(i_k, j_k)} \qquad x = 2,...,M \tag{4.24}$$

$$w_{xy} = -w_{xx} \frac{\sum_{k=1}^{N} P_y(i_k, j_k) h_x(i_k, j_k)}{\sum_{k=1}^{N} [P_y(i_k, j_k)]^2} \qquad \begin{array}{l} x = 2,...,M \\ y = 2,...,M-1 \end{array} \tag{4.25}$$

This result is used to estimate the parameters of function *f* in Equation (4.21) using the following relation:

$$a_x = \frac{\sum_{k=1}^{N} i_k' P_x(i_k, j_k)}{\sum_{k=1}^{N} [P_x(i_k, j_k)]^2} \tag{4.26}$$

In our implementation, the following functions $h_i(x, y)$ are used:

$$
\begin{aligned}
h_1(i,j) &= i^0 j^0 = 1 \\
h_2(i,j) &= i^1 j^0 = i \\
h_3(i,j) &= i^0 j^1 = j \\
h_4(i,j) &= i^2 j^0 = i^2 \\
h_5(i,j) &= i^1 j^1 = ij \\
&\cdots
\end{aligned}
\tag{4.27}
$$

Thus, the complexity of the spatial transform can be iteratively increased, by increasing the order of the desired mapping *f* and evaluating all polynomials having equal or smaller order. Thus, there is no preference for the *i* or *j* directions.

Performing least squares with orthogonal polynomials offers several advantages over ordinary polynomials. First, closed-form expressions for the calculation of parameters are available (Equations 4.25 – 4.27), eliminating the need to solve a system of equations. Further, the parameters of lower order terms do not have to be recalculated when a higher order polynomial mapping is deemed necessary.

Distortions that actually are polynomial mappings themselves are likely to be easily corrected. Some (non-linear) distortions that can be effectively approached by polynomial expansion are likely to be corrected effectively, given enough correct matching points. On a global scale, highly non-linear distortions are expected not to be corrected effectively, yielding a high BER after distortion correction.

4.4.3. Implementation and results

In the test setting, a multibit watermark (144 bits) is embedded in a gray scale image. The watermarking bits are embedded in the spatial domain by adding pseudo-noise patterns to image pixel blocks [3]. This scheme is chosen because it is quite sensitive to geometrical distortion and therefore gives a good indication of the performance of the distortion compensation.

In [13] and [14], several feature point detectors are evaluated. Both select the Stephens and Harris corner detector [15], among others because it preserves most feature points after geometrical distortion. The detector used has pixel accuracy in the spatial dimension.

4.4.3.1. Perfect match experiments results

In our controlled experiments, the feature points were matched artificially, fully exploiting the knowledge of the applied distortion, resulting in a *perfect match*. By perfect match we mean that no matching errors are present. In the case of feature points, only points that are detected in both the reference and the distorted image are used. This was done to assess the potential of the strategy in a mismatch-free environment. In the image registration experiments, three distributions of corresponding points are evaluated:

1. Uniformly distributed grid markers (625);
2. A large number of matched points from a Harris detector (350-500);
3. A smaller number of matched points from a Harris detector (50-100);

Grid markers are artificially created markers, not present in the image, that undergo the same distortion as the image. Besides simple RST, affine, projective and bilinear transforms, several bending transformations have been applied, including sinusoidal bending, the barrel and pincushion transform [12].

The experimental setup is shown in Figure 4.6. On the left side of the figure, watermark *m* is embedded in a test image. Then, one of the distortions listed in Table 4.1 is applied on the image. The distorted image, X_{wd}, is registered using the strategy presented in Section 4.4.2. Watermark detection is performed on the registered image, X_{we}. Comparison of the detected watermark, *m'*, with the original watermark *m* yields the BER.

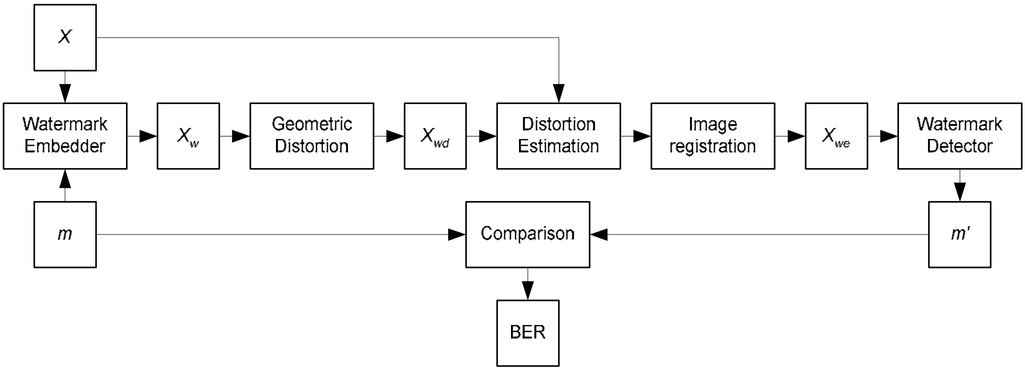


Figure 4.6. Image registration scheme

In the estimation and correction block, feature points (or grid markers) are extracted from both the distorted and the reference image. Exploiting the knowledge of the applied distortion, the match between the corresponding points is generated. The distorted image is registered using the strategy described in earlier sections.

The results of our experiment for the Lena image are shown in Table 4.1. The results can be compared with the BERs of the unregistered, i.e., distorted, image (listed in the first column of the table). Also listed are the BERs for detection when the exact inverse of the distortion is applied on the image. Because the exact values of the BERs depend on the interpolation schemes used in the distortion and the detection, not one value is listed, but the minimum, maximum, mean and median values are listed. The last columns list the mean polynomial order that was selected for each distortion type and distribution of corresponding points. The images were not cropped after distortion. The performance of the proposed scheme is lower when the images are cropped, but otherwise shows a similar behavior [12].

Table 4.1. Measured BERs (in %) of the experiment using perfectly matched corresponding points for the Lena image.

		Bit Error Rate								Selected		
		D	Inversion				Registration			polynomial order		
		Distorted	Min	Median	Mean	Max	Gridmarkers	500 Feature points	100 Feature points	Gridmarkers	500 Feature points	100 Feature points
Distortion applied	Affine	50	1	4	5	12	1	3	4	2	2	1
	Barrel	47	1	9	7	13	5	10	16	9	5	5
	Bending	46	0	2	2	6	1	1	4	4	4	3
	Bilinear	49	0	3	3	10	1	1	2	3	3	2
	Pincushion	49	0	3	3	8	1	10	32	7	8	5
	PLM	49	0	1	1	1	13	11	31	6	8	5
	Projective	51	1	3	4	11	1	2	3	3	2	2
	Rotation	50	1	3	4	10	1	2	1	2	1	1
	Scale	49	1	4	4	11	7	3	3	2	1	1
	Sinus	49	1	3	3	6	4	5	29	8	8	5
	Sinus+Bending	51	1	4	4	10	3	5	20	8	8	6
	Translation	10	4	9	15	52	10	5	5	1	2	1

4.4.3.2. Experiment results with mismatches

To assess the performance of the system under the presence of mismatches, mismatches are artificially introduced in the corresponding points set obtained in the experiments described in the previous section.

There are several ways to introduce the mismatches. For example, they can be introduced such that a correspondence vector is limited in length (complying with the constraint of limited visual distortion), but has an arbitrary direction. However, this is not realistic: a matching procedure can have some capabilities to detect and correct mismatches. Mismatches that are more difficult to correct, are the mismatches that have almost the same direction and length as some of the matching vectors. Therefore, mismatches are introduced based on the direction and the standard deviation of the correct matches. For each of the experiments and distortions listed in the previous section, an increasing percentage of mismatches was introduced (1%, 5%, 10%, 15 and 20% mismatches).

In some cases, the degrees of freedom to introduce mismatches had to be increased, especially in cases when only few feature points were available in the correct match, in combination with a high percentage of mismatches. In these situations, the feature points on average are located quite far-off. This is an extra effect, degrading the performance when fewer correspondence vectors are used.

Table 4.2. Measured BERs (in %) of the experiments with mismatches introduced. The numbers are the averaged results of the Lena image.

		GridMarkers						500 Feature points						100 Feature points					
	Mismatches (%)	0	1	5	10	15	20	0	1	5	10	15	20	0	1	5	10	15	20
Transform	Affine	1	3	6	10	13	22	3	4	8	14	21	29	4	8	30	36	38	37
	Barrel	5	11	26	33	38	38	10	16	40	38	44	43	16	25	37	46	46	47
	Bending	1	3	11	19	25	29	1	2	13	22	26	30	4	7	23	35	42	43
	Bilinear	1	2	6	14	24	27	1	2	5	10	16	23	2	3	11	33	33	36
	Pincushion	1	8	29	37	40	40	10	13	27	36	42	44	32	36	45	44	48	50
	PLM	13	11	22	26	29	29	11	16	27	31	35	37	31	30	41	45	49	48
	Projective	1	3	10	19	23	29	2	4	16	21	30	34	3	8	19	36	39	45
	Rotation	1	1	6	14	16	19	2	2	8	14	18	25	1	3	25	34	40	45
	Scale	7	8	11	16	18	28	3	4	7	12	18	25	3	8	16	28	41	40
	Sinus	4	11	23	32	32	36	5	13	29	38	40	42	29	35	44	47	47	47
	Sinus+Bending	3	10	23	31	35	36	5	12	33	38	41	40	20	32	42	44	46	48
	Translation	10	11	17	23	26	30	5	4	8	11	20	20	5	6	19	25	34	35

Because mismatches were introduced randomly, each realization of mismatches for a given percentage is different. Therefore, the experiment was repeated six times for each distortion type. For reasons of limited available computational power, this experiment was only performed on the Lena image, using the same correspondence vectors that were used to generate the results listed in Table 4.2. To assist comparison with the result of the perfect match experiments used in Section 4.4.3.1., these results of this experiment are repeated in Table 4.2. As in the previous section, the images are not cropped after the distortions are applied.

4.5. Conclusions

In this chapter, we have presented two approaches to deal with the watermark synchronization problem due to the presence of geometric distortions. The first approach deals with this problem by removing the strict dependence on spatial synchronization between the watermark and the watermark detector. This approach has the following advantages over classic noise-based schemes:

- Invariant to translations.
- Higher robustness against rotation and scaling.

However, some aspects of the proposed method still have to be improved. In the first place, the performance of the detector should be optimized further. As shown in our experiments, this is critical to the performance of the system. Secondly, the proposed scheme suffers from some security issues. Our experiments suggest that relying solely on the key used to generate P_0 and P_1 may not be enough to prevent an attacker from using simple detectors (e.g., a high-pass filter or randomly generated P_1) to get some indication of the areas where the watermark patch has been embedded.

The second approach discussed in this chapter deals with the synchronization problem by inverting the geometric distortion and thus recovering the watermark synchronization. From the experiment results shown in Section 4.4.3, we can draw the following conclusions:

- There is a large improvement of the BER for distortions that actually are a polynomial transform. For these transformation classes the achieved BERs drop below or around 5% for all cases.
- For some extremely non-linear distortions, a large improvement in BER performance is achievable. Depending on the correspondence vectors used, large improvements can be made for most highly non-linear distortion types. However, the BERs increase rapidly as the available corresponding points decrease.
- Uniformly distributed feature points (with pixel accuracy) may give less improvement in BER performance than non-uniform distribution. It is most significant when the image was translated by ½ pixel.
- The use of orthogonal polynomials makes the proposed approach quite sensitive to the presence of mismatches. Even in the case where the applied geometrical distortion actually is a polynomial, performance goes down rapidly with an increasing percentage of mismatches. This can be seen in Table 4.2. In particular, the presence of mismatches degrades the performance of the scheme when the number of corresponding points are low or when the geometric distortion is highly non-linear.

4.6. Acknowledgement

The author is deeply indebted to P.J.O. Doets who has generously given permission for his work, as published in [12] and [16], to be included in Sections 4.4 and 4.5 of this chapter.

4.7. References

1. F.A.P. Petitcolas, R.J. Anderson, M.G. Kuhn, *Information Hiding – A Survey*, in Proceedings of the IEEE, Special Issue: Identification & Protection of Multimedia Information, pp. 1062-1078, July 1999
2. D. Delannay, J.-F. Delaigle, B. Macq, *Compensation of Geometrical Deformations for Watermark Extraction in the Digital Cinema Application*, in Proceedings of SPIE, Security and Watermarking of Multimedia Contents III, Vol. 4314, pp. 149-157, San Jose, CA, 2001
3. G.C. Langelaar, I. Setyawan, R.L. Lagendijk, *Watermarking Digital Image and Video Data: A State-of-the-Art Overview*, IEEE Signal Processing Magazine, Vol. 17, No. 5, pp. 20-46, September 2000
4. J.J.K. Ó Ruanaidh, T. Pun, *Rotation, scale and translation invariant digital image watermarking*, in Proceedings of IEEE, ICIP 1997, Vol. I, pp. 536 – 539, Santa Barbara, CA, 1997
5. P-C Su, C.-C. J. Kuo, *Synchronized Detection of the Block-based Watermark with Invisible Grid Embedding*, in Proceedings of SPIE, Security and Watermarking of Multimedia Contents III, Vol. 4314, pp. 406 – 417, San Jose, CA, 2001
6. M.U. Celik, E.S. Saber, G. Sharma, A.M. Tekalp, *Analysis of Feature-based Geometry Invariant Watermarking*, in Proceedings of SPIE, Security and Watermarking of Multimedia Contents III, Vol. 4314, pp. 261 – 268, San Jose, CA, 2001
7. G.W. Braudaway, F.C. Mintzer, *Automatic Recovery of Invisible Image Watermarks from Geometrically Distorted Images*, in Proceedings of SPIE, Security and Watermarking of Multimedia Contents II, Vol. 3971, pp. 74 – 81, San Jose, CA 2000
8. J. Lichtenauer, I. Setyawan, R.L. Lagendijk, T. Kalker, *Exhaustive Geometrical Search and the False Positive Watermark Detection Probability,* in the Proceedings of SPIE, Security and Watermarking of Multimedia Contents V, Vol. 5020, pp. 203 – 214, Santa Clara, CA, January 2003
9. S. Voloshynovskiy, F. Deguillaume and T. Pun, *Multibit digital watermarking robust against local nonlinear geometrical distortions*, in the Proceedings of IEEE, ICIP 2001, Vol. III, pp. 999 – 1002, 2001
10. P. Loo, and N. Kingsbury, *Motion estimation based registration of geometrically distorted images for watermark recovery*, Proceedings of SPIE, Security and Watermarking of Multimedia Contents III*,* Vol. 4314, pp. 606 – 617, San Jose, CA, 2001
11. A. Goshtasby, *Image registration by local approximation methods*, in Image and Vision Computing, Vol. 6, No. 4, pp. 255 – 261, November 1988
12. P.J.O. Doets, *Complexity-scalable compensation of geometrical distortions in image watermarking*, MSc. Thesis, Delft University of Technology, April 2003. Downloadable from http://www-ict.its.tudelft.nl/~inald/

13. P. Bas, J.-M. Chassery and B. Macq, *Geometrically invariant watermarking using feature points*, IEEE Trans. on Image Proc., Vol. 11, No. 9, pp. 1014 – 1028, September 2002.
14. C. Schmid, R. Mohr, and C. Bauckhage, *Comparing and evaluating interest points*, Proceedings of the 6th Int. Conf. on Computer Vision, pp. 230 – 235, Bombay, 1998.
15. C. Harris and M. Stephens, *A combined corner and edge detector*, 4th Alvey Vision Conf., pp. 147 – 151, 1988.
16. P.J.O. Doets, I. Setyawan, R.L. Lagendijk, *Complexity-Scalable Compensation of Geometrical Distortions in Image Watermarking*, in the Proceedings of IEEE, ICIP 2003, Vol. I, pp. 513 – 516, Barcelona, 2003
17. D. Delannay, B. Macq, *Method for hiding synchronization marks in scale and rotation resilient watermarking schemes*, in Proceedings of SPIE, Security and Watermarking of Multimedia Contents IV, Vol. 4675, pp. 520 – 529, San Jose, CA, 2002
18. J. Lichtenauer, I. Setyawan, R.L. Lagendijk, *Hiding correlation-based watermark templates using secret modulation*, to appear in Proceedings of SPIE, Security and Watermarking of Multimedia Contents VI, Vol. 5306, San Jose, CA, 2004

Chapter 5
OBJECTIVE QUALITY MEASUREMENT OF GEOMETRICALLY DISTORTED IMAGES

5.1. Introduction

Geometric distortion has always been a problem in the development of watermarking systems. This distortion happens when the watermarked data undergoes a geometric operation. This can happen due to various reasons, but basically geometric distortion occurs either due to the explicit application of geometric transformations or as a by-product of other processes (or attacks). Explicit application of geometric transformation includes non-malicious operations performed by a user, for example resizing of an image to fit one's desktop, and malicious operations for example application of random bending to an image using tools such as StirMark [1]. Examples of processes or attacks that produce geometric distortion as a by-product are the distortions incurred during the printing and scanning process [2] (due to the imperfections of the printer and/or scanner) or the distortions in video frames captured using a hand-held camera in a theatre in the digital cinema scenario [3] (due to the position of the camera, lens distortions, etc.).

We can also classify geometric distortion based on its locality. In this respect, geometric distortions can be classified as either global or local. In global geometric distortions, the underlying geometric transformation describing the geometric distortion applied to the whole image can be described using a single analytical expression and a single set of parameters associated with the expression. In local geometric distortions, the underlying geometric transformation uses different analytical expressions and/or different parameter sets for each part of the image.

The work presented in this chapter has been published in *Perceptual quality evaluation of geometrically distorted images using relevant geometric transformation modeling*, by I. Setyawan, D. Delannay, B. Macq and R.L. Lagendijk, in the Proc. of SPIE, Security and Watermarking of Multimedia Contents V, Vol. 5020, Santa Clara, CA, 2003

There are two aspects of geometric distortion that are of interest for the watermarking community, namely:

1. **The watermark de-synchronizing aspect.** Geometric distortion poses a problem for watermarking systems because it can de-synchronize the watermark detector, making the watermark undetectable. A lot of research effort has been performed in this area within the watermarking community. The research effort focusses on three approaches to dealing with this problem. The first approach is designing watermarking schemes that are invariant or insensitive to robust against geometric distortion [4,5]. The second approach involves research on methods (that is independent of the watermark detection) to invert the geometric distortion [6,7,10]. Finally, the third approach is to embed a synchronization signal in the watermark itself, to facilitate re-synchronization of the watermark by the embedder in the event of geometric distortion [8].

2. **The visual quality degradation aspect.** Geometric distortion degrades the visual quality of the watermarked data. Like all other distortions that affect watermarking systems, distortions due to geometric transformation are also bounded by the maximum visual quality degradation it can incur before the distorted image loses any commercial value. It is therefore important to be able to measure such distortion. The result of such measurement can be fed back into the design process of watermarking systems robust against geometric distortions. This aspect of geometric distortion has not been widely discussed in the literature. As a result, we are currently lacking an objective measure to quantify such distortion. Existing objective visual quality assessment tools, for example PSNR, are not suitable to be used to quantify visual quality due to geometric distortions because they rely on the pixel-per-pixel relationship between the original and the distorted images. An image distorted by geometric transformation loses most, if not all, such relationship to the original image. Measuring a geometrically distorted image using PSNR would, therefore, yield no meaningful result.

In this chapter we address the second aspect of the geometric distortion problem for watermarking systems. We propose a new visual quality measurement method suitable for this class of image distortion. Our approach is based on our previous work [9]. In this paper we limit ourselves to the visual quality measurement of global geometric distortions on still images. This chapter is organized as follows. In Section 5.2, we will present the underlying hypothesis on which our proposed method is based. In Section 5.3, we will present how we test the hypothesis and quantify the geometric distortion applied to an image. In Section 5.4, we will present the test setup we used to test our proposed method, as well as some preliminary results. Finally, in Section 5.5, we will present our conclusions and an outlook for further research.

5.2. The underlying hypothesis

5.2.1. Modeling global geometric transformation

The number of possible geometric transformations that can be applied to an image is essentially limitless. The possibility ranges from simple transformations to more complex ones. An example of geometric transformations is the RST (rotation, scaling and translation) transform described by the following equation:

$$\begin{pmatrix} u \\ v \end{pmatrix} = S\begin{pmatrix} \cos R & -\sin R \\ \sin R & \cos R \end{pmatrix}\begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} T_x \\ T_y \end{pmatrix} \tag{5.1}$$

Alternatively, an example of more complex geometric transformations is the bilinear transform described by the following equation:

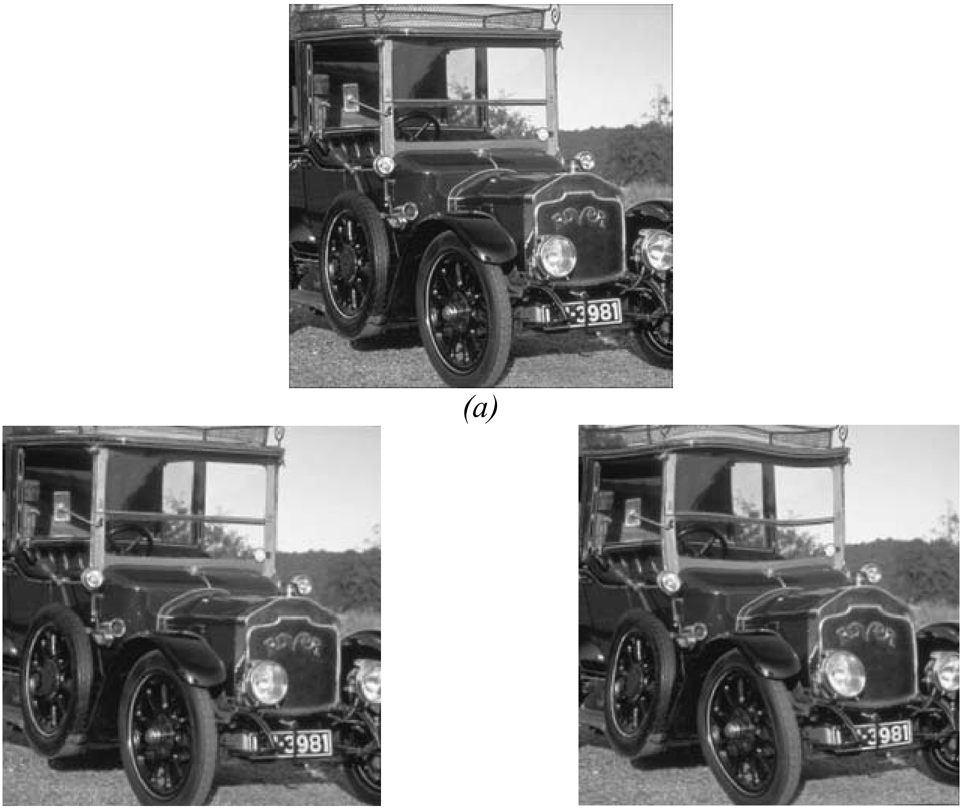
$$\begin{pmatrix} u \\ v \end{pmatrix} = \begin{pmatrix} a & c \\ b & d \end{pmatrix}\begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} e \\ f \end{pmatrix} xy + \begin{pmatrix} g \\ h \end{pmatrix} \tag{5.2}$$

Due to the vast number of possible geometric transforms applied to the image, it is impossible to model each of them individually. There are some approaches that can be used to solve this problem. One approach is to use simpler transformation models, for example RST or affine transform, to approximate the underlying complex, global geometric transform [8]. The approach is based on the assumption that a complex geometric transformation applied on a global scale can be approximated by a simpler transformation model applied on a more local scale. Another possible approach is to use orthogonal polynomials to do the approximation [10]. In this chapter, we use local RST transform to approximate the global underlying transform.

5.2.2. The hypothesis

At this point, we would like to present our definition of the *homogeneity* of a global geometric distortion, as follows: *A distortion is said to be homogenous if the underlying global transform can be approximated by one RST or affine transform with one set of parameters associated with it.* The reader should note that from this definition we make a distinction between *global* and *homogenous* distortions. The first term refers to the locality with which we apply the underlying geometric transformation, while the second term refers to the locality of the approximation of the underlying global transformation using RST or affine transforms. In other words, nonhomogenous distortions must be approximated by multiple local RST/affine geometric transforms. These local transforms have parameters that are varying from one part of the image to the other.

The following figure presents an original image, along with two distorted versions of the image. The first distorted version (Figure 5.1(b)) is the result of rotating the original image by 3 degrees followed by cropping and rescaling. The second distorted version (Figure 5.1(c)) is the result of applying a sinusoid-based transform to the original image.



(a)

(b) *(c)*

Figure 5.1. Example of geometrically distorted images.

From the visual quality point of view, it is easy to see that the first distortion is less disturbing compared to the second distortion. From the distortion homogeneity point of view, the first distortion can be classified as homogenous, since it can be approximated by one RST transform and its corresponding parameter set. The second distortion is not homogenous, because this distortion has to be approximated by multiple local RST transforms with parameter sets that are varying from one part of the image to the other.

Based on these observations, we propose the following hypothesis regarding the visual quality of geometrically distorted images: *The visual* *quality of an image distorted by a global geometric distortion is determined by the degree of homogeneity of the geometric distortion. The less homogenous the distortion, the worse the visual quality would be.* Furthermore, it is obvious that the severity of the geometric distortion itself also determines the overall visual quality.

We have searched the literature to find supporting evidence for, or arguments against, this hypothesis. The literature we considered includes topics in digital watermarking, computer vision, computer graphics, image coding and medical sciences. However, so far we have not been able to find any related work on the human perceptual quality assessment for geometrically distorted images.

5.3. Measuring distortion homogeneity

In order to be able to measure visual quality according to our hypothesis, we need to be able to measure distortion homogeneity. To measure homogeneity, we use the basic idea we presented in our previous work [9]. Basically, we measure distortion homogeneity by measuring the locality of the simple geometric transformation used to approximate the global transform.

5.3.1. Distortion locality approximation

There are two approaches that we can use to find the parameters of the transformation which best approximate the global distortion. The first approach is performed using the analytical description of the underlying global distortion, while the second approach uses the original and distorted images directly. In both approaches, we first perform the approximation on a global scale and then, if necessary, increase the locality of the approximation to achieve the final result.

5.3.1.1. Approximation using analytical description

In this approach, we assume that we know the analytical description of the function *D(•)* that transforms the original image *I* into the distorted image *I'*. Therefore, the registration process can rely on the exact displacement vector of every pixel position in the image. Considering a field of displacement vectors for a given region of the image, the parameters of the simple geometric transformation can be computed using a least square error optimization. The registration criterion consists of the mean error ε of the resulting approximation.

Let (x_i,y_i) be a set of original coordinates and (u_i,v_i) be the corresponding set of coordinates transformed by the function *D(•)*. The least square error optimization consists of finding the set of transform parameters $(p_1, p_2, .., p_n)$ that minimizes the cost function ε. Let *F(•)* be the simple geometric transformation function used to approximate the global geometric distortion.

This function transforms the original coordinates (x_i, y_i) to the corresponding coordinates (x_i', y_i'),

$$\begin{pmatrix} x'_i \\ y'_i \end{pmatrix} = F(\mathbf{p}, x_i, y_i), \tag{5.3}$$

where

$$\mathbf{p} = \begin{pmatrix} p_1 \\ p_2 \\ \ldots \\ p_n \end{pmatrix}, \tag{5.4}$$

The cost function to be minimized can then be expressed as follows:

$$\min_{\mathbf{p}} \varepsilon = \min_{\mathbf{p}} \sum_i b_i \left\| \begin{pmatrix} u_i \\ v_i \end{pmatrix} - F(\mathbf{p}, x_i, y_i) \right\|^2 \tag{5.5}$$

where parameter b_i is a weighting factor. When the simple geometric transformation *F(•)* is the RST model, this optimization yields a linear system whose solution can be found in [9].

5.3.1.2. Approximation using the original and distorted images directly

In this approach, we do not assume knowledge of the underlying function describing the global distortion. Instead, only the original (*I*) and the distorted (*I'*) images are available. We apply the simple geometric transformation to the original image *I* to produce an intermediate image *I''*. The parameters of this simple geometric transform are taken within a certain range of parameters. There are some strategies that can be used to search the parameters within this set, for example exhaustive search, gradient search or coarse-to-fine search.

The next step is to compute the approximation error based on pixel value (e.g., luminance or color) comparison. The approximation error ε is computed between *I''* and *I'* as follows:

$$\varepsilon = \left(I'' - I'\right)^2 \tag{5.6}$$

where *I''* and *I'* refers to the luminance value of the intermediate and distorted images (or local areas of those images), respectively. The error measurement in Equation (5.6) is valid if we assume that only geometric distortion has occurred and there are no luminance changes (e.g., brightness or contrast changes) between the original and the distorted images.

5.3.2. Comparison of the two approximation approaches

The advantage of the first approach is that it does not involve actual images and the computationally expensive operations associated with them. This approach only compares the *pixel position* and is therefore faster. Furthermore, it enables precise characterization of a known distortion model. The second approach operates directly on the images. In other words, this approach compares actual *pixel values* (e.g., luminance) and is therefore computationally expensive. However, since the second approach deals directly with the image content, it has some advantages when we are trying to assess the quality of the distorted image. Furthermore, this approach can be used in scenarios where the analytical description underlying geometric distortion is not known.

The second approach, as described above, is more sensitive in areas with texture/structure than in flat areas. As a result, the locality of the approximation will be less accurate in flat areas. In other words, the locality of the approximation in flat areas is less in areas with structure, even if both areas experience the same geometric distortion. Since we base our distortion measurement on the locality of this approximation, this means that in this case the flat area will be declared to have less distortion than the area with structure. This property can be seen as an advantage of the second approximation approach over the first approach, because a human observer will also be less likely to notice distortion in flat areas. Using the first approach, every part of the image experiencing the same distortion would yield the same approximation. This may result in a measurement that does not correspond to human perception. However, if one wants to characterize the distortions occurring in a particular system, it might be advantageous not to depend on a specific content in order to measure the average (or worst case) degradation that the system introduces.

In this work, we chose to use the second approach to perform the test on our hypothesis. Nevertheless, similar (although content-independent) results could be obtained using the first approach.

5.3.3. The proposed method to measure distortion homogeneity

The proposed methodology proceeds by iterative computations of the approximation error over progressively increasing approximation locality. This operation is repeated until either the approximation error is lower than a predetermined threshold or the locality of the transform reaches a predetermined level. We use quadtree partitioning to increase the locality of the approximation. The first quadtree partitioning is performed on the whole image. Further quadtree partitioning in subsequent iterations is performed on any quadtree blocks in which the approximation error is still above the predetermined threshold. The block size of the quadtree structure is therefore dependent on the locality and accuracy of the approximation.

The proposed procedure is illustrated in Figure 5.2. The system has two inputs, namely the original (*I*) and the distorted (*I'*) images. Furthermore, there are three parameters that control the system. The parameters are the minimum block size B_{min}, the maximum error threshold θ and the parameter set range *P*. The first parameter, B_{min}, controls the precision of the locality approximation of the global geometric distortion. The choice of this parameter is a trade-off between the precision and the reliability of the approximation, as block sizes that are too small will make the approximation less reliable. The error threshold θ controls the precision of the approximation and must be traded-off with computation time. Finally, the choice of parameter set range (and the precision of its step size) in *P* controls the accuracy of the approximation. This parameter has the biggest influence on the computation time needed for the procedure, so one has to trade-off accuracy and computation time. As mentioned in Section 5.3.1.2, there are some strategies that can be used to search for the correct parameters within the parameter set range *P*. For simplicity, we chose to do an exhaustive search in our experiments.

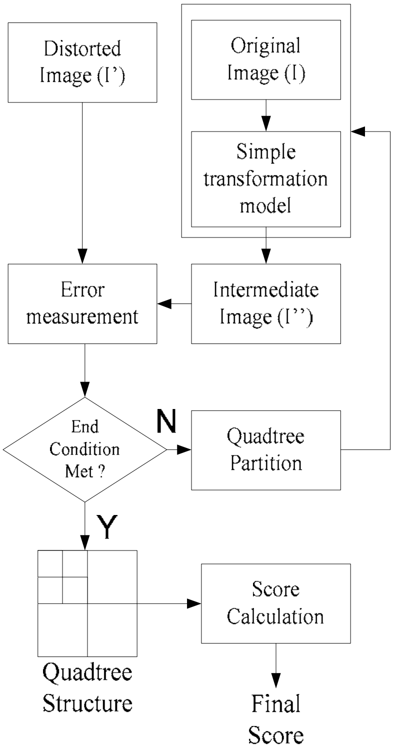


Figure 5.2. The procedure used to measure distortion homogeneity

The procedure goes as follows. In the first iteration, we try to approximate the global underlying geometric distortion with one global RST transform and compute the approximation error ε. This approximation error is then compared to θ. If the minimum ε obtained in this process is larger than θ, we go to the second iteration and increase the locality of the approximation by performing a quadtree partitioning to both I and I'. Then we repeat the process described above to each corresponding block of the quadtree. In the subsequent iterations, we perform further quadtree partitioning for any blocks in which the minimum ε is larger than θ. The iterations are continued until the end condition is met. In our case, this means that all blocks already have $\varepsilon < \theta$, B_{min} is reached or both. The result of this procedure is a quadtree partition structure showing the locality of the RST transform approximation. Examples of such a structure are shown in Figure 5.3. In this example, we set the minimal block size to be 32 pixels. Furthermore, the parameter set of the global distortion applied to Figure 5.3(a) is chosen to be more severe than the one applied to Figure 5.3(b). Here we can see that the image with the larger distortion is more finely partitioned than the one with less distortion.

In order to obtain the final numerical score that will indicate the visual quality of the distorted image, we need to be able to quantify this quadtree structure. There are some possibilities to do so, including evaluating the average block size, the variance of the block size or the variance of the parameter sets associated to each block in the quadtree structure. In our experiments, we chose to use the average block size to quantify this structure, with blocks that already reach B_{min} but with approximation error $\varepsilon > \theta$ being given a special weighting factor. The final score is computed using the following equation

$$S = \frac{B}{N} \tag{5.7}$$

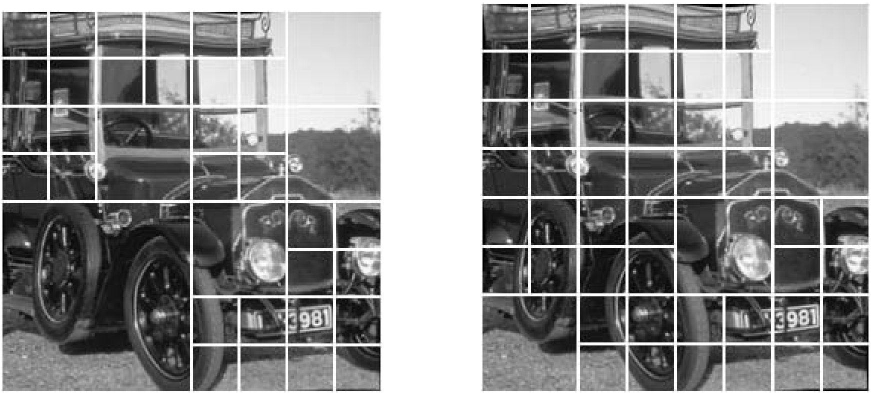
where

$$\begin{gathered} B = \sum_{i=1}^{N^o} B_i^o + \sum_{j=1}^{N^w} \frac{B_j^w}{2} \\ N = N^o + 4N^w \end{gathered} \tag{5.8}$$

and

S = *Final score*
B = *Total block size*
B^o = *block size of blocks where final* $\varepsilon < \theta$
B^w = *block size of blocks where final* $\varepsilon > \theta$
N = *total number of blocks*
N^o = *total number of blocks where final* $\varepsilon < \theta$
N^w = *total number of blocks where final* $\varepsilon > \theta$

As a final note, we would like to point out that the quadtree structure examples in Figure 5.3 show how the image content influences the measurement procedure, as already pointed out in Section 5.3.2. In this example we can see that areas with a lot of texture or structure are more accurately approximated and finely partitioned compared to the flat areas or areas with less details although they undergo similar distortion. As a consequence, flat areas are given higher scores than more detailed areas.

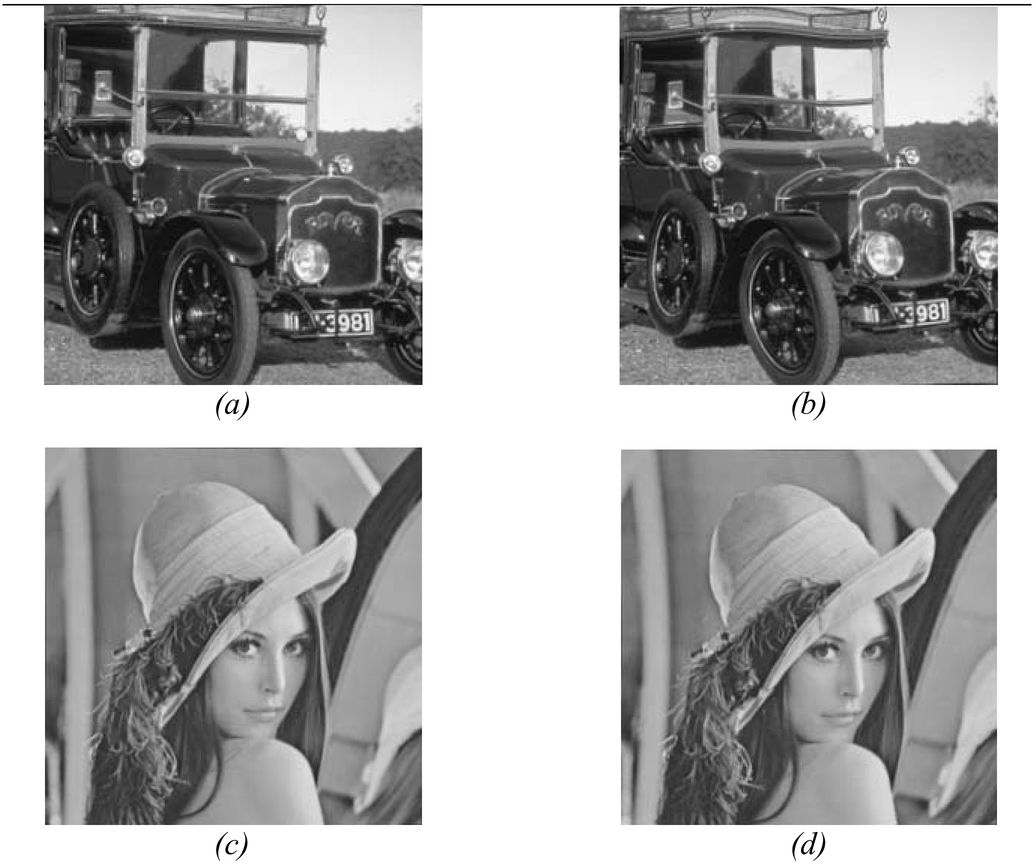


(a) *(b)*

Figure 5.3. Examples of the quadtree structure

5.4. Test setup and results

In order to test our hypothesis, we performed both objective and subjective tests. In the objective test, we performed the measurement procedure described in Section 5.3.3. The purpose of the subjective test is to give an indication of the correlation of the measurement score obtained by the objective test to human perception of the visual distortion. The objective test is performed on 8-bit grayscale images (256 × 256 pixels). The distortions applied to the test images are rotation, bending and swirl. For each of the last two distortions, two different parameter sets were used, denoted with the numbers 1 and 2 (e.g., we have Swirl 1 and Swirl 2). The second parameter set was chosen so that the transformation using this set will give more severe distortion to the image compared to the transformation performed using the first set. Parameter B_{min} is set at 32 pixels. The threshold θ is determined experimentally and is chosen so that the measurement has enough approximation precision as well as having the ability to differentiate among the different distorted images. The range of parameter set P is as follows: rotation from -5 to 5 degrees, translation of maximum 10 pixels in both x and y directions, and scaling from 80% to 120% of the original image (or image block) size. From Equations (5.7) and (5.8), we can see that the maximum score that can be achieved is 256, while the minimum score would be 16. Some example images, along with their scores, are shown in Figure 5.4. The objective test scores along with the subjective test result for the test images are presented in Table 5.1.



(a) (b)

(c) (d)

Figure 5.4. Examples of the measurement result.
(a) Rotated image, score=256, (b) Swirl distortion, score=21.97,
(c) Bending distortion, parameter set 1, score=21.69,
(d) Bending distortion, parameter set 2, score=20.08

The subjective test is performed as follows. The test subjects are asked to look at five sets of images that are also used in the objective test. Each set contains the distorted versions of one test image. They are then requested to rank the images from the same set, starting from the one that they find the most distorted. Thus, we did not ask the test subjects to compare images from different sets. Then we compare the ranking of the images with the scores obtained from the objective test. This is done to get an indication of the correlation of the scores and the user preference. The user score shown in Table 5.1 represents the average ranking given to a given distorted image by the test subjects. Thus, the lower score indicates less perceived distortion. As we can see from Table 5.1, the result of the subjective test can be summarized as follows. For images distorted using the same geometric distortion, but with different severity, the subjective test result is consistent with that of the objective test. In other words, the test subject prefers the image with the higher score to the one with the lower score. However, for images distorted using different geometric distortions, the subjective test result is as yet inconclusive.

Table 5.1. Objective and subjective test results

Image	Distortion	Objective test score	Subjective test score
F15C	Rotation	256	1
	Swirl 1	29.54	4.25
	Swirl 2	27.70	3.25
	Bending 1	23.56	3.25
	Bending 2	21.43	3.25
Baboon	Rotation	256	1
	Swirl 1	24.97	3.5
	Swirl 2	22.57	3.5
	Bending 1	20.93	3
	Bending 2	18.34	4
Oldcar	Rotation	256	1
	Swirl 1	21.97	3.5
	Swirl 2	21.42	5
	Bending 1	18.89	2.5
	Bending 2	17.91	3
MotoX	Rotation	256	1
	Swirl 1	21.42	3
	Swirl 2	20.08	4.25
	Bending 1	18.09	3
	Bending 2	17.14	4
Island	Rotation	256	1
	Swirl 1	256	3.25
	Swirl 2	37.16	4.25
	Bending 1	35.20	2.75
	Bending 2	26.09	4

5.5. Conclusions and future works

The conclusions we can draw based on the discussion in this chapter are as follows:

1. We have proposed a definition of *geometric distortion homogeneity*, based on the locality of the approximation of the underlying global geometric transformation using RST/affine transforms.

2. We have proposed an hypothesis of how to quantify a geometric distortion, based on its homogeneity.

3. We have proposed and tested a method to measure the perceptual quality of geometrically distorted images.

The proposed system is still a work in progress and currently there are still some limitations that should be addressed. The improvements of these limitations are the topics for our future works. In particular, we think that the following topics should be investigated more thoroughly:

1. Refinement of the procedure used to determine distortion homogeneity. In the first place, the other outputs of the approximation procedure, namely the parameter sets of the local transform and the approximation errors of each block, are also useful to measure distortion homogeneity. In this paper, we have not taken these into account. Secondly, as we can see in Table 5.1, the discriminating power of the objective test scores is fairly small. This could be due to the discriminating power of the equations we use to compute the final score being too small or due to the diversity in block sizes of the quadtree structure being too small. We would like to investigate the behavior of these factors and find a solution to this problem. Finally, we would like to look into other possible alternatives to the quadtree structure (for example, we want to look into the option to merge one or more blocks in the structure with similar distortion characteristics).

2. Take image content more into account, since human perception of geometric distortion is highly influenced by the presence of certain structures in the image. In our experiments, this aspect has been indirectly taken into account due to the fact that our distortion homogeneity measurement procedure is influenced by image content. However, we would like to look into ways to explicitly involve the image content in the final score calculation.

3. The subjective test described in this paper was intended to give a preliminary indication of the performance of the proposed method. In order to achieve a more reliable and representative result, we need to perform a more elaborate subjective test with more test images and test subjects. The design and implementation of such a test and the analysis of the test results, are discussed in the next chapter of this thesis.

5.6. References

1. F.A.P. Petitcolas, R.J. Anderson and M.G. Kuhn, *Attacks on copyright marking systems,* in Information Hiding: 2nd Int. Workshop (Lecture Notes in Computer Science), Vol. 1525, pp. 218 – 238, Berlin, Springer-Verlag, 1998
2. I.J. Cox, J. Kilian, T. Leighton and T. Shamoon, *Secure spread spectrum watermarking for images, audio and video*, in Proceedings of IEEE, ICIP 1996, pp. 243 – 246, Lausanne, 1996
3. J. Haitsma and T. Kalker, *A watermarking scheme for digital cinema*, in Proceedings of IEEE, ICIP 2001, pp. 487-489, Thessaloniki, 2001
4. J.J.K. Ó Ruanaidh and T. Pun, *Rotation, scale and translation invariant digital image watermarking*, in Proceedings of IEEE, ICIP 1997, pp. 536 – 539, Santa Barbara, CA, 1997
5. I. Setyawan, G. Kakes and R. L. Lagendijk, *Synchronization-insensitive video watermarking using structured noise pattern*, in Proceedings of SPIE, Security and Watermarking of Multimedia Contents IV, Vol. 4675, pp. 520 – 530, San Jose, CA, 2002
6. P. Loo and N. Kingsbury, *Motion-estimation-based registration of geometrically distorted image for watermark recovery*, in Proceedings of SPIE, Security and Watermarking of Multimedia Contents III*,* Vol. 4314, pp. 606 – 617, San Jose, CA, 2001
7. D. Delannay, J-F Delaigle, B. Macq and M. Barlaud, *Compensation of geometrical deformations for watermark extraction in the digital cinema application*, in Proceedings of SPIE, Security and Watermarking of Multimedia Contents III*,*Vol. 4314, pp. 149 – 157, San Jose, CA, 2001
8. F. Deguillaume, S. Voloshynovskiy and T. Pun, *A method for the estimation and recovering from general affine transforms in digital watermarking applications*, in Proceedings of SPIE, Security and Watermarking of Multimedia Contents IV*,* Vol. 4675, pp. 313 – 322, San Jose, CA, 2002
9. D. Delannay, I. Setyawan, R.L. Lagendijk and B. Macq, *Relevant modeling and comparison of geometric distortions in watermarking systems*, in Proceedings of SPIE, Application of Digital Image Processing XXV, Vol. 4790, pp. 200 – 210, Seattle, WA, 2002
10. P.J.O. Doets, I. Setyawan and R.L. Lagendijk, *Complexity scalable compensation of geometrical distortions in image watermarking*, in Proceedings of IEEE, ICIP 2003, Vol. I, pp. 513 – 516, Barcelona, 2003

Chapter 6
EVALUATING THE OBJECTIVE QUALITY MEASUREMENT METHOD

6.1. Introduction

Research on human perception of image quality has been widely performed. Aspects of images considered in such research are, for example, color, granularity or sharpness. Another example is to test specific artifacts of a compression algorithm (e.g., the blocking artifact of JPEG compression) or watermarking system (e.g., the random noise artifact of noise-based watermarking systems). Some examples of the image quality assessment for these distortions can be found in [1]. As a result, we already have a good understanding of how these aspects influence human perception of quality and we are able to quantify these perceptual aspects in cases where the distortion is near the visibility threshold. We can use the result, for example, to build a system to objectively measure image quality based on these aspects which corresponds quite well to subjective quality perception. We can also use the result of this research to improve the performance of various applications dealing with images by designing the systems such that most changes or distortions to the images occur in the areas that have small perceptual impact for human observers. The examples mentioned above, namely the compression algorithms and watermarking systems, are two examples of applications that can take advantage of this knowledge. However, the research on human perception of image quality has not dealt with another type of distortion that an image can undergo, namely geometric distortion (i.e., distortions due to geometric operations). As a result, we are currently unable to quantify the perceptual impact of geometric distortions on images.

This chapter presents a study of the impact of geometric distortions on human perception of the quality of the affected images. The aim of this study is to provide both a better understanding of human perception of geometric distortion and a reference point with which to evaluate the performance of our novel objective geometric distortion measure scheme described in Chapter 5.

The work presented in this chapter is an expanded version of the work published in *Human perception of geometric distortions in images* by I. Setyawan and R.L. Lagendijk, to appear in Proc. SPIE, Security, Steganography and Watermarking of Multimedia Contents VI, Vol. 5306, San Jose, CA, 2004

In order to perform this study we propose a user test system that is specifically designed to observe the impact of geometric distortion on human perception of image quality. The results we obtain from this test can also be useful to other researchers performing similar research in the fields of watermarking, image processing and human visual systems. Therefore, we have also made our test set and test results available for download on our website [3].

The rest of this chapter is organized as follows. In Section 6.2, we present the design of our user test experiment and statistical analysis methods used to process the test results. In Section 6.3, we present the actual setup of our user test. In Section 6.4, we present and analyze the result obtained from this user test. In Section 6.5, we will briefly review our objective geometric distortion measure algorithm, present scores obtained using this method and evaluate its performance based on the subjective test result and compare its performance with other possible objective perceptual quality measurement systems. Finally, in Section 6.6, we present our conclusions and provide an outlook for further research.

6.2. Test design & analysis method

In this section we shall discuss in more detail the test design and the analysis tools we use to analyze the test results. The test design and analysis tools we use are well known in the literature [4, 6]. They have been used, for example, in experiments to determine consumer preference to certain products or product variants (e.g., different flavors of food) [4]. However, their usage in evaluating perceptual impact of geometric distortions in images, to the best of the author's knowledge, is novel and has never been discussed in the literature.

6.2.1. Test design

In order to evaluate the perceptual impact of geometric distortion, we performed a subjective test involving a panel of users, who are asked to evaluate a test set comprised of an original image and various distorted versions of it. The test subjects evaluate one pair of images at a time, comparing 2 images and choosing the one they think is more distorted. This type of experiment is called the paired comparison test. There are two experiment designs for a paired comparison test, namely the *balanced* and *incomplete* designs [4, 5]. In a balanced design, a test subject has to evaluate all possible comparison pairs taken from the test set. In the incomplete design, a test subject only has to perform comparisons of part of the complete test set. The latter design is useful when the number of objects in the test set is very large. In our experiment, we used the balanced paired-comparison design. Our choice for this design is based on three factors. Firstly, the number of objects in our test set is not very large and a test subject can finish the test within a reasonable time frame (as a rule of thumb, we consider a test lasting 60 minutes or less to be reasonable). Secondly, by asking every test subject to evaluate all objects in the test set we will be able to get a more complete picture of the perceptual quality of the images in the test set. Finally, in this design we make sure that each test subject evaluates an identical test set. This makes it easier to evaluate and compare the performance of each test subject.

Let t be the number of objects in the test set. One test subject performing all possible comparisons of 2 objects A_i and A_j from the test set, evaluating each pair once, will make $\binom{t}{2}$ paired comparisons in total. The result of the comparisons is usually presented in a $t \times t$ matrix. If ties are not allowed (i.e., a test subject must cast his/her vote for one object of the pair), the matrix is also called a *two-way preference matrix* with entries containing 1's if the object was chosen and 0's otherwise. An example of such a matrix for $t = 4$ is shown in Figure 6.1. Each entry $A_{i,j}$ of the matrix is interpreted as *object A_i is preferred to object A_j*. The indices i and j refer to the rows and columns of the matrix, respectively.

	A_1	***A_2***	***A_3***	***A_4***
A_1	×	1	1	**0**
A_2	0	×	1	**1**
A_3	0	0	×	**0**
A_4	**1**	**0**	**1**	×

Figure 6.1. An example of a preference matrix

Let a_i be the number of votes object A_i received during the test. In other words, $a_i = \sum_{\substack{j=1, \\ i \neq j}}^{t} A_{i,j}$. We call a_i the *score* of object A_i. It is easy to see that the total score for all objects is

$$\sum_{i=1}^{t} a_i = \frac{1}{2} t(t-1) \qquad (6.1)$$

and that the average score among all objects is

$$\bar{a} = \frac{\sum_{i=1}^{t} a_i}{t} = \frac{1}{2}(t-1) \qquad (6.2)$$

We can extend these results to the case where we have n test subjects performing the paired comparison test. In this case, the test result can also be presented in a preference matrix similar to the one presented in Figure 6.1. However, each entry $A_{i,j}$ of this matrix now contains the number of test subjects who prefer object A_i to object A_j. If again we do not allow ties, the values of $A_{i,j}$

will be integers ranging from 0 to n. We also note that in this case $A_{j,I} = n - A_{i,j}$. Finally, in this case, the total and average scores are expressed as $\frac{1}{2}nt(t-1)$ and $\frac{1}{2}n(t-1)$, respectively.

6.2.2. Statistical analysis of the experiment

After performing paired comparison tests, we obtain a preference matrix for each test set. Now we have to perform an analysis of this test result. We have two main objectives for this analysis. In the first place, we want to obtain the overall ranking of the test objects. The second objective is to see the relative quality differences between the test objects, that is, whether object A_i is perceived to be either similar to or very different in quality from object A_j. The analyses we perform on the data to achieve these objectives are the *coefficient of consistency*, the *coefficent of agreement* and the *significance test on score differences*. Each of these analyses is discussed in the following sections.

6.2.2.1. Coefficient of consistency

A test subject is consistent when he/she, in evaluating three objects A_x, A_y and A_z from the test set, does not make a choice such that $A_x \rightarrow A_y \rightarrow A_z$ but $A_z \rightarrow A_x$. The arrows can be interpreted as “preferred to”. Such a condition is called a *circular triad*. While circles involving more than three objects are also possible, any such circles can easily be broken up into two or more circular triads. The preference matrix presented in Figure 6.1 has one such triad, namely $A_1 \rightarrow A_2 \rightarrow A_4$ but $A_4 \rightarrow A_1$.

For smaller values of t, one can easily enumerate the circular triads encountered. For larger t, this task becomes very tedious. However, we can compute the number of circular triads, c, from the scores a_i using the following relation [4,6]:

$$c = \frac{t}{24}(t^2 - 1) - \frac{T}{2} \tag{6.3}$$

where

$$T = \sum_{i=1}^{t} (a_i - \overline{a})^2 \tag{6.4}$$

The number of circular triads c can be used to define a measure of consistency of the test subjects. There are different approaches to do this [4]. Kendall/Babington-Smith compared the number of circular triads found in the test to the maximum possible number of circular triads. The coefficient of consistency ζ is defined as follows:

$$\zeta = 1 - \frac{24c}{t(t^2 - 1)}, \text{ if } t \text{ odd} \tag{6.5a}$$

$$\zeta = 1 - \frac{24c}{t(t^2 - 4)}, \text{ if } t \text{ even} \tag{6.5b}$$

There are no inconsistencies if, and only if, $\zeta = 1$. This number will move to zero as the number of circular triads, thus the inconsistencies, increases.

The coefficient of consistency can be used in the following ways. In the first place, we can use this coefficient to judge the quality of the test subject. Secondly, we can use this coefficient as an indication of the similarity of the test objects. If, on average, the test *subjects* are inconsistent (either for the whole data set or a subset thereof), we can conclude that the test *objects* being evaluated are very similar and thus it is difficult to make a consistent judgement. Otherwise, if one particular test *subject* is inconsistent while the other test subjects are – on average – consistent, we may conclude that this particular subject is not performing well. If the consistency of this subject is significantly lower than average, we may consider removing the result obtained by this subject from further analysis.

6.2.2.2. Coefficient of agreement

The coefficient of agreement shows us the diversity of preferences among n test subjects. Complete agreement is reached when all n test subjects make identical choices during the test. From Section 6.2.1, we see that if every subject had made the same choice during the test (in other words, if there has been complete agreement), then half of the entries in the preference matrix will be equal to n, while the other half would be zero. Alternatively, in the worst case situation, all entries will be equal to $n/2$ (if n is even) or $(n \pm 1)/2$ if n is odd.

It is obvious that the minimum number of test subjects, n, that we need in order to be able to measure agreement is 2. Each time 2 test subjects make the same decision regarding a pair of test objects A_i and A_j, we say that we have one agreement regarding this pair. In other words, we measure the agreement by counting the number of pairs of test subjects that make the same decision about each pair of test objects. We do this by computing τ, defined as

$$\tau = \sum_{i=1}^{n} \sum_{j=1}^{n} \binom{A_{ij}}{2}, \qquad i \neq j \tag{6.6}$$

In Equation (6.6), $\begin{pmatrix} A_{ij} \\ 2 \end{pmatrix}$ gives us the number of pairs of test subjects making the same choice regarding objects A_i and A_j. Thus τ gives us the total number of agreements among n test subjects evaluating t objects. Obviously, when $A_{i,j}$ = 1 we do not have any agreement among the subjects and the contribution of this particular $A_{i,j}$ to τ would be zero. If $A_{i,j}$ = 0, it means that all test subjects agree *not* to choose A_i over A_j. Although the contribution of this $A_{i,j}$ to τ is also zero, the number of agreements regarding this pair of test objects will be reflected by the value of $A_{j,i}$.

We have $\begin{pmatrix} t \\ 2 \end{pmatrix}$ pairs of comparisons and $\begin{pmatrix} n \\ 2 \end{pmatrix}$ possible pairs of subjects, therefore the maximum number of agreements between the subjects is given by

$$\max(\tau) = \begin{pmatrix} t \\ 2 \end{pmatrix}\begin{pmatrix} n \\ 2 \end{pmatrix} \tag{6.7}$$

Meanwhile, the minimum value of τ is given by

$$\min(\tau) = \begin{pmatrix} t \\ 2 \end{pmatrix}\begin{pmatrix} \lfloor n/2 \rfloor \\ 2 \end{pmatrix} \tag{6.8}$$

We can also express τ in a more computationally convenient way, as follows.

$$\tau = \frac{1}{2}\left[\sum_{i \neq j} \alpha_{i,j}^2 - n\begin{pmatrix} t \\ 2 \end{pmatrix}\right] \tag{6.9}$$

Kendall/Babington-Smith [6] defines the coefficient of agreement, u, as follows

$$u = \frac{2\tau}{\max(\tau)} - 1 = \frac{2\tau}{\begin{pmatrix} t \\ 2 \end{pmatrix}\begin{pmatrix} n \\ 2 \end{pmatrix}} - 1 \tag{6.10}$$

The value of u = 1 if and only if there is complete agreement among the test subjects, and it decreases when there is less agreement among the test subjects. The minimum value of u is -1/(n-1) if n is even or -1/n if n is odd. The lowest possible value of u is -1 which can only be achieved when n = 2. This value of u shows the strongest form of disagreement between the test subjects, namely that the test subjects completely contradict each other.

We can perform a hypothesis test to test the significance of the value u. The null hypothesis is that all test subjects cast their preference completely at random. The alternative hypothesis is that the value of *u* is greater than what one would expect if the choices would have been made completely at random. To test the significance of *u* we use the following statistic, as proposed in [4]

$$X^2 = \frac{4}{n-2}\left[\tau - \frac{1}{2}\binom{t}{2}\binom{n}{2}\frac{(n-3)}{(n-2)}\right] \tag{6.11}$$

which has χ^2 distribution with $\binom{t}{2}\frac{n(n-1)}{(n-2)^2}$ degrees of freedom.

As *n* increases, the expression in Equation (6.11) reduces to a simpler form [7]

$$X^2 = \binom{t}{2}[1 + u(n-1)] \tag{6.12}$$

with $\binom{t}{2}$ degrees of freedom.

It is important to note that consistency and agreement are two different concepts. Therefore, a high *u* value does not necessarily imply the absence of inconsistencies and vice versa.

The coefficient of agreement also shows whether the test objects, on average, received equal preference from the test subjects. If the overall coefficient of agreement is very low we can expect that the score of each test object will be very close to the average scores of all test objects, i.e., there is no significant difference among the scores. As a consequence, assigning ranks to the objects or drawing the conclusion that one object is better (or worse) than the others is pointless since the observed score differences (if any) cannot be used to support the conclusion. On the other hand, strong agreement among the test subjects indicates that there exist significant differences among the scores.

6.2.2.3. Significance test of the score difference

A significance test of the score difference is performed in order to see whether the perceptual quality of any 2 objects from the test set is perceived as different. In other words, the perceptual quality of object A_i is declared to be different from the quality of object A_j, only if a_i is significantly different from a_j. Otherwise, we have to conclude that the test subjects consider the perceptual quality of the 2 objects to be similar.

This problem is equivalent to the problem of dividing the obtained set of scores $S = \{a_1, a_2, \ldots, a_t\}$ into sub-groups such that the variance-normalized *range* (the difference of the largest and lowest values) of the scores within each group,

$$R = \frac{(a_{\max} - a_{\min})}{\sigma_{a_i}} \tag{6.13}$$

is lower or equal to a certain value $\lceil R_c \rceil$ (in other words, the difference of any 2 scores within the group must be lower or equal to $\lceil R_c \rceil$), which depends on the value of the significance level α. In other words, we want to find R_c such that the probability $P[R \geq R_c]$ is lower or equal to the chosen significance level α. We declare the objects within each group to be not significantly different, while those from different groups are declared to be significantly different. By adjusting the value of α, we can adjust the size of the groups. This in turn controls the probability of false positives (declaring 2 objects to be significantly different when they are not) and false negatives. The larger the groups, the higher the probability of false negatives. On the other hand, the smaller the groups, the higher the probability of false positives.

The distribution of the range R is asymptotically the same as the distribution of variance-normalized range, W_t, of a set of normal random variables with variance = 1 and t samples [4]. Therefore, we can use the following relation to approximate $P[R \geq R_c]$

$$P[W_{t,\alpha} \geq \frac{2R_c - \frac{1}{2}}{\sqrt{nt}}] \tag{6.14}$$

In Equation (6.14), $W_{t,\alpha}$ is the value of the upper percentage point of W_t at significance point α. The values of $W_{t,\alpha}$ are tabulated in statistics books for example the one provided in [8].

The significance test for the differences between scores proceeds as follows:

1. Choose the desired significance level α.
2. Compute the critical value R_c using the following relation
$$R_c = \left\lceil \frac{1}{2} W_{t,\alpha} \sqrt{nt} + \frac{1}{4} \right\rceil \tag{6.15}$$
3. Any difference between 2 scores that is lower than R_c is declared to be insignificant. Otherwise, the score difference is declared significant.

6.3. Test procedure

User test mechanism to measure the impact of geometric distortion on the human perception of image quality is not widely discussed in the literature. Therefore, we have proposed a new user test system that is specifically designed for this purpose. In this section we shall describe in more detail the design of a suitable test set and user interface for such user test system.

6.3.1. Test set

We used two images, Bird and Kremlin, as a basis to build the test set for our experiment. These images, shown in Figure 6.2, are 8-bit grayscale images with 512 × 512 pixels resolution. The images are chosen primarily due to their content. The first image, Bird, does not have many structures such as straight lines. Furthermore, not every test subject is very familiar with the shape of a bird (in particular the species of bird depicted in the image). So in this case, a subject should have little (if any) "mental picture" of what things should look like. On the other hand, the Kremlin picture has a lot of structures and even though a test subject may not be familiar with the Kremlin, he/she should have some prior knowledge of what buildings should look like.

We used 17 different versions of the images. Each version is geometrically distorted in a different way. Thus in our test we use $t = 17$. The geometric distortions used in the experiment are shown in Table 6.1. In this table, we use the notation A_i, with $i = 1, 2, \ldots 17$ to identify each image.

The distortions chosen for the test set range from distortions that are perceptually not disturbing to distortions that are easily visible. The global bending distortions $\{A_6, A_7, A_8, A_9\}$ are chosen because these kinds of distortions are, up to a certain extent, visually not very disturbing in natural images. However, this distortion severely affects the PSNR value of the distorted images. The sinusoid (stretch-shrink) distortions $\{A_{10}, A_{11}, A_{12}, A_{13}\}$ distort the image by locally stretching and shrinking the image. Depending on the image content, this kind of distortion may not be perceptually disturbing. The rest of the distortions distorts the image by shifting the pixels to the left/right or upwards/downwards. These distortions are easily visible, even when the severity is low. The distortions $\{A_2, A_3, A_4, A_5\}$ apply the same distortion severity over the whole image, while the severity of distortions $\{A_{14}, A_{15}, A_{16}, A_{17}\}$ is varied within the image. Some examples of the geometric distortions used in the experiment are shown in Figure 6.3.

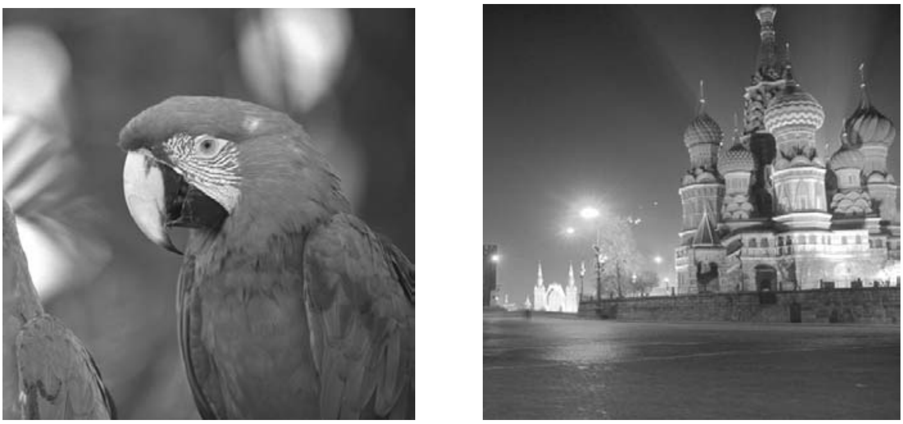
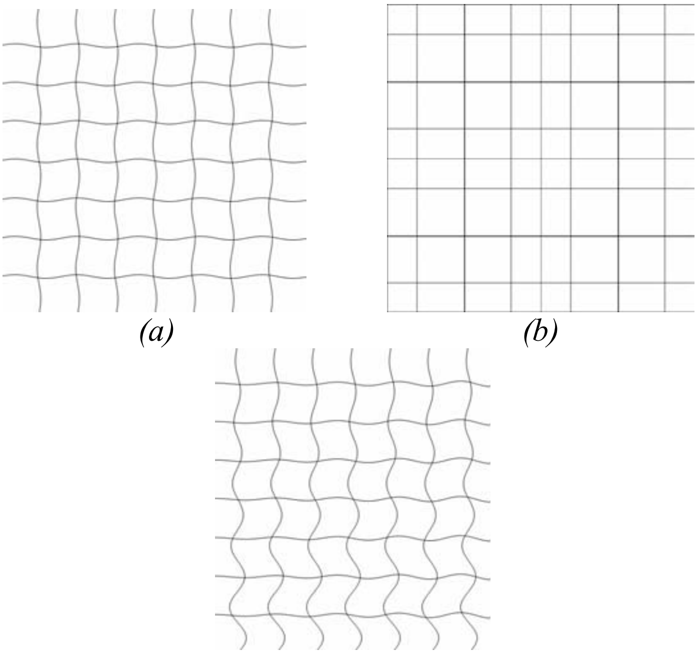


Figure 6.2. The 2 basis images: (a) Bird and (b) Kremlin

Table 6.1. Geometric distortions used in the experiment

Image	Description
A_1	No distortion (original image)
A_2	Sinusoid, amplitude factor = 0.2, 5 periods
A_3	Sinusoid, amplitude factor = 0.2, 10 periods
A_4	Sinusoid, amplitude factor = 0.5, 5 periods
A_5	Sinusoid, amplitude factor = 0.5, 10 periods
A_6	Global bending, bending factor = 0.8
A_7	Global bending, bending factor = - 0.8
A_8	Global bending, bending factor = 3
A_9	Global bending, bending factor = -3
A_{10}	Sinusoid (stretch-shrink), scaling factor 1, 0.5 period
A_{11}	Sinusoid (stretch-shrink), scaling factor 1, 1 period
A_{12}	Sinusoid (stretch-shrink), scaling factor 3, 0.5 period
A_{13}	Sinusoid (stretch-shrink), scaling factor 3, 1 period
A_{14}	Sinusoid (increasing freq), amplitude factor = 0.2, starting period = 1, freq increase factor = 4
A_{15}	Sinusoid (increasing freq), amplitude factor = 0.2, starting period = 1, freq increase factor = 9
A_{16}	Sinusoid (increasing amplitude), start amplitude factor = 0.1, 5 periods, amplitude increase factor = 4
A_{17}	Sinusoid (increasing amplitude), start amplitude factor = 0.1, 5 periods, amplitude increase factor = 9

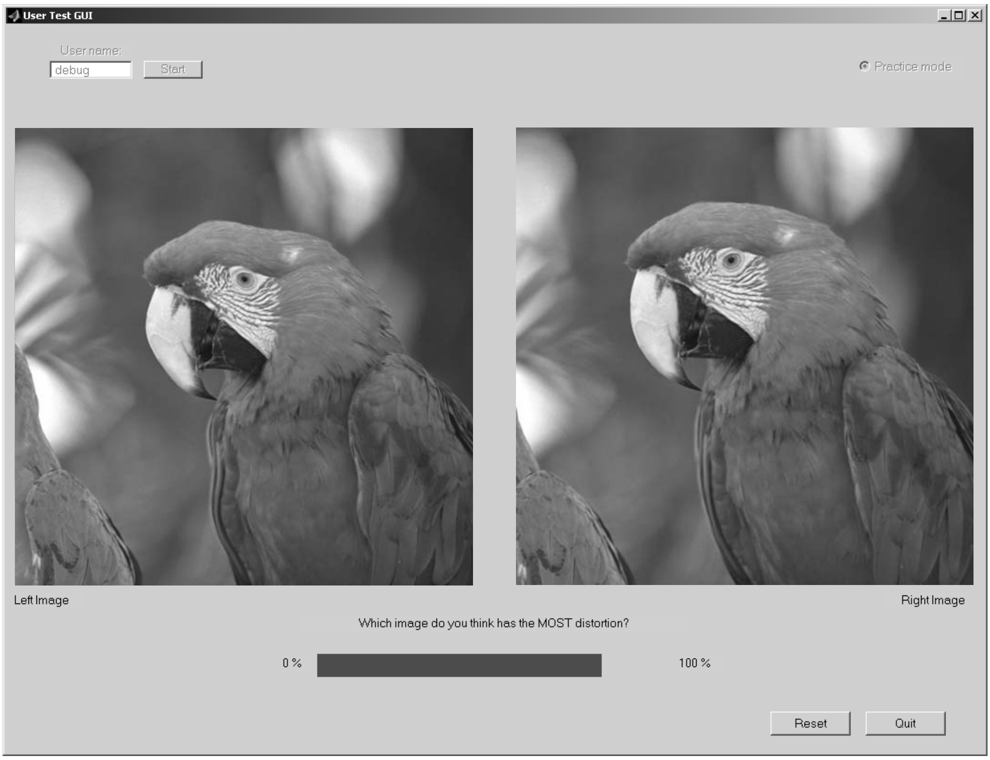
We then proceed to make all possible comparison pairs out of the 17 images, including the comparison of an image with itself. In each pair, we designate the first image as the left image and the other as the right image. This refers to how the images are to be presented to the subjects (see Figure 6.4). We then repeat each pair once, with the left-right ordering of the images reversed. Thus we have 306 pairs of images for each of the two images for a total of 612 pairs of images in the test set.



(a) (b)

(c)

Figure 6.3. Examples of the geometric distortions:
(a) Distortion A_5, (b) Distortion A_{13} and (c) Distortion A_{16}



User name: debug

Which image do you think has the MOST distortion?

Figure 6.4. The user interface used in the experiment

6.3.2. Test subjects

The user test experiment involved 16 subjects, consisting of 12 male (IL, ON, PD, AH, ES, DS, IS, JO, JK, JJ, KK and RH) and 4 female (KC, CL, CE and ID) subjects. The subjects have different backgrounds and levels of familiarity with the field of digital image processing. As discussed in Section 6.3.1, each user will examine each pair of test images twice in one test session. Furthermore, subjects IL, DS and IS each perform 3 test sessions. Therefore, in the tables found in Section 6.4, a number will be added to the subject names to show different test sessions (eg., IL1 shows the result of subject IL from the 1st test, etc.). These repetitions are done to see the difference between test results for one person when the test is repeated. We assume that each repetition of the test (both within a single test session and between test sessions) is independent. Therefore, we have the total number of test repetitions $n = 44$.

6.3.3. Test procedure

The test is performed on a PC with a 19-inch flatscreen CRT monitor. The resolution is set at 1152 × 864 pixels. The vertical refresh rate of the monitor is set at 75 Hz. To perform the test, a graphical user interface is used. This user interface is shown in Figure 6.4.

6.4. Test results and analysis

6.4.1. User preference matrix

After performing the user test, we obtain the preference matrices for the Bird and Kremlin images. In Figures 6.5(a) and 6.5(b), we show the preference matrices obtained for the Bird and Kremlin test images. These preference matrices are available for downloading from our website [3]. The images codes refer to Table 6.1. The column a_i shows the sum of each row, i.e., the score of each image A_i. Since in our experiment the test subject is asked to choose the image with the *most* distortion, a smaller score a_i means that the image is perceptually better.

	A_1	A_2	A_3	A_4	A_5	A_6	A_7	A_8	A_9	A_{10}	A_{11}	A_{12}	A_{13}	A_{14}	A_{15}	A_{16}	A_{17}	a_i
A_1	×	7	3	0	0	11	21	19	8	8	24	2	10	9	1	0	0	123
A_2	37	×	4	0	0	33	28	29	24	30	36	10	11	12	5	1	1	261
A_3	41	40	×	3	1	42	43	39	39	41	42	25	18	37	9	5	0	425
A_4	44	44	41	×	3	44	44	42	43	43	43	37	39	43	31	15	1	557
A_5	44	44	43	41	×	44	44	44	43	43	44	42	43	44	43	42	24	672
A_6	33	11	2	0	0	×	25	15	17	15	33	4	11	8	1	0	0	175
A_7	23	16	1	0	0	19	×	15	13	21	28	3	11	5	2	0	0	157
A_8	25	15	5	2	0	29	29	×	12	17	27	6	10	9	1	1	0	188
A_9	36	20	5	1	1	27	31	32	×	30	40	8	15	15	2	2	0	265
A_{10}	36	14	3	1	1	29	23	27	14	×	34	6	9	9	0	0	0	206
A_{11}	20	8	2	1	0	11	16	17	4	10	×	4	5	6	1	0	0	105
A_{12}	42	34	19	7	2	40	41	38	36	38	40	×	20	31	9	5	1	403
A_{13}	34	33	26	5	1	33	33	34	29	35	39	24	×	25	17	5	0	373
A_{14}	35	32	7	1	0	36	39	35	29	35	38	13	19	×	6	1	0	326
A_{15}	43	39	35	13	1	43	42	43	42	44	43	35	27	38	×	7	2	497
A_{16}	44	43	39	29	2	44	44	43	42	44	44	39	39	43	37	×	1	577
A_{17}	44	43	44	43	20	44	44	44	44	44	44	43	44	44	42	43	×	674

Figure 6.5(a) Preference matrix for the Bird image

	A_1	A_2	A_3	A_4	A_5	A_6	A_7	A_8	A_9	A_{10}	A_{11}	A_{12}	A_{13}	A_{14}	A_{15}	A_{16}	A_{17}	a_i
A_1	×	1	1	1	1	17	3	2	1	23	8	22	4	0	0	1	0	85
A_2	43	×	1	0	0	43	41	32	17	42	40	44	30	13	1	1	0	348
A_3	43	43	×	3	0	44	44	36	38	44	44	44	42	41	21	2	0	489
A_4	43	44	41	×	0	44	44	43	42	44	44	44	44	43	43	14	2	579
A_5	43	44	44	44	×	44	44	44	43	44	44	44	43	44	44	42	26	681
A_6	27	1	0	0	0	×	8	2	2	25	16	24	6	1	1	0	0	113
A_7	41	3	0	0	0	36	×	10	0	36	26	35	15	0	0	0	0	202
A_8	42	12	8	1	0	42	34	×	10	39	41	42	29	12	3	0	0	315
A_9	43	27	6	2	1	42	44	34	×	44	43	43	39	24	7	0	0	399
A_{10}	21	2	0	0	0	19	8	5	0	×	16	15	6	1	0	0	1	94
A_{11}	36	4	0	0	0	28	18	3	1	28	×	26	9	1	0	0	0	154
A_{12}	22	0	0	0	0	20	9	2	1	29	18	×	4	2	1	0	0	108
A_{13}	40	14	2	0	1	38	29	15	5	38	35	40	×	13	3	1	0	274
A_{14}	44	31	3	1	0	43	44	32	20	43	43	42	31	×	4	0	0	381
A_{15}	44	43	23	1	0	43	44	41	37	44	44	43	41	40	×	7	0	495
A_{16}	43	43	42	30	2	44	44	44	44	44	44	44	43	44	37	×	0	592
A_{17}	44	44	44	42	18	44	44	44	44	43	44	44	44	44	44	44	×	675

Figure 6.5(b) Preference matrix for the Kremlin image

6.4.2. Statistical analysis of the preference matrix

6.4.2.1. Coefficient of consistency (ζ)

We measured the coefficient of consistency for individual test subjects using Equation (6.5a) since we have $t = 17$. Since each test subject performs the user test twice per session, we use the average value of ζ as an indication of each subject's consistency. The average coefficient of consistency is presented in Table 6.2.

Table 6.2. Coefficient of consistency (ζ)

Subject	Bird	Kremlin	Subject	Bird	Kremlin
IL1	0.83	0.93	DS1	0.67	0.87
IL2	0.83	0.91	DS2	0.73	0.92
IL3	0.85	0.95	DS3	0.82	0.93
KC	0.85	0.86	IS1	0.92	0.95
ON	0.94	0.98	IS2	0.94	0.93
PD	0.70	0.87	IS3	0.94	0.97
AH	0.87	0.96	JO	0.93	0.97
CL	0.82	0.90	JK	0.90	0.96
CE	0.83	0.94	JJ	0.85	0.88
ES	0.89	0.94	KK	0.70	0.79
ID	0.66	0.90	RH	0.90	0.95

From Table 6.2 we can conclude that in general the test subjects are consistent in their decisions. We can also see that in general the values of ζ for the Bird image are lower than those of the Kremlin image. This is due to the fact that the Kremlin image contains more structure compared to the Bird image, which helps the test subjects to make consistent decisions. Furthermore, the unfamiliarity of the test subjects with the particular species of bird depicted in the image also makes it difficult to make consistent decisions.

6.4.2.2. Coefficient of agreement (u)

We measured two types of coefficient of agreement from the preference matrix. The first is the overall coefficient of agreement that measures the agreement among all test subjects in the experiment. The second is the individual coefficient of agreement that measures the agreement of a test subject with him-/herself during the two repetitions in a test session. A low u value in this case would indicate that the subject is confused and does not have a clear preference for the images being shown.

For the overall coefficient of agreement, we have $n = 44$ and $t = 17$. For these values, the maximum and minimum values of u are 1 and -0.0227, respectively. From the preference matrices, we can calculate that the overall coefficient of agreements are $u_{bird} = 0.574$ and $u_{kremlin} = 0.731$. Performing the significance test on both u values using the method described in Section 6.2.2.2 shows that in both cases the value of u is significant at $\alpha = 0.05$. Therefore, we can conclude that in both cases there are strong agreements among the test subjects. However, we can also see that the agreement in the case of the Bird image is much weaker than the Kremlin image, due to the image content.

Table 6.3. Individual Coefficient of Agreements(u)

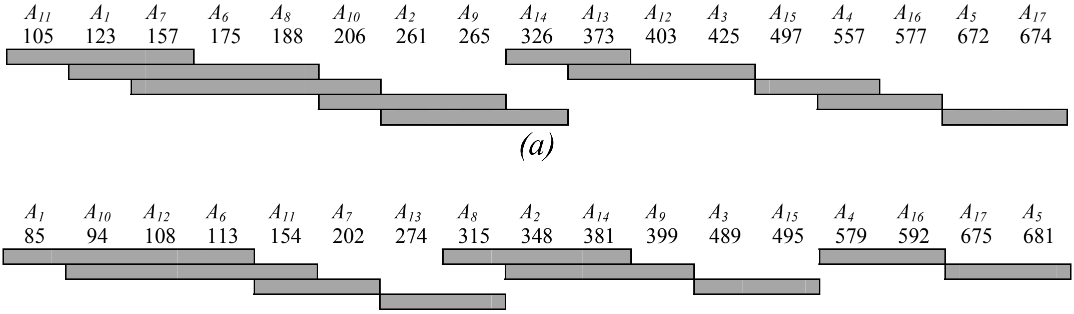
Subject	Bird	Kremlin	Subject	Bird	Kremlin
IL1	0.559	0.750	DS1	0.265	0.647
IL2	0.574	0.721	DS2	0.471	0.794
IL3	0.677	0.779	DS3	0.559	0.750
KC	0.662	0.691	IS1	0.721	0.882
ON	0.779	0.868	IS2	0.809	0.721
PD	0.485	0.677	IS3	0.735	0.838
AH	0.559	0.794	JO	0.721	0.853
CL	0.456	0.750	JK	0.824	0.735
CE	0.618	0.691	JJ	0.529	0.691
ES	0.765	0.765	KK	0.368	0.515
ID	0.279	0.691	RH	0.691	0.765

For the individual coefficient of agreement, we have $n = 2$ and $t = 17$. In this case, we have $-1 \leq u \leq 1$. The individual coefficient of agreements are presented in Table 6.3. As expected, we see that all subjects have larger u values for the Kremlin image. The exceptions to this are subject ES, who has the same u values for both images, and subjects IS2 and JK, who have larger u for the Bird image. After performing the significance test on the values of u, we can conclude that all subjects have u values that are significant at $\alpha = 0.05$ for both the Bird and Kremlin images.

6.4.2.3. Significance test of score differences

The strong agreements among the test subjects for both images, as shown in the previous section, show that there exist significant differences among the scores of the test objects. We use the procedure described in Section 6.2.2.4 to find the critical value for the score difference for the images, at significance level $\alpha = 0.05$. From [8] we have $W_{t,\ \alpha}= 4.89$. Substituting this value into Equation (6.15), we have $R_c = 67.12$ and thus we set $R = 68$. Therefore, only objects having a score difference of more than 68 are to be declared significantly different.

In Figure 6.6, we present the grouping of the images in the test set based on the significance of the score differences. The images have been sorted from left to right based on their scores, starting from the image with the smallest score (i.e., perceived to have the highest quality) to the one with the largest score. The score for each image is shown directly under the image code. Images having a score difference smaller than 68 are grouped together. This is represented by the shaded boxes under the image code. For example, in Figure 6.6(a), images A_{14} and A_{13} belong to one group.



(a)

(b)

Figure 6.6. Score grouping for: (a) Bird image and (b) Kremlin image

From Figure 6.6, we can see that the images occupying the last 6 positions of the ranking for both the Bird and Kremlin images are distorted using the same distortion. Furthermore, they are sorted in the same order (except for images A_5 and A_{17}, but the difference between their scores is not significant). Thus we can conclude that these distortions are perceived similarly by the test subjects, regardless of the image content. These distortions occupy the "lower quality" segment of the ranking so we can also conclude that the distortions are so severe that the image content no longer plays a significant role. For the other images, the influence of image content on the perceived quality of the distorted images is larger.

Table 6.4. Group u values

Bird			Kremlin		
Group	*u*	Significant?	Group	*u*	Significant?
$A_{11}A_1A_7$	0.006	No	$A_1\,A_{10}\,A_{12}\,A_6$	0.008	No
$A_1A_7A_6A_8$	0.061	Yes	$A_{10}\,A_{12}\,A_6\,A_{11}$	0.03	Yes
$A_7A_6A_8A_{10}$	0.041	Yes	$A_{11}\,A_7$	0.011	No
$A_{10}A_2A_9$	0.07	Yes	$A_{13}\,A_8$	0.08	Yes
$A_2A_9A_{14}$	0.085	Yes	$A_8\,A_2\,A_{14}$	0.175	Yes
$A_{14}A_{13}$	-0.004	No	$A_2\,A_{14}\,A_9$	0.054	Yes
$A_{13}A_{12}A_3$	-0.003	No	$A_3\,A_{15}$	-0.021	No
$A_{15}A_4$	0.148	Yes	A_4A_{16}	0.112	Yes
A_4A_{16}	0.08	Yes	$A_{17}\,A_5$	0.011	No
A_5A_{17}	-0.015	No	-	-	-

Table 6.4 shows the overall *u* values for each score group. We expect that when the images in a group do not have significantly different scores, there will not be any clear preference for any of them among the test subjects and therefore the *u* values should be low. The groups are presented in the 1st and 4th columns using their members as group names. The 3rd and 6th columns of the

table show the result of the significance test for *u*, as described in Section 6.2.2.2, with significance level $\alpha = 0.05$.

We can conclude from Table 6.4 that the *u* values for each group are very low. Some groups even have *u* values that are not significantly larger than the *u* values that would have been achieved had the votes within that group been cast at random. This result shows that indeed the grouping of the images performed based on the significance of score differences has produced groups within which the perceived quality is difficult to distinguish.

6.4.3. Conclusions

From the analysis of the user test results, we can draw the following conclusions:

1. The test objects are generally perceptually distinguishable by the test subjects. This is supported by the fact that the consistency of the test subjects is relatively high, as shown in Table 6.2. Furthermore, we also see that the individual *u* values (shown Table 6.3) are also high.
2. There is a general agreement as to the relative perceptual quality of the test images among the test subjects. This is supported by both the high overall *u* values. Therefore, we can make a ranking of the images based on their perceived quality.
3. For some images, the relative perceptual quality among them is not clearly distinguishable. We can see this from the grouping of the scores based on the significance test of score differences. This is further supported by the lack of agreement among test subjects regarding the relative quality of images within such groups.

6.5. Evaluation of the objective perceptual quality measurement method

6.5.1. Overview of the method

The objective geometric distortion measurement is based on the ideas in our previously published work [9] and further developed and described in [2]. The algorithm is based on the hypothesis that the perceptual quality of a geometrically distorted image depends on the homogeneity of the geometric distortion. We call our proposed scheme the Homogeneity-based Perceptual Quality Measurement (HPQM). The less homogenous the geometric distortion is, the lower the perceptual quality of the image will be. We proposed a method to measure this homogeneity by approximating the underlying geometric distortion using simple RST approximation. We increase the locality of our approximation until the level of approximation error is lower than a predetermined threshold or until the locality of the approximation reaches a predetermined maximum. The locality is increased using quadtree partitioning of the image, where smaller block sizes indicate higher approximation locality.

We then determine the score (i.e., the quality) of the image based on the resulting quadtree structure. In the objective test, the score that can be achieved by an image is normalized to the range of 0 – 100. A more detailed discussion of the proposed objective measurement algorithm has been presented in Chapter 5 of this thesis.

We have implemented some modifications to the algorithm to improve its performance. We briefly discuss the modifications as follows. The first modification is applied to the RST/affine parameter estimation procedure. In Chapter 5, we use an exhaustive (brute-force) search to estimate the local RST/affine transformation parameters. The main drawback of this approach is its slow speed. In order to be able to finish the calculations within a reasonable time-frame we have to severely limit the range of the RST/affine parameter search space. Furthermore, we also have to coarsely sample this search space. This in turn will limit the precision of the estimation process. In this chapter we base our scheme on the Optical Flow Estimation (OFE) algorithm to estimate the RST/affine parameters [10, 11]. The use of this algorithm significantly improves the speed of the system and also increases the precision of the parameter estimation process. We performed the (OFE) algorithm in four resolution levels, namely $^1/_8$ resolution, ¼ resolution, ½ resolution and the original full resolution.

The second modification is applied to the computation of the final score of the distorted image. In Chapter 5, we only take into account the average block size of the quadtree partitioning and the residual error of the blocks to compute the final score. In this chapter, we further fine tune the scoring system by taking into account the estimated RST/affine parameters associated with each block. We look at these parameters to see they deviate from the RST/affine parameters when there is no RST/affine distortion. In our experiments, the deviation is expressed as the *l*-2 norm distance between the 2 sets of parameters. In calculating the deviation, the parameters for Rotation and Scaling are given larger weights compared to the parameters for Translation (the weights are experimentally determined). This is done since in our observation, the presence of Rotation and Scaling seems to be more visually disturbing compared to Translation. The larger the deviation of the estimated parameters, the lower the score for the block. Our experimental results show that this modification improves the performance of the measurement algorithm and makes it better match the result of the subjective test results. This is because even when the RST/affine transformation of a block can be perfectly estimated (i.e., zero residual error), such a block can still heavily influence the overall perceptual quality of the image if the local RST/affine transformation is severe.

6.5.2. Performance evaluation

In this section we shall evaluate the performance of our proposed objective quality measurement algorithm. In this performance evaluation we use the results of the subjective-test as a ground truth. In other words, the proposed algorithm will be considered to be performing well if its results have a good correspondence to the subjective-test results. Furthermore, in order to evaluate the performance of the proposed objective quality measurement algorithm relative to the performance of other possible measurement schemes, we also evaluate the performances of two other possible objective quality measurement schemes. The other possible measurement schemes we evaluate in this section are PSNR measurement and Motion-Estimation (ME)-based measurement scheme.

The PSNR measurement is a widely used tool used to evaluate the objective quality of images. Although this measurement does not always correspond well to human perception of quality, its performance is good enough to evaluate the quality of, for example, images degraded by additive noise. However, PSNR measurement relies heavily on the pixel-per-pixel correspondence between the images being evaluated. Since geometric distortion destroys this correspondence, PSNR measurement is not well suited for evaluating geometrically distorted images. Therefore, in our experiments the results of the PSNR measurement are used to indicate the worst-case scenario (i.e., an ineffective measurement scheme).

The second alternative objective quality measurement scheme we evaluate is an ME-based measurement scheme. This measurement scheme is inspired by the use of motion estimation techniques in image and video watermarking to deal with geometric distortion for example the technique presented in [12]. In order to use the motion estimation technique as a measurement scheme we take into account two outputs of the motion estimation process, namely the motion vector entropy and the variance of the prediction error. The motion vector entropy is used to indicate the “activity” of the distortion. A high activity means that various parts of the image are distorted in a different way. The higher the activity of the distortion, the lower the perceptual quality of the image. The variance of the prediction error shows the residual error after the motion estimation and compensation process. A large error variance indicates a heavy distortion and thus a lower perceptual quality. Our observations indicate that the motion vector entropy plays a more important role in determining the perceptual quality of the image. Therefore, we give this measurement parameter a larger weight than the residual error variance. These weights are determined experimentally. The proposed ME-based quality measurement (MEQM) scheme is presented in Figure 6.7. In our experiments, we chose a block size of 16 × 16 pixels, maximum displacement of 7 pixels and full-search method. This ME-based measurement approach is somewhat similar to the HPQM approach with two main differences. The first difference between the two is the simpler approximation model of the MEQM scheme. The MEQM scheme uses only translation instead of an RTS/affine model used by HPQM. The second difference is in the locality of the approximation. The MEQM scheme uses a fixed locality for the approximation. This locality is determined by the chosen block size. In other words, we can regard the MEQM scheme as a simpler, more restricted, version of the HPQM.

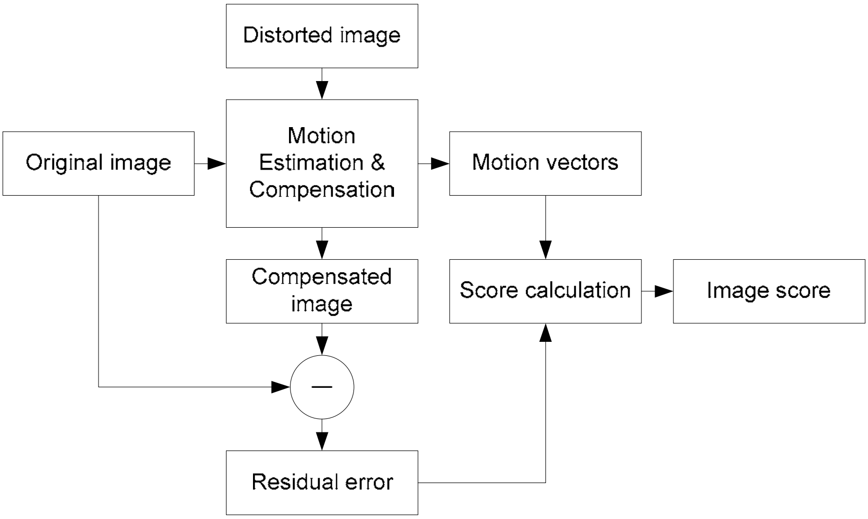
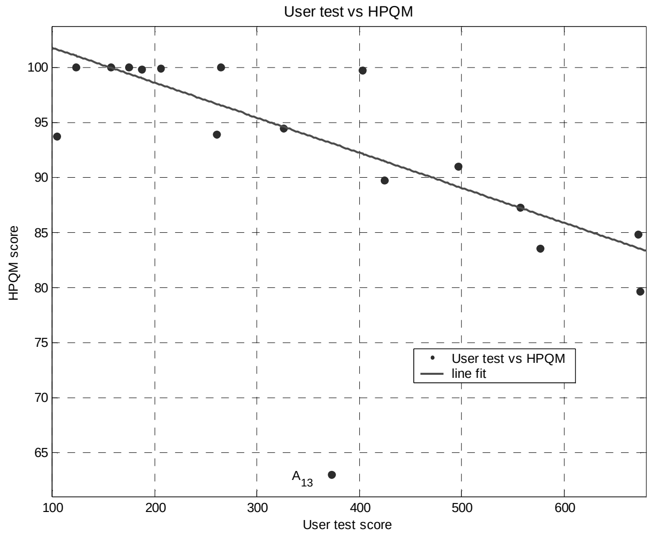


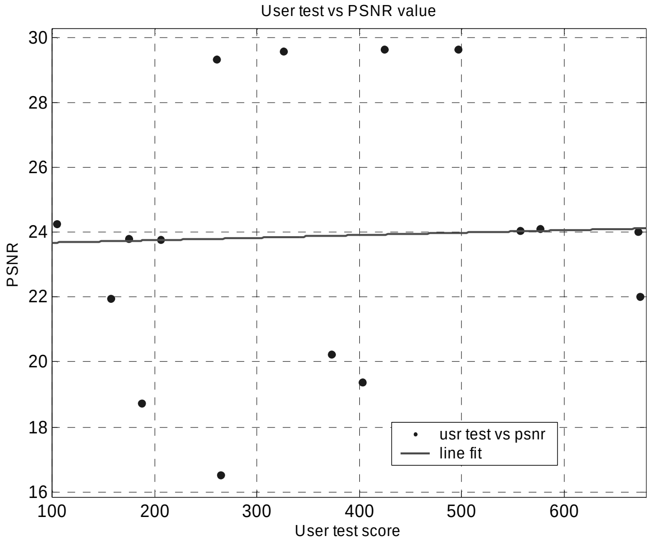
Figure 6.7. An ME-based measurement scheme

In evaluating the performance of the objective quality measurements, we look at the *intra-* and *inter-distortion* comparisons. For intra-distortion comparisons, we evaluate the scores of the images within one type of geometric distortion, but with different distortion parameters. For example, we perform an intra-distortion comparison by evaluating the scores of images A_2, A_3, A_4 and A_5 that are distorted by the same sinusoid distortion but with different parameters (see Table 6.1). In this comparison, an image with a more severe distortion parameters should get a lower score. For inter-distortion comparisons, we evaluate the scores of all images in the test set. This is a more difficult test for the objective quality measurement schemes since they have to be able to indicate the relative perceptual qualities between different types of geometric distortions.

All measurement schemes that we evaluated in our experiments, including PSNR measurement, perform well in the intra-distortion comparison. In other words, the images distorted with a more severe parameter set are correctly given lower scores. In order to evaluate the performance of the objective quality measurement schemes in performing inter-distortion comparisons, we plot their results against the subjective test scores. The comparison plots for the Bird image set is shown in Figure 6.8. The plots for the Kremlin image set show similar behavior.

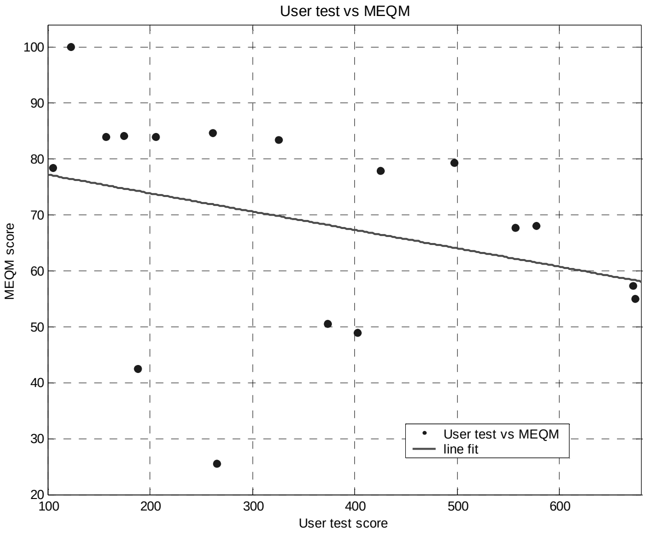


(a)



(b)

Figure 6.8. Result comparisons for the Bird image: (a) User test vs. HPQM, (b) User test vs. PSNR



(c)
Figure 6.8. (continued):
(c) User test vs. MEQM

From Figure 6.8(b) we can see that the PSNR measurement has a very poor correspondence to the subjective test result. This is shown by the regression line that is virtually horizontal. The value of the correlation coefficient ρ in this case also reflects this fact, namely we have ρ_{up} = 0.14. The MEQM scheme performs much better than PSNR measurement as shown in Figure 6.8(c) and with ρ_{um} = -0.32. We can also see that the HPQM scheme gives the best performance among the three evaluated schemes, as shown in Figure 6.8 (a) and with ρ_{uh} = -0.6. The negative values of ρ_{um} and ρ_{uh} correctly reflect the fact that in our experiments a larger subjective test score represents a lower perceptual quality.

If we evaluate Figure 6.8(a) we can see that image A_{13} does not properly fit the behavior of the rest of the data set and can be considered an outlier. Removing this image from the data set and recalculating the correlation coefficient, we get ρ_{uo} = -0.87. In general, we observe the HPQM scheme cannot handle images distorted by the *sinusoid (stretch-shrink)* distortion (see Table 6.1) well, except for image A_{10}[1]. At present, we do not yet have a satisfactory explanation regarding this phenomenon. In the case of image A_{10}, the geometric transformation applied to this image is similar to the one implemented in television broadcasting when it is necessary to convert video frames from one aspect ratio to another. This transformation is perceptually not disturbing (unless there is a lot of movement, for example camera panning), and

[1] Similarly, the MEQM scheme also seems to have difficulties in dealing with this type of distortions.

therefore, our test subjects give this image a high ranking. In this distortion, the image is stretched slightly in the horizontal and vertical direction. The slight increase in image width and height is compensated by shrinking the outer parts of the image. This distortion can be approximated by slightly enlarging the original image. Since this is a homogenous RTS approximation, the HPQM scheme gives this image a high score. Image A_{11} of the Bird test set is interesting since the subjects prefer this image to the undistorted image A_1. This is probably due to the unfamiliarity of the subjects to the bird species shown in the picture. Apparently, the test subjects get the impression that the size of the bird's head in the original image was either too large or too flat. Therefore, they preferred the image in which the head of the bird is slightly shrunk horizontally (and consequently slightly rounder). The fact that this does not happen in the Kremlin test set (see Figure 6.6.(b)) seems to support this conclusion.

6.6. Conclusion and future works

In this chapter, we have described the method we use to perform a perceptual user test for geometrically distorted images. We also described the statistical tools we use to analyze the results of the user test. The result of the user test is then used as a ground truth to validate our objective perceptual quality measurement scheme, the HPQM, which is based on the hypothesis that the perceptual quality of a distorted image depends on the homogeneity of the geometric transformation causing the distortion. Furthermore, in order to have a better assessment of the performance of the HPQM, we also compare its performance to the performance of the PSNR measurement and the MEQM scheme. In our experiments, we evaluate the performance of all three objective measurement schemes in two areas, namely in performing intra- and inter-distortion comparisons.

All objective measurements evaluated in our experiments, the HPQM, PSNR and MEQM, give similar performance in performing intra-distortion comparisons. For inter-distortion comparisons, the PSNR measurement performs poorly. The MEQM and HPQM schemes outperform PSNR measurement in this category, with the HPQM giving the best performance of among these two schemes.

While the amount of data collected in our experiments is not yet large enough to form firm conclusions, we observe a very strong tendency that our HPQM scheme has a very good overall correspondence to the results of the subjective test. The scheme is not yet perfect, however, and we still observe some discrepancies between the ranking of the images generated by HPQM to that generated by the subjective test result.

In the future, more measurements and user test experiments similar to the one described and analyzed in this chapter should be performed. The data collected from such experiments can than be used to further validate or refine the hypothesis and to further fine-tune the performance of the HPQM scheme. Finally, other objective quality measurement approaches should also be explored and tested.

6.7. References

1. Keelan, B.W., *Handbook of Image Quality: Characterization and Prediction*, Marcel Dekker, Inc., New York, 2002
2. I. Setyawan, D. Delannay, B. Macq and R.L. Lagendijk, *Perceptual Quality Evaluation of Geometrically Distorted Images using Relevant Geometric Transformation Modelling*, in Proceedings of SPIE, Security and Watermarking of Multimedia Contents V, Vol. 5020, pp. 85 – 94, Santa Clara, CA, 2003
3. www-ict.its.tudelft.nl/~iwan/user_test_result.html.
4. David, H.A., *The Method of Paired Comparisons*, 2nd ed., Charles Griffin & Company, Ltd., London, 1988
5. Bechhofer, R.E., T.J. Santner and D.M. Goldsman, *Design and Analysis of Experiments for Statistical Selection, Screening and Multiple Comparisons*, John Wiley & Sons, Ltd., New York, 1995
6. Kendall, M.G., *Rank Correlation Methods*, 4th ed., Charles Griffin & Company, Ltd., London, 1975
7. Siegel, S., and N. J. Castellan, Jr., *Nonparametric Statistics for the Behavioral Sciences*, 2nd ed., McGraw-Hill, Boston, 1988
8. Pearson, E.S. and H.O. Hartley, *Biometrika Tables for Statisticians*, Vol 1, 3rd ed., Cambridge University Press, 1966.
9. D. Delannay, I. Setyawan, R.L. Lagendijk and B. Macq, *Relevant Modelling and Comparison of Geometric Distortions in Watermarking Systems*, in Proceedings of SPIE, Application of Digital Image Processing XXV, Vol. 4790, pp. 200 – 210, Seattle, WA, 2002
10. Tekalp, A.M., *Digital Video Processing*, Prentice-Hall, Inc., Upper Saddle River, 1995
11. Tekalp, A.M., *Differential Methods*, part of the lecture notes for Digital Video Processing, University of Rochester, New York, USA, 2001
12. D. Delannay, J-F Delaigle, B. Macq and M. Barlaud, *Compensation of geometrical deformations for watermark extraction in the digital cinema application*, in Proceedings of SPIE, Security and Watermarking of Multimedia Contents III,Vol. 4314, pp. 149 – 157, San Jose, CA, 2001

Chapter 7 CONCLUDING REMARKS

7.1. Looking back: Summary of the results

In this thesis, we discussed two challenges in watermarking image and video data. In Chapter 3, we discussed the challenges of embedding digital watermarks in low bit-rate compressed video, where we presented a scheme to embed watermarks into low bit-rate MPEG2 video. As discussed in Chapter 3, at low bit-rates (less than 256 kbps) the main limitation to the performance of this scheme is the performance of the MPEG2 encoder. Further research in the area of low bit-rate video watermarking should therefore move to video encoders with better low bit-rate performance, for example MPEG4. Examples from the literature of low bit-rate video watermarking for standard MPEG4 video have been discussed in Chapter 3. Furthermore, currently we have the popular DivX and Xvid formats which are also based on MPEG4. These formats are particularly popular for the production of small, but relatively high quality "rips" of movies distributed on DVD, which are then distributed illegally over the internet. Therefore, in copyright protection scenarios, the development of watermarking schemes robust against these compression schemes is very important.

Despite the continuing decline of the price of bandwidth and storage space today, low bit-rate images and video will continue to play an important role. In addition to the application discussed above, nowadays we see the increasing importance and popularity of devices such as mobile phones and PDA's. Currently, these devices are also used to send images and video. These images and videos will have to be compressed in low bit-rate due to bandwidth and device limitations. If these images and videos are to be watermarked, a watermarking scheme designed for low bit-rate data will play a crucial role.

The second problem discussed in this thesis is the problem of dealing with geometric distortion in images and video. As discussed in Chapters 4, 5 and 6, geometric distortion has two aspects. The first aspect deals with the synchronization of the watermark and the second aspect deals with the perceptual quality of the attacked images or videos. The main quality a watermarking algorithm has to posses to be able to be robust against this distortion is the ability to resist the desynchronizing effect of the distortion. One way to achieve this is by removing the dependence of the watermark on strict synchronization to the watermark detector. The first part of Chapter 4 discusses the concept of such a watermarking system. Currently, the problem with this concept is the security aspect, since it is fairly easy for an attacker to detect and possibly remove the watermark. An alternative solution to this problem is to design the watermark such that it can be resynchronized with the detector after a geometric distortion. One way to do this is by using the features of the original undistorted host data as reference points to reverse the distortion incurred by the geometric transformation. In the second part of Chapter 4, a watermarking scheme employing this approach has been discussed. We show that this scheme can be implemented on top of an existing watermarking system to increase its robustness against geometric distortion. However, this solution still suffers from the unreliability of current feature extraction and matching algorithms. Such unreliability can result in the presence of mismatches of the feature points. Currently the watermarking approach presented is very sensitive to such mismatches.

Finally, we address the perceptual quality problem of geometric distortion by proposing a method to objectively measure the quality of geometrically distorted images. We test the performance of the proposed method by comparing it to the result of a subjective test and the result of a PSNR measurement and an ME-based measurement system. From this comparison, we conclude that the proposed hypothesis and measurement method show promising results. The method gives a better performance compared to the ME-based measurement and the PSNR measurement. Furthermore, it shows good correspondence to the result of the user test. The main limitation to the proposed method is the fact that we have not been able to completely take image content into account when performing the measurement. Modifying the system to fully take image content into account will be very challenging. However, this is not the only modification that can be applied to the system to further improve its performance. In particular, the behavior of the local RST transformation should be further investigated. In Chapter 6, we looked at how severe the individual local RST transformation is by evaluating its transform parameters. However, looking at the behavior of these RST transformation blocks individually may not be enough. In future research, we should also investigate how the relative behavior of a local RST transformation block with respect to its neighbors affects the overall perceptual quality.

7.2. Looking forward: Future challenges

Digital watermarking technology has achieved significant progress since the initial burst of research activities in the mid 1990’s and currently research activities to develop watermarking schemes with even better performance are still being performed. The research activities are not only directed at developing new watermarking methods, but also at establishing a stronger understanding of the underlying theoretical questions in watermarking. Despite the advances that have been achieved there are still many unresolved problems that will keep digital watermarking an active and challenging research area in the future.

The first main challenge in the future will be watermark robustness against synchronization attacks, especially the spatial geometric distortion applied to images and video frames. In particular, more research effort should be devoted to methods to quantify the perceptual quality impact of the geometric distortion. While the desynchronization aspect of geometric distortion problem has not been completely solved yet, a lot of research efforts have been performed in this area and currently many approaches exist to deal with this problem. Examples of the existing approaches have been presented in Chapters 2, 4, 5 and 6 of this thesis. On the other hand, research on a method to quantify the perceptual quality impact of geometric distortion is still in its infancy. The benefits of such a method for the watermarking community have been discussed in the previous chapters. In addition to the benefits to the watermarking community, this research will also benefit other research areas dealing with image and video processing by giving more understanding of how humans perceive geometric distortion and a tool with which to objectively measure the distortion. The watermarking community should take up the challenge and spearhead the research in this area.

The second main challenge in the future is the development of applications in which digital watermarking is not used in a copyright protection scenario. As stated in the beginning of this thesis, digital watermarking technology was originally expected to solve the copyright protection problem of digital media. The performance of current watermarking schemes is still considered to be insufficient to perform this task, particularly their robustness against intentional attacks. While it would be naïve to expect a watermarking system that is robust against any possible attack, the performance of current watermarking systems still leaves some room for improvements. That being said, researchers should not concentrate only on the copyright protection scenario, but should also look into other possible applications of digital watermarking.

In order to do this, we should return to the main essence of digital watermarking, namely that it is a method of embedding unobtrusive information into host data. In other words, digital watermarking should be seen as a method to transmit extra information without requiring extra bandwidth or storage space. One scenario that can take advantage of this concept is, for example, a combination of text and picture messaging for a mobile device. Instead of sending the text and the picture separately, watermarking techniques can be used to embed the text into the picture. The receiving device can extract the text and display it separately. Another example is to use digital watermarks to embed additional information in pictures printed in a magazine. The user can then use a scanner or a webcam to read and decode the watermark. This is particularly useful when the additional information cannot be printed due to page budget constraints or when the information needs to be updated regularly.

The applications discussed above have different watermarking requirements compared to copyright protection scenarios, namely:

- ***More limited attack types.*** Unlike in copyright protection scenarios where an attacker actively looks for ways to remove the watermark or hamper watermark detection, little gain can be achieved by deliberately attacking the watermark in the scenarios described above. Thus, the main types of attacks that will be encountered are unintentional attacks which should be easier to deal with.
- ***More relaxed imperceptibility requirements.*** The requirements for watermark imperceptibility imposed in copyright protection are very high since any perceived distortion on the watermarked data may reduce its value. In the scenarios discussed above, this requirement is much more relaxed.
- ***More emphasis on watermark payload.*** Since the watermark in the scenarios described above has a primary function as carrier of additional information, it is essential that the watermark has enough capacity. The capacity requirement for these scenarios is generally higher than in copyright protection scenarios where as little as 1 bit of information per picture is enough.

The examples mentioned in the previous paragraph are by no means complete and there are still other applications beyond copyright protection that can take advantage of watermarking technology; for example, compression and error detection can also make use of watermarking technology. These applications will help to familiarize ordinary users with watermarking technology and give a better picture of the capabilities and limitations of the technology.

SUMMARY

Digital watermarking, a technique used to embed information securely into digital data such as digital images, audio material and video, was born to anticipate the steadily increasing use of digital media. Digital media offer a lot of advantages for both user and content creator, including superior quality and ease of use. However, the ease with which digital media can be reproduced and manipulated is of great concern for the copyright owners of the material. Digital watermarking can play an important role in this situation to protect the copyright associated with the material. Digital watermarking can also be used in other applications not directly associated with copyright protection, for example data tracking, error detection, and correction and compression.

A more complete introduction to digital watermarking techniques is discussed in Chapter 2, where we discuss the basic concepts of digital watermarking technology, the requirements for a robust, invisible watermark and the applications in which digital watermarking can play a role. In this chapter, we also discuss attacks on the watermarking systems. We present a classification of these attacks based on how the attacks are performed and which part of the watermarking system is targeted.

Chapter 3 of this thesis discusses the challenge of watermarking low bit-rate compressed video. Low bit-rate compression severely limits the amount of space in which we can embed the watermark, which means that extra care has to be taken to ensure that the requirements for watermark imperceptibility, robustness and capacity are satisfied. We present in this chapter a watermarking system (based on our previous watermarking system, the DEW algorithm) suitable for low bit-rate applications.

The rest of the thesis deals with one of the most challenging problems in watermarking, namely geometric distortion in image and video. This distortion happens when geometric operations are applied to images or videos. Geometric distortion is relatively easy to perform, but it is very difficult to combat. Geometric distortion problems have two aspects. The first aspect is the watermark desynchronization aspect, which makes watermark detection impossible or very difficult. The second aspect is the perceptual quality impact which is very difficult to assess due to the lack of an appropriate model of human perception of such distortion.

In Chapter 4, we deal with the first aspect of the geometric distortion problem. We present two approaches to solve this problem in image and video watermarking:

- ***Removing watermark dependence on spatial synchronization.*** We propose a watermarking system, based on structured noise patterns, that does not require strict synchronization between the watermark and watermark detector. This system is invariant to translation and shows a better robustness against rotation and scaling than classic noise-based systems.
- ***Using host data features as reference points to invert the distortion.*** We propose a complexity-scalable strategy to invert the geometric distortion. We use features from the host data as reference points to register the distorted watermarked data and thus invert the geometric distortion. This strategy can be implemented on top of existing watermarking schemes to improve their robustness against geometric distortion.

Finally, we discuss the problem of quantifying the perceptual quality impact of geometric distortion in images. This aspect of the problem has not been widely studied in the literature and currently we do not have an objective measure to assess the perceptual quality impact of geometric distortion. We propose a new quality measurement method based on the hypothesis that the perceptual quality of a geometrically distorted image depends on the homogeneity of the distortion in Chapter 5. We evaluate the performance of this system in Chapter 6 by comparing it to the results of a user test.

The results described in this thesis can be summarized as follows:

- Development of a video watermarking scheme suitable for MPEG-1/-2 video encoded at low bit-rate (Chapter 3).
- Development of a watermarking scheme that does not rely on strict spatial synchronization for images and video (Chapter 4).
- Development of a complexity-scalable strategy to invert geometric distortion in image watermarking (Chapter 4).
- Development of an objective perceptual quality assessment method for geometrically distorted images (Chapters 5 and 6).

SAMENVATTING

Digitaal watermerken, een techniek die gebruikt wordt om informatie veilig te verweven met digitale data zoals digitaal beeld, geluid en video, is ontstaan doordat er een sterke toename is in het gebruik van digitale media. Digitale media bieden vele voordelen voor zowel de gebruiker als de maker ervan, met name hogere kwaliteit en verbeterd gebruiksgemak. Echter, het gemak waarmee de digitale media kunnen worden gereproduceerd en gemanipuleerd vormt een grote zorg voor de rechtmatige eigenaren van het materiaal. Digitaal watermerken kan in deze situatie een grote rol spelen om de intellectuele eigendom van het materiaal te beschermen. Digitaal watermerken kan ook gebruikt worden in andere toepassingen, die niet direct met het beschermen van intellectueel eigendom te maken hebben, bijvoorbeeld het traceren van data, detecteren en corrigeren van fouten en compressie.

Hoofdstuk 2 bevat een uitgebreide introductie op digitale watermerk-technieken, waar de basisconcepten van digitale watermerktechnologie, de eisen voor een robuust, onzichtbaar watermerk en de toepassingen waarin digitaal watermerken een rol kan spelen besproken wordt. Hetzelfde hoofdstuk behandelt ook aanvallen op watermerk systemen. Er wordt een classificatie van deze aanvallen gebaseerd op hoe deze aanvallen zijn uitgevoerd en op welk gedeelte van het watermerk systeem de aanval gericht is gepresenteerd.

Hoofdstuk 3 van dit proefschrift bespreekt de uitdaging van het watermerken van video die op lage bit-rate gecomprimeerd is. Compressie op een lage bit-rate zorgt voor een sterke beperking van de ruimte waarin het watermerk aangebracht kunnen worden, hetgeen extra zorg vergt om te garanderen dat de eisen voor onzichtbaarheid van het watermerk, robuustheid en de capaciteit gewaarborgd zijn. In dit hoofdstuk presenteren we een watermerk systeem (gebaseerd op ons vorig watermerk systeem, het DEW algoritme) dat geschikt is voor lage bit-rate toepassingen.

De rest van dit proefschrift behandelt één van de meest uitdagende problemen in watermerken, te weten geometrische vervorming van beeld en video. Deze vervorming vindt plaats wanneer geometrische handelingen op beelden en video's plaatsvinden. Geometrische vervorming treedt relatief eenvoudig op, maar is erg moeilijk te bestrijden. Het probleem van geometrische vervorming heeft 2 aspecten. Het eerste aspect betreft de desynchronisatie van het watermerk, waardoor detectie van het watermerk wordt bemoeilijkt of zelfs verhinderd. Het tweede aspect betreft het

waarneembare kwaliteitseffect, dat erg moeilijk te waarderen is bij gebrek aan een geschikt model van de menselijke perceptie van dergelijke vervormingen.

In hoofdstuk 4 behandelen we het eerste aspect van het probleem rondom geometrische vervorming. We presenteren 2 benaderingen om het probleem met beeld en video watermerken op te lossen:

- ***Verwijdering van de afhankelijkheid van spatiële synchronisatie van het watermerk.*** Wij stellen een watermerk systeem voor gebaseerd op een gestructureerd ruispatroon, dat geen strikte synchronisatie tussen het watermerk en de watermerkdetector vereist. De werking van dit systeem is ongevoelig voor translatie en is meer robust tegen rotatie en schaling dan de klassieke op ruis gebaseerde systemen.
- ***Het gebruik van kenmerken van de oorspronkelijke data als referentiepunten om de vervorming te inverteren.*** We introduceren een aanpak om de geometrische vervorming te inverteren die schaalbaar is in complexiteit. We gebruiken kenmerken van de oorspronkelijke data als referentiepunten om de gewatermerkte data uit te lijnen en daarmee de geometrische vervorming ongedaan te maken.

Tenslotte bespreken we het probleem om het perceptuele kwaliteitseffect op geometrische vervorming in beelden te kwantificeren. Dit aspect van het probleem wordt nog niet uitgebreid in de literatuur bestudeerd en we hebben tot op heden geen objectieve maat om het perceptuele kwaliteitseffect van geometrische vervorming te meten. We stellen in hoofdstuk 5 een nieuwe methode voor om kwaliteit te meten gebaseerd op hypotheses, waarin de perceptuele kwaliteit van een geometrisch vervormd beeld afhangt van de homogeniteit van de vervorming. We evalueren de prestatie van dit systeem in hoofdstuk 6 door het te vergelijken met de resultaten van een gebruikerstest.

De resultaten, zoals in dit proefschrift beschreven, kunnen als volgt worden samengevat:

- Ontwikkeling van een algoritme dat geschikt is om MPEG-1/-2 video gecodeerd als ‘low bit-rate’ van een watermerk te voorzien (Hoofdstuk 3).
- Ontwikkeling van een watermerk programma, dat niet afhankelijk is van een strikte spatiële synchronisatie van beelden en video (Hoofdstuk 4).
- Ontwikkeling van een aanpak om geometrische vervorming in gewatermerkt beeld ongedaan te maken, die schaalbaar is in complexiteit (Hoofdstuk 4).
- Ontwikkeling van een methode voor een objectieve bepaling van de perceptuele kwaliteit van geometrisch vervormde beelden (Hoofdstuk 5 en 6).

ACKNOWLEDGEMENTS

Doing research for your Ph.D. degree is a very demanding work. When you are someone coming from a very different cultural background, you also have to adapt to a new and unfamiliar environment. This process is crucial to the success of your work and your mental well being. I feel lucky, however, to have met and worked with so many people that helped make this transition as smooth as possible.

First of all, I would like to thank my promotor, Inald Lagendijk, for believing in me even when things are not moving as smoothly as we wanted. I really enjoyed the informal, yet serious, atmosphere during our discussions. Thank you not only for the encouragement but especially for the honest criticism of me and my work.

I would like to thank my colleagues in the Information and Communication Theory group. All of you have contributed in some way to this thesis. However, I would like to mention a few names in particular. Annett, Jeanine, Anja, Sanne, Ben and Hans, thank you for your support and dedication. Andrei, Jesper and Cigdem, thank you for your invaluable inputs and interesting discussions. Peter Jan, it has been a pleasure working with you and again thank you for letting me include the results of our paper into this thesis. And for the rest of you (especially those volunteering for the torture sessions in the dungeons of the ITS building) even though I do not mention your names here, your contribution to my work is bigger than you may have thought.

During my involvement with the European Community Project CERTIMARK, I have also met many people whose contributions to my research work have been invaluable. In particular, I would like to extend my gratitude to Damien Delannay and Benoit Macq from UCL and Ton Kalker an Job Ostveen from Philips Natlab for their ideas and collaboration.

I would also like to thank my parents and my brother for their unending moral support. I want to also thank my friends at the GKIN (Gereja Kristen Indonesia Nederland) for being my “extended family” during my stay in The Netherlands.

Last, but definitely not least, I would like to thank Christina for being always by my side. You are, above all, the one who makes all of this worthwhile.

CURRICULUM VITAE

Iwan Setyawan was born in Semarang, Central Java, Indonesia. He spent his childhood and the first part of his education in the smaller, cooler city south of Semarang called Salatiga. After finishing High School, he went to the Bandung Institute of Technology to work for his Bachelor's degree in Electrical Engineering (majoring in Computer Engineering). He received his B.Sc. degree in 1996 and continued his education in the same institution to earn his Master's degree in Electrical Engineering (majoring in Information Systems) in 1999.

In November 1999, he started to work towards his Ph.D. degree as a Research Assistant at the Delft University of Technology. His main research subject was watermarking digital image and video data. This work was performed in part within the scope of the European Community project CERTIMARK. His research interests includes watermarking low bit-rate video data and the problem of geometric distortions in images and video. In particular, he is interested in developing a system to objectively measure the perceptual quality impact of geometric distortions.

When not busy doing his research and teaching duties, he likes to pretend himself a dashing combat pilot strapped on his F-16, a special forces soldier carrying his trusty MP5 or a Field Marshall plotting brilliant tactical maneuvers to capture strategic victory hexes.