國立交通大學

資訊工程學系

博士論文

網路多媒體數位浮水印及其應用之研究

A Study on Digital Watermarking and Its Application on Network Multimedia

研究生：陳岳宏

指導教授：傅心家　教授

中華民國九十七年七月

網路多媒體數位浮水印及其應用之研究
A Study on Digital Watermarking and Its Application on Network Multimedia

研 究 生：陳岳宏　　Student：Yueh-Hong Chen
指導教授：傅心家 教授　　Advisor：Prof. Hsin-Chia Fu

國 立 交 通 大 學
資訊工程學系
博 士 論 文

A Dissertation
Submitted to Department of Computer Science
College of Computer Science
National Chiao Tung University
in Partial Fulfillment of the Requirements
for the Degree of
Doctor of Philosophy
in

Computer Science

July 2008
Hsinchu, Taiwan, Republic of China

中華民國九十七年七月

誌謝

本篇論文得能順利完成，首先要感謝我的指導教授—傅心家教授的辛勤指導，及不斷的鼓勵。其次要感謝論文的口試委員：陳正教授、林正中教授、詹寶珠教授、陶金旭教授、孫光天教授，以及江政欽教授，對本篇論文所提出的各項建議，不僅使這篇論文更完備，也讓我獲益良多。另外，要感謝在我求學之路上一路相伴的好朋友：實驗室學長徐永煜博士，以及學弟學妹們，與你們的相互切磋砥礪，是我最美好的回憶之一。最後要感謝多年來一直支持我的家人與女友，有你們的關心、鼓勵與包容，才有今日的我，謝謝你們。

陳岳宏

九十七年夏于交大

網路多媒體數位浮水印及其應用之研究

學生：陳岳宏　　　　　　　　　　　　指導教授：傅心家教授

國立交通大學資訊工程學系博士班

摘　　要

在本論文中，我們提出了一種用於影像的浮水印技術以及一種用於視訊的指印技術，並將之應用於一個以網頁為基礎的多媒體新聞資料庫系統。由於影像的漸進傳輸在許多應用中均相當重要且常用，特別是在網際網路上傳輸影像時。另一方面，複製及傳輸數位影像變得容易，使得利用數位浮水印技術來嵌入版權資訊的需求應運而生。因此，在本論文中，我們提出一個用於漸進影像傳輸的浮水印方法。此方法可以在影像仍在傳輸的過程中，即可偵測部份的浮水印資訊。隨著傳輸的進行，所取出的浮水印，其錯誤的位元數也會逐漸減少。此方法不僅能在影像傳輸的過程中，漸進地偵測浮水印，也會針對不同的係數大小，適當的選取修改值來嵌入浮水印。因此能有效降低嵌入浮水印後影像品質上的損失。

我們也提出了一個新的視訊加密與指印嵌入方法。此方法包含兩部份：(1) 伺服器端視訊加密，及 (2) 客戶端指印嵌入。首先，伺服器將一段視訊加密並以群播方式傳送給所有客戶端。接著，利用(v, k, 1)-BIBD 來將解密金鑰分解成 v = O($\sqrt{n}$)把解密子金鑰，並群播給 n 個客戶端。在收到各自的解密金鑰後，每個客戶端便可組合他們各自的解密子金鑰，得到一把嵌入指印後的解密金鑰。當使用這把金鑰來解開加密的視訊後，便會在視訊中嵌入一個對應於該客戶端的指印。通常嵌入指印會使視訊中產生若干雜訊。由實驗結果發現，當指印由 15 個以下的浮水印組成，或浮水印強度小於 0.5 時，畫面的訊噪比(PSNR)即高於 35db。故本技術極適合用於網路環境中的視訊著作權保護。

最後，我們整合數種多媒體資訊擷取技術，建立了一個多媒體新聞網站。其中所使用到的技術包括針對新聞標題、主播與過場，進行語音、影像與視訊分析，以建立一個有組織、有意義的新聞視訊摘要。讓使用者可以利用非循序、快速而具彈性的方式來瀏覽大量的新聞資料。藉由聲音分析，我們可以將新聞節目分成新聞片段與廣告片段。利用 2003 年七月至 2004 年 8 月間的 400 小時電視新聞做為實驗資料發現，本方法具有 96%的正確率。我們並使用 OCR 技術來辨視銀幕上的新聞標題或子標題，做為每段新聞故事的文字資訊。此外，所抽取出的新聞標題也可以用在網路上的相關新聞的連結與導覽。藉由整合人臉與場景分析與辨識技術，OCR 的結果可用來提供使用者針對特定新聞的多種查詢。

我們也將所發展的浮水印與指印技術實際用於所建立的多媒體新聞資料庫中。在關鍵畫面的抽取過程中，浮水印系統會為每個畫面嵌入浮水印以保護內容擁有者的著作權。此外，換鏡偵測的結果可以提供給指印系統，做為更新客戶端指印的時間點，以增強所嵌入指印的強固性。在多媒體的應用中，演算法的效能是一項重要的課題。由測試結果發現，我們所提出的浮水印與指印方法，均極適合於網路上的各種多媒體應用。

A Study on Digital Watermarking and Its Application on Network Multimedia

student：Yueh-Hong Chen Advisors：Prof. Hsin-Chia Fu

Department of Computer Science
National Chiao Tung University

ABSTRACT

In this dissertation, we propose a progressive image watermarking scheme and a video fingerprinting scheme for copyright protection of multimedia applications on the Internet. The ease of transmission and copying of images creates the need to use digital watermarking to embed the copyright information seamlessly into the media. On the other hand, progressive transmission of images is very useful in many applications, especially in image transmission over the Internet. Since the progressive image transmission has been widely used, in this dissertation, we first propose a progressive image watermarking scheme. In this scheme, the watermark is embedded in such a way that we can retrieve part of it even when the watermarked image is still being transmitted. As transmission progresses, the retrieved watermark has a decreasing bit error rate. Our proposed method can not only detect the watermarked image progressively, but also intelligently select watermark embedding locations and is robust to various attacks.

We also propose a new video scrambling and fingerprinting approach for digital media right protection. The proposed method contains two parts: (1) video scrambling at server side, and (2) fingerprint embedding at client side. First, a content server scrambles and multicasts video contents to end users. Then, by applying a (v, k, 1)-BIBD scheme, the server partitions a descrambling key into v=O($\sqrt{n}$) descrambling subkeys, and multicasts to n users. On receiving descrambling subkeys from the content server, each user combines descrambling subkeys into a descrambling key embedded with a fingerprint. By using he's descrambling key, a scrambled video becomes a fingerprinted video designated to the user.According to the experiment results, when the fingerprint consisting of less than 15 watermarks or watermark strength α s less than 0.5 the PSNR of video frames can be 35 or higher. Thus, this approach is suitable for multimedia applications over The Internet.

Finally, an integrated information mining techniques for multimedia TV-news archive is addressed. The utilized techniques from the fields of acoustic, image, and video analysis, for information on news story title, newsman and scene identification. The goal is to construct a compact yet meaningful abstraction of broadcast news video, allowing users to browse through large amounts of data in a non-linear fashion with flexibility and efficiency. By using acoustic analysis, a news program can be partitioned into news and commercial clips, with 90% accuracy on a data set of 400 hours TV-news recorded off the air from July 2003 to August of 2004. By applying speaker identification and/or image detection techniques, each news stories can be segmented with an accuracy of 96%. On screen captions or subtitles are recognized by OCR techniques to produce the text title of each news stories. The extracted title words can be used to link or to navigate more related news contents on the WWW. In cooperation with facial and scene analysis and recognition techniques, OCR results can provide users with multimodality query on specific news stories. Some experimental results are presented and discussed for the system reliability and performance evaluation and comparison.

The proposed web based TV news archive was also used as a test bed of the proposed image watermarking and video fingerprinting methods. Watermarks can be embedded automatically when key frames of a news story are extracted, and information obtained from video analysis process can be used to increase the robustness of video fingerprinting. In multimedia applications, computational efficiency is one of the important issues. Our testing shows that the proposed watermarking and fingerprinting methods are efficient to real world applications.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Motivations

The growth of high speed computer networks and that of Internet, in particular, has explored means of new business, scientific, entertainment, and social opportunities in the form of electronic publishing and advertising, real-time information delivery, product ordering, transaction processing, digital repositories and libraries, web newspapers and magazines, network video and audio, personal communication, lots more. The new opportunities can be broadly grouped under the label ”electronic commerce” . The cost effectiveness of selling software, high quality art work in the form of digital images and video sequences by transmission over World Wide Web (www) is greatly enhanced consequent to the improvement of technology. Sending hard copies by post may soon be a thing of past.

Though the commercial exploitation of the www is steadily being more appreciated, apprehension on the security aspect of the trade has only funneled the exploitation to be restricted to the transmission of demo and free versions of software and art. Ironically, the cause for the growth is also of the apprehension-use of digital formatted data.

Digital media offer several distinct advantages over analog media: the quality of digital audio, images and video signals are higher than that of their analog counterparts. Editing is easy because one can access the exact discrete locations that should be changed. Copying is simple with no loss of fidelity. A copy of a digital media is identical to the original.

The ease by which a digital information can be duplicated and distributed has led to the need for effective copyright protection tools. Various software products have been recently introduced in attempt to address these growing concerns. It should be possible to hide data (information) within digital audio, images and video files. The information is hidden in the sense that it is perceptually and statistically undetectable.

One way to protect multimedia data against illegal recording and retransmission is to embed a signal, called digital signature or copyright label or watermark [1, 2], that completely characterizes the person who applies it and, therefore, marks it as being his intellectual property.

However, most of digital watermarking methods prevent illegal use or spreading digital contents by passively checking whether the user owns lawful ownership. Thus, if a method that can detect the source of illegal distribution actively, so as to stop the transmission of unauthorized digital media will be a much better digital right protection approach. In other words, the system can actively poll other networks (e.g., underground media distribution centers) to determine if they own the rights to circulate the content. If a watermark is detected, the system will notify the content owner, and thus creating a natural deterrent to piracy.

When an embedded watermark is associated with a particular user, it can be considered as a fingerprint. Once a fingerprinted media has been illegally distributed, the original user could be easily traced from redistributed version. Unfortunately, fingerprint embedding process often makes a signal media copy into many different versions, which have to be transmitted via unicast method. Generally, it is more efficient to transmit a single media via multicast method to massive users. It becomes quite important in the field of video-based applications to transmit video data embedded with fingerprints efficiently to all the users. Thus, how to allow most of video content to be transmitted via multicast and only the very little fingerprint related contents are transmitted via unicast have been an important issue to network multimedia applications.

1.2 Research goals

In this dissertation, we first proposed a robust watermarking scheme for digital images. Then, we study the properties of a watermarking scheme that can be used in the active watermark detection system discussed above, and propose a progressive image watermarking method. To decrease the bandwidth required for delivering fingerprinted video, we also propose a new video scrambling and fingerprinting approach for digital media copyright protection. Finally, as an application of the proposed watermarking and fingerprinting methods, a WWW-based multimedia TV-news archive, which integrates text, image and video data is addressed.

1.3 Dissertation organization

In the rest of this dissertation, survey of various image and video watermarking systems are presented in Chapter 2. In Chapter 3, we present an adaptive watermarking scheme for images. The progressive watermarking scheme for image data is discussed in Chapter 4. In Chapter 5, the problem of embedding a fingerprint for each multicast client is addressed. Then, two methods are proposed respectively to reduce the bandwidth required by multicast server when video clips with fingerprints are transmitted to clients. In Chapter 6, we present a web-based news archive system in which the proposed image and video watermarking schemes were used. Finally, conclusions and suggestions appear in Chapter 7.

Chapter 2

Background

2.1 A Brief introduction of digital watermarking

The watermark can be regarded as an additive signal W, which is a binary string or a sequence of independent Gaussian random numbers. Although a watermark consisting of Gaussian random numbers is more secure, binary watermark can be used to represent a user ID or logo. To achieve a perceptually indistinguishable watermarked and original signal, a watermarking approach usually keep the power of the watermark signal very low. Commonly used embedding techniques can be classified into *additive*, *multiplicative*, *quantization-based*, and *relationship-based* schemes. In additive schemes, a very weak W is added into original signal x, as shown in Eq. 2.1

$$Y = X + \alpha W, \tag{2.1}$$

where X is the original signal, Y is the watermarked signal and α is a constant, referred to as *watermark strength*. In multiplicative schemes, samples of the original data are multiplied by an independent signal $(1 + \alpha W)$. Precisely, multiplicative schemes can be described by Eq. 2.2:

$$Y = X \times (1 + \alpha W). \tag{2.2}$$

In quantization based watermarking schemes, X is modified such that the quantization indices imply a watermark for a certain quantization step q. For example, a binary watermark W can be embed into the signal X with following Eq. 2.3:

$$\begin{cases} \lfloor \frac{Y}{q} + 0.5 \rfloor \bmod 2 = 0 & \text{if} \quad W = 0 \\ \lfloor \frac{Y}{q} + 0.5 \rfloor \bmod 2 = 1 & \text{if} \quad W = 1 \end{cases} . \tag{2.3}$$

In this example, signal X is modified into Y such that its quantization index is an even number to imply a binary value '0', and vice versa.

The basic idea of relationship-based watermarking is to use two pixel values or transform domain coefficients in a image to represent each bit of a binary watermark. For example, if the value of the first coefficient is larger than the second, then an '1' is encoded; otherwise, a '0' is encoded [3, 4]. Some watermarking approaches use more than two coefficients for each watermark bit to increase the robustness. Typically, these approaches adequately modify the selected coefficients such that a pre-specified one among them becomes smallest or largest, according to the binary value to be embedded [5, 6, 7].

2.2 Progressive transmission of images on Internet

Generally, the regular scanning pattern is used in image transmission on Internet. In other words, image pixels are displayed from left to right and top to bottom. Often an user can recognize the image and decide whether or not to download the whole image only when a substantial portion of data has been transmitted. Considering that significant amounts of data are needed to represent large digital images, image transmission across low-bandwidth channels can be exceedingly slow. Thus, a better solution is to improve the transmission method so that the image is transmitted progressively. Progressive image transmission makes effective use of communication bandwidth. Instead of transferring an image at full resolution sequentially, progressive transmission first transmits an approximate version of the entire image so that its structural information is sent early in the transmission. The quality of this image is progressively improved over a number of transmission passes. The advantage is that it allows the user to quickly recognize an image. Two classes of progressive transmission methods are commonly used and briefly introduced in this section: successive approximation methods and transmission sequence-based methods.

2.2.1 Transmission-sequence based methods

A progressive transmission method using the transmission-sequence (TS) based approach is composed of a classifier that separates the image data into different transmission groups, a mechanism for ordering the groups, and a method for specifying the ordering. Generally, an increase in the number of groups improves the buildup quality but involves a higher overhead cost. The approach is adopted by progressive JPEG compression, and is referred to as *spectral selection mode* in progressive JPEG. In spectral selection mode, JPEG encoder takes advantage of the spectral (spatial frequency spectrum) characteristics of the DCT coefficients: the higher AC components provide only detail information. So, the order of coefficients to be transmitted may be as follows:

Group 1: Encode DC and first few AC components, e.g., AC1, AC2.

Group 2: Encode a few more AC components, e.g., AC3, AC4, AC5.

...

Group k: Encode the last few ACs, e.g., AC61, AC62, AC63.

2.2.2 Successive approximation methods

In successive approximation methods, progressive transmission is achieved by refining the precisions of the coded data at each transmission stage. In the initial stage, a low-precision form of the data is sent. In subsequent stages, the precision is gradually increased until the full precision of the coded data at the final stage. This approach is adopted by both JPEG and JPEG2000 standards. Taking JPEG as an example, instead of gradually encoding spectral bands, all DCT coefficients are encoded simultaneously, but with their most significant bits (MSBs) first.

Scan 1: Encode the first few MSBs, e.g., bits 7, 6, 5, and 4.

Scan 2: Encode a few more less-significant bits, e.g., bit 3.

...

Scan m : Encode the least significant bit (LSB), bit 0.

Successive approximation used in JPEG2000 is similar to that in JPEG, but a more efficient bit-plain encoding method is used.

2.3 Fingerprinting methods for multicast video

2.3.1 Independent Gaussian fingerprints and c-secure code

Fingerprinting is an useful technique because the original user could be easily traced from redistributed version. However, fingerprints could be identified and removed by comparing certain amount of media versions, which are embedded with various fingerprints. This kind of attack is often referred to as *collusion attack*. Several studies [8, 9, 10] have been proposed to addressing collusion-resistant ability among fingerprint approaches. In [8], Su et al., showed that fingerprints constructed from independent Gaussian watermarks required shorter fingerprint sequence than fingerprints constructed from c-secure code [9, 10] did. Moreover, [8] indicated that in the circumstance of a simple linear collusion attack consisting of adding noise to the average of several fingerprinted copies, no other watermarking schemes could offer better collusion resistance than independent watermarks. However, embedding independent watermarks into a video as fingerprints often makes a single video into a large amount of different copies, even though each copy is only slightly different from each other. Thus, video contents embedded with independent watermark based fingerprints can hardly be transmitted by multicast methods.

2.3.2 Performance metric for multicast fingerprinting schemes

Multicast transmission described in this paper is similar to that in [11]. First, we assume that there is only a public channel between a media server and all clients. Data transmitted with the public channel can be received by all clients simultaneously. In other words, sending data directly with the public channel is *broadcast transmission*. If the server needs to send secret data to a specific client, the data should be encrypted with the client's secret key before transmission. All clients' secret keys are delivered with via a secure channel. Encrypting and transmitting data to a specific client is referred to as *unicast transmission*. We use the term *multicasting* for the transmission of data using both the unicast and broadcast methods. Qualitatively, the transmission of media content is efficient if it incorporates both the broadcast and unicast methods such that the broadcast channel is used a few times, while the unicast channel is seldom employed. Quantitatively, the efficiency of a distribution method is measured relative to the purely naive broadcasting scenario and can be defined by the ratio given in Eq. 2.4 [11],

$$\eta_D := \frac{m_D}{m_0}, \tag{2.4}$$

where m_D is a value proportional to the bandwidth used by a fingerprinting scheme, and m_0 is a value proportional to the bandwidth used in the unicast channel case. In particular, m_0 is defined to be the number of times the public channel is used when the fingerprinted content is sent to each user respectively, and m_D is the number of times the public channel is used by the fingerprinting scheme. We expect η to be between 0 and 1. In addition, for a fingerprinting method 1 that is more efficient than a fingerprinting method 2, we have $\eta_1 < \eta_2$.

Chapter 3

Adaptive Watermarking Using Relationships between Wavelet Coefficients

3.1 Introduction

In this Chapter, we propose an image watermarking approach for the purpose of proving ownership. This approach hides watermarks in relationships between wavelet coefficients and afterward detects the watermarks blindly.

The rest of the chapter is organized as follows. A brief review of those research efforts is described in Section 3.2. Section 3.3 introduces the assumption in this chapter and presents the extreme value based watermarking approach. The experimental results are shown in Section 3.4. Finally, Section 3.5 summarizes our approach and provides a brief concluding remarks.

3.2 Related works

As described in Section 2.1, the relationship-based watermarking is to use two pixel values or transform domain coefficients in a image to represent each bit of a binary watermark. If the first value is larger than the second, then an '1' is encoded; otherwise, a '0' is encoded. Based on this idea, several watermarking methods have been proposed to use relationships between pixels or transform domain coefficients. In [3], Koch and Zhao proposed to group selected coefficients of an 8×8 DCT block in the image into ordered pairs. Each bit of the watermark is then encoded using one of the coefficient pairs. However, there was no experimental result showing the robustness and imperceptibility of the watermark in the paper. Hsu and Wu [4] proposed an approach using middle frequency coefficients chosen from one or more 8×8 DCT blocks to embed watermarks. Quantization operation is taken into account in this approach so that watermarks can survive the JPEG lossy compression. For watermark extraction, the original image and the watermark used during the embedding step are required. They are, however, unavailable in some applications such as copy control.

Some watermarking approaches use more than two coefficients for each watermark bit to increase the robustness. Typically, these approaches adequately modify the selected coefficients such that a pre-specified one among them becomes smallest or largest, according to the binary value to be embedded. In [5], three coefficients selected from an 8×8 DCT block are altered to meet the situations in which the third coefficient is largest or smallest, in accordance with the watermark bit to be embedded. However, the extractor proposed in [5] ignores the three-coefficient sets in which the third coefficient is between other two. As a result, a failure of detecting one single bit may make the whole watermark undetectable. A similar approach in [6] selects six coefficients from a DCT block and then exchanges the first coefficient with the largest or smallest coefficient among them. Nevertheless, a significant degradation in image quality may be caused if the difference between two coefficients to be exchanged is large. A closely related approach was proposed in [7]. This approach divides the pixels of an image into two groups. The sum of the pixels in one group is subtracted from the sum of the pixels in the other group to obtain a detect statistic, which is then compared against a certain threshold to determine whether the watermark is present. However, [7] did not use a human vision system (HVS) model to increase imperceptibility of watermarks.

Intuitively, the distortion caused by watermark embedding operation should be minimized in the sense of an appropriate distortion metric. Thus, an extreme value based watermarking (EVBW), using the relationship between wavelet coefficients, is proposed. When embedding watermarks, the EVBW will adaptively minimize the distortion, measured with PSNR or just noticeable distortion (JND). By minimizing the images distortion, the strength of watermarks can be enlarged to increase the robustness of the watermarks, while keeping the quality of watermarked images visually accepted. The experimental results shown in section 3.4 illustrate the performance of the proposed approach.

3.3 Extreme-value based watermarking

It is assumed in this chapter that a watermark is a binary string consisting of two symbols, -1 and 1. All bits of the watermark are embedded into an image with the same manner, separately. The manner to represent a binary string using relationships of wavelet coefficients is first introduced in this section. An overview of EVBW is then proposed in Subsection 3.3.2. The embedding algorithm of EVBW is discussed in detail in Subsection 3.3.3.

3.3.1 Hiding binary string into relationships of wavelet coefficients

To embed a watermark, an image is firstly transformed into Haar wavelet domain. For each bit of the watermark, a number of coefficients in pre-specified subband (e.g., $LH2$, $HL2$ or $HH2$) are then randomly chosen and modified. Finally, inverse wavelet transform is applied to obtain the watermarked image.

When one bit of the watermark is to be embedded, an user-specified number of coefficients are chosen randomly. These coefficients are then modified such that the first coefficient, in the order of being chosen, becomes the largest one if an '1' is to be embedded. If a '-1' is to be embedded, the coefficients should be modified such that the first coefficient becomes the smallest one. Suppose c_i, $i = 1, \cdots, n$, are the chosen coefficients, n is the number of chosen coefficients, $W = \{w_l | w_l \in \{1, -1\}, 1 \leq l \leq L\}$ is the watermark to be embedded, and L is the length of the watermark W. Precisely, after modification step, the relationship behind the coefficients is as equation (3.1)

$$\begin{cases} c_1^{'} \geq \max(c_2^{'}, c_3^{'}, \cdots, c_n^{'}) + \delta & \text{if } w_l = 1 \\ c_1^{'} < \min(c_2^{'}, c_3^{'}, \cdots, c_n^{'}) + \delta & \text{if } w_l = -1 \end{cases} \tag{3.1}$$

where $c_i^{'}$, $i = 1, \cdots, n$ are the modified coefficients, w_l is a particular bit of the watermark code, and δ, $\delta \geq 0$, is the strength parameter specifying the difference between the first coefficient, $c_1^{'}$, and the largest (smallest) one among remaining coefficients. Intuitively, the larger the value of δ, the more robust the watermark. However, the perceptual fidelity of the watermarked image will decrease when a larger δ is adopted. Thus, for different applications, the value of δ should be specified by the user.

To clarify the description, a simple example of implying a watermark bit with coefficients is given. Suppose an '1' is to be embedded and five coefficients, -5, 112, -1, 107 as well as 13, are chosen randomly. A straightforward manner is to increase the value of the first coefficient, -5, to a value equal to or larger than 112, and other coefficients are left unchanged. By this manner, the first coefficient, -5, should be increase to $112 + \delta$ to embed an '1'.

The simplest watermark extracting method is to pick up the same coefficients and determine if the first coefficient is the largest (smallest) one. However, the watermarked image may be distorted due to some image-processing operations, and the first coefficient is possibly no longer the largest (smallest) one. Hence, the purposed extracting method is to compare the first coefficient with the largest and smallest ones among remaining coefficients. If the value of the first coefficient is closer to the largest one among remaining coefficients, an '1' will be extracted; otherwise, a '-1' will be extracted. This method can be described as equation (3.2):

$$w_l^{'} = \begin{cases} 1 & \text{if } c_1^{''} \geq \frac{c_{max}^{''} + c_{min}^{''}}{2} \\ -1 & \text{otherwise} \end{cases} \tag{3.2}$$

$$\begin{aligned} c_{max}^{''} &= \max(c_2^{''}, c_3^{''}, \cdots, c_n^{''}) \\ c_{min}^{''} &= \min(c_2^{''}, c_3^{''}, \cdots, c_n^{''}) \end{aligned} \tag{3.3}$$

where $c_i^{''}, i = 1, \cdots, n$ are the coefficients obtained from an image to be judge, and $w_l^{'}$, $1 \leq l \leq L$, is the extracted binary value. The percentage of matching bits between the extracted binary string W', $W' = \{w_l^{'} | w_l^{'} \in \{1, -1\}, 1 \leq l \leq L\}$, and the watermark W is then calculated. Finally, the percentage of matching bits between W' and W is compared with a certain threshold to determine if the watermark exists or not.

3.3.2 An overview of EVBW

The main idea of EVBW is to modify more than one coefficient at the same time. To embed an '1', if the first coefficient c_1 is increased to $x + \delta$, all coefficients larger than x should be decreased to x to fit the rule shown in equation (3.1). Therefore, it is possible to find the optimal value of x such that the watermarked image have the best visual quality according to an appropriate quality metric.

One of the image quality metrics that are widely used is peak signal-to-noise ratio (PSNR). Based on the definition of PSNR, if the mean square error (MSE) of the modified coefficients is minimized, the PSNR value is maximized simultaneously. Suppose a bit '1' is to be embedded into n coefficients. If c_1 is increased to $x + \delta$ and all coefficients larger than x are decreased to x, the square error (SE) value can be calculated as equation (3.4):

$$SE(x) = ((x + \delta) - c_1)^2 + \sum_{c_i > x} (c_i - x)^2 \tag{3.4}$$

Then the minimum of $SE(x)$ can be obtained by finding out the value of x where the first derivative of $SE(x)$ is equal to 0. The first derivative of $SE(x)$ is shown in equation (3.5), and the optimal value of x is shown in equaion (3.6).

$$\frac{d}{dx} SE(x) = 2 \times (x + \delta - c_1) + 2 \times \sum_{c_i \in M(x)} (x - c_i) \tag{3.5}$$

$$x = \frac{\left(\sum_{c_i > x} c_i \right) + c_1 - \delta}{k + 1}, \tag{3.6}$$

where $M(x)$ is a set that consists of the coefficients larger than x except c_1, i.e., $M(x) = \{c_i | c_i > x, 2 \leq i \leq n\}$ and k is the number of elements in $M(x)$. In equation (3.6), it is assumed that only k largest coefficient and c_1 be modified. Therefore, the value of

x should be larger than the $(k+1)$-th largest coefficient but smaller than k-th largest coefficient. Therefore, the algorithm to find the optimal value x is as follows:

```
Obtain $d_1, d_2, \cdots, d_n$ by sorting $c_1, c_2, \cdots, c_n$
    such that $d_1 \geq d_2 \geq \cdots \geq d_n$
Suppose $c_1$ is the $(k+1)$-th largest value
If $(k+1) = 1$
    $x_{opt} = d_2$, Stop
End If
For $i = 1$ to $k$
    $x = \frac{\left(\sum_{j=1}^{i} d_j\right) + d_{k+1} - \delta}{i+1}$
    If $d_{i+1} < x \leq d_i$
        $x_{opt} = x$, Stop
    End If
End For
$x_{opt} = c_1$, Stop
```

After the algorithm finishes, the optimal value of x can be found. c_1 can then be modified to $x + \delta$, and all coefficients larger than x be modified to x to embed a bit '1'. A similar algorithm to find the optimal value to embed a '0' is as follows:

```
Obtain $d_1, d_2, \cdots, d_n$ by sorting $c_1, c_2, \cdots, c_n$
    such that $d_1 \leq d_2 \leq \cdots \leq d_n$
Suppose $c_1$ is the $(k+1)$-th smallest value
If $(k+1) = 1$
    $x_{opt} = d_2$, Stop
For $i = 1$ to $k$
    $x = \frac{\left(\sum_{j=1}^{i} d_j\right) + d_{k+1} + \delta}{i+1}$
    If $d_{i+1} > x \geq d_i$
        $x_{opt} = x$, Stop
```

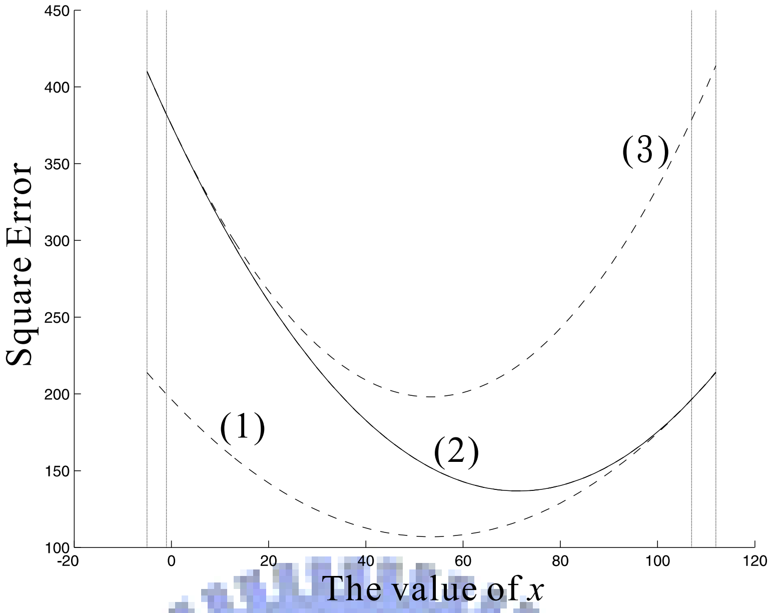


Figure 3.1: Curves of $(x-112)^2+(x+5)^2$ (curve 1), $(x-112)^2+(x-107)^2+(x+5)^2$ (curve 2) and $(x-112)^2+(x-107)^2+(x+1)^2+(x+5)^2$ (curve 3)

```
        End If
    End For
    $x_{opt} = c_1$, Stop
```

Finally, c_1 is decreased to $x-\delta$, and all coefficients smaller than x be increased to x to embed a bit '0'.

Continuing the example in previous subsection, if $\delta = 0$, the $SE(x)$ value is:

$$SE(x) = \begin{cases} (x-112)^2+(x+5)^2 \\ \quad \text{if } 107 \le x < 112 \\ (x-112)^2+(x-107)^2+(x+5)^2 \\ \quad \text{if } -1 \le x < 107 \\ (x-112)^2+(x-107)^2+(x+1)^2+(x+5)^2 \\ \quad \text{if } -5 \le x < -1 \end{cases}$$

By applying the proposed algorithm, the optimal value of x, about 71.3, can be obtained. The curve of $SE(x)$ is shown in figure 3.1. It is clear that SE(x) is the minimum when $x = 71.3$.

3.3.3 Perceptually adaptive embedding

Several researchers have indicated that PSNR may not be a ideal measurement of image quality [2]. Thus, the algorithm proposed in previous subsection is extended so that the watermark can be embedded based on JND.

A level of distortion that can be perceived in 50% experimental trials is often referred to as just noticeable difference, or JND [2]. Suppose a certain JND evaluation method in wavelet domain is used, and a JND value is always larger than zero. The perceptual distortion could be measure by weighting with the JND value the square error between original and modified coefficients. If a bit '1' is to be embedded into n coefficients, the weighted square error caused by watermark embedding process is:

$$\Delta(x) = \left(\frac{(x+\delta)-c_1}{J_1}\right)^2 + \sum_{c \in M(x)} \left(\frac{c-x}{J_i}\right)^2, \tag{3.7}$$

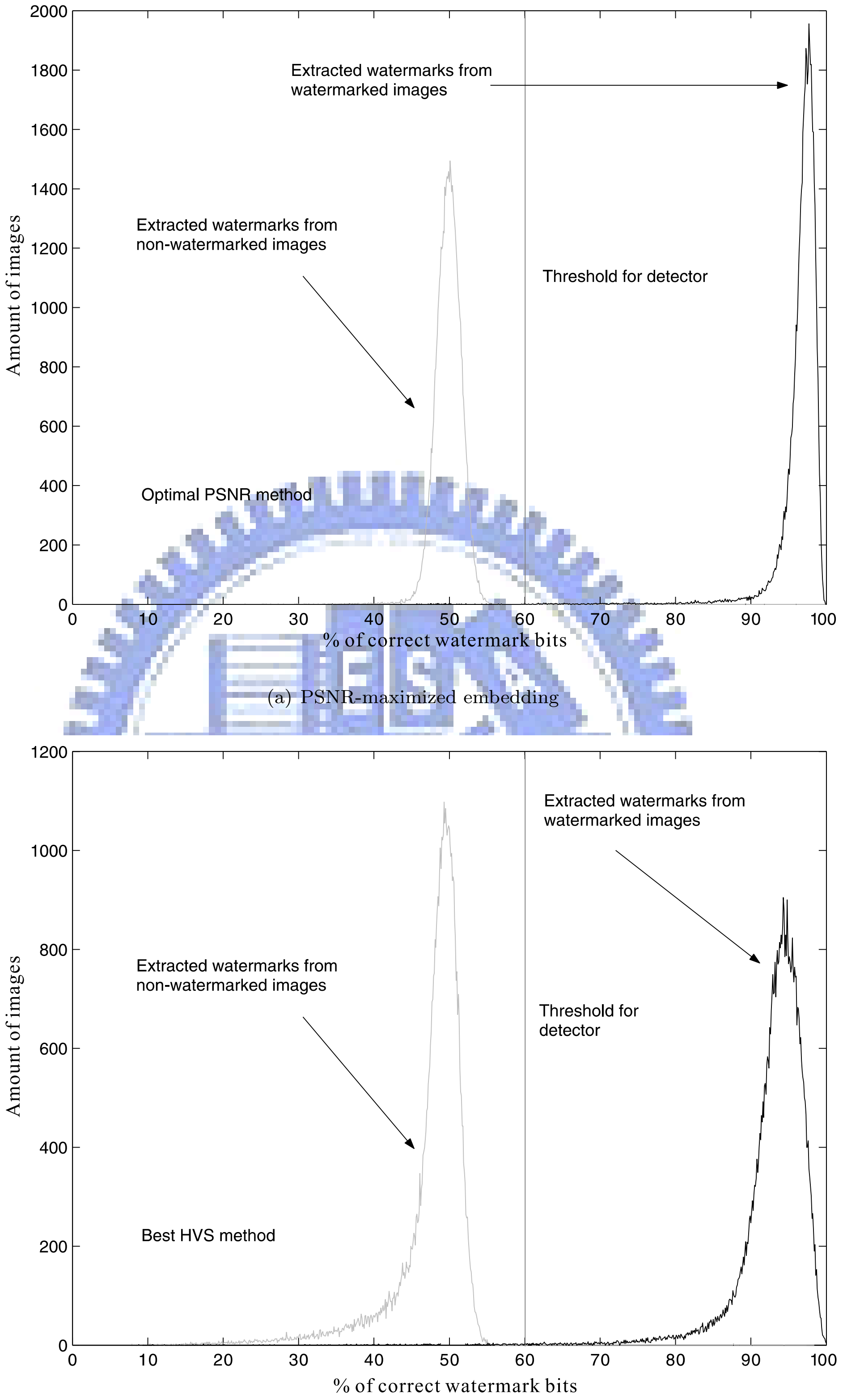
where J_i, $i = 1, \cdots, n$ is the JND value of c_i. The minimum of $\Delta(x)$ can then be obtained by finding out the value of x where the first derivative of $\Delta(x)$ is equal to zero. The first derivative of $\Delta(x)$ is shown in equation (3.8), and the optimal value of x is shown in equation (3.9).

$$\frac{d}{dx}\Delta(x) = \frac{2 \times (x+\delta-c_1)}{J_1^2} + 2 \times \sum_{c_i \in M(x)} \left(\frac{x-c_i}{J_i^2}\right), \tag{3.8}$$

$$x = \frac{\left(\sum_{c_i > x} \frac{c_i}{J_i^2}\right) + \frac{c_1}{J_1^2} - \frac{\delta}{J_1^2}}{\frac{1}{J_1^2} + \left(\sum_{c_i \in M(x)} \frac{1}{J_i^2}\right)}. \tag{3.9}$$

According to equation (3.9), an algorithm similar to that described in previous subsection can be used to find the optimal value of x for embedding watermarks.

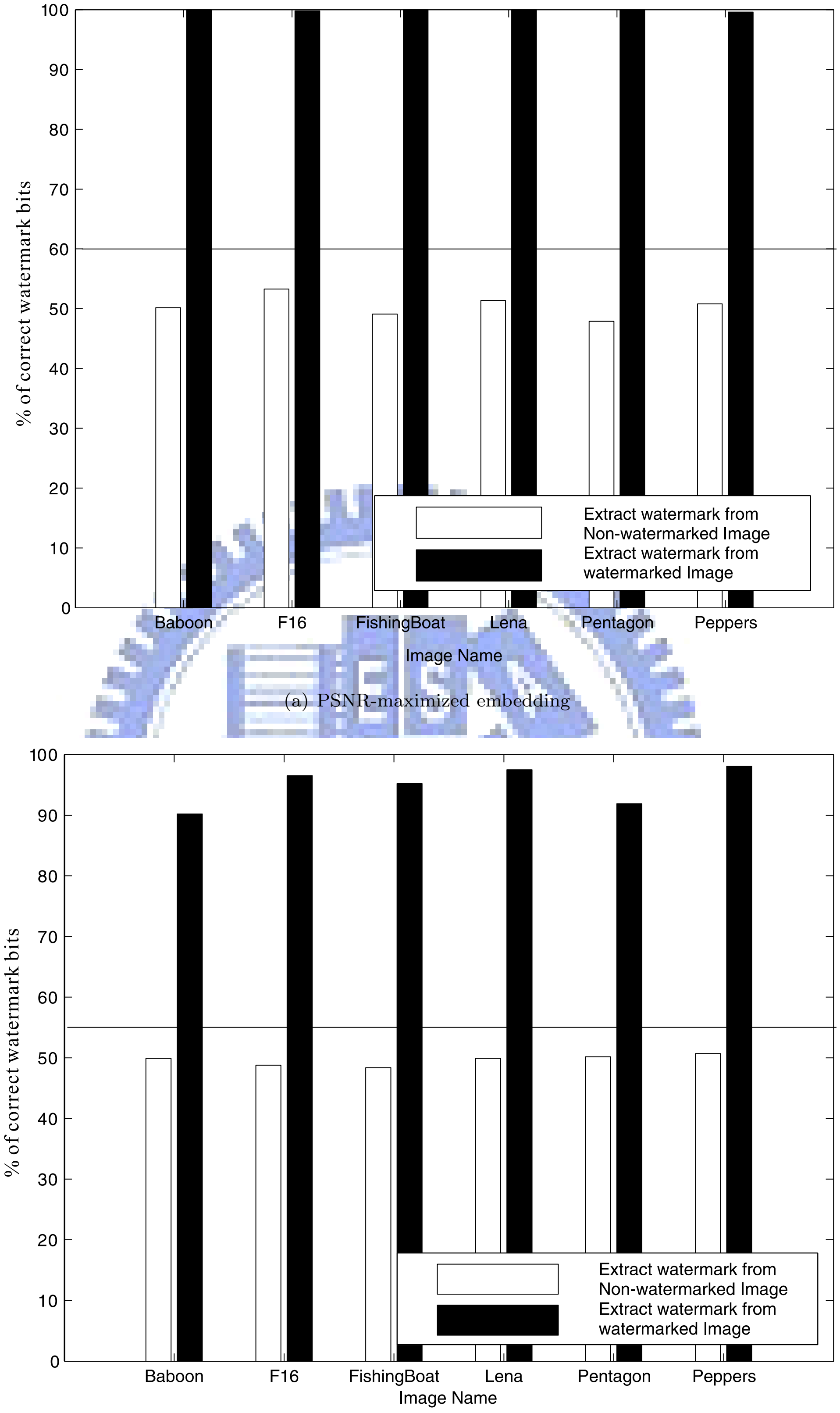
Intuitively, the extracting algorithm described in Section 3.3.1 can be used to extract watermarks embedded with PSNR-maximized or perceptually adaptive embedding algorithm. In the next section, several experimental results are presented to show the robustness and fidelity of the proposed algorithms.



(a) PSNR-maximized embedding

(b) perceptively adaptive embedding

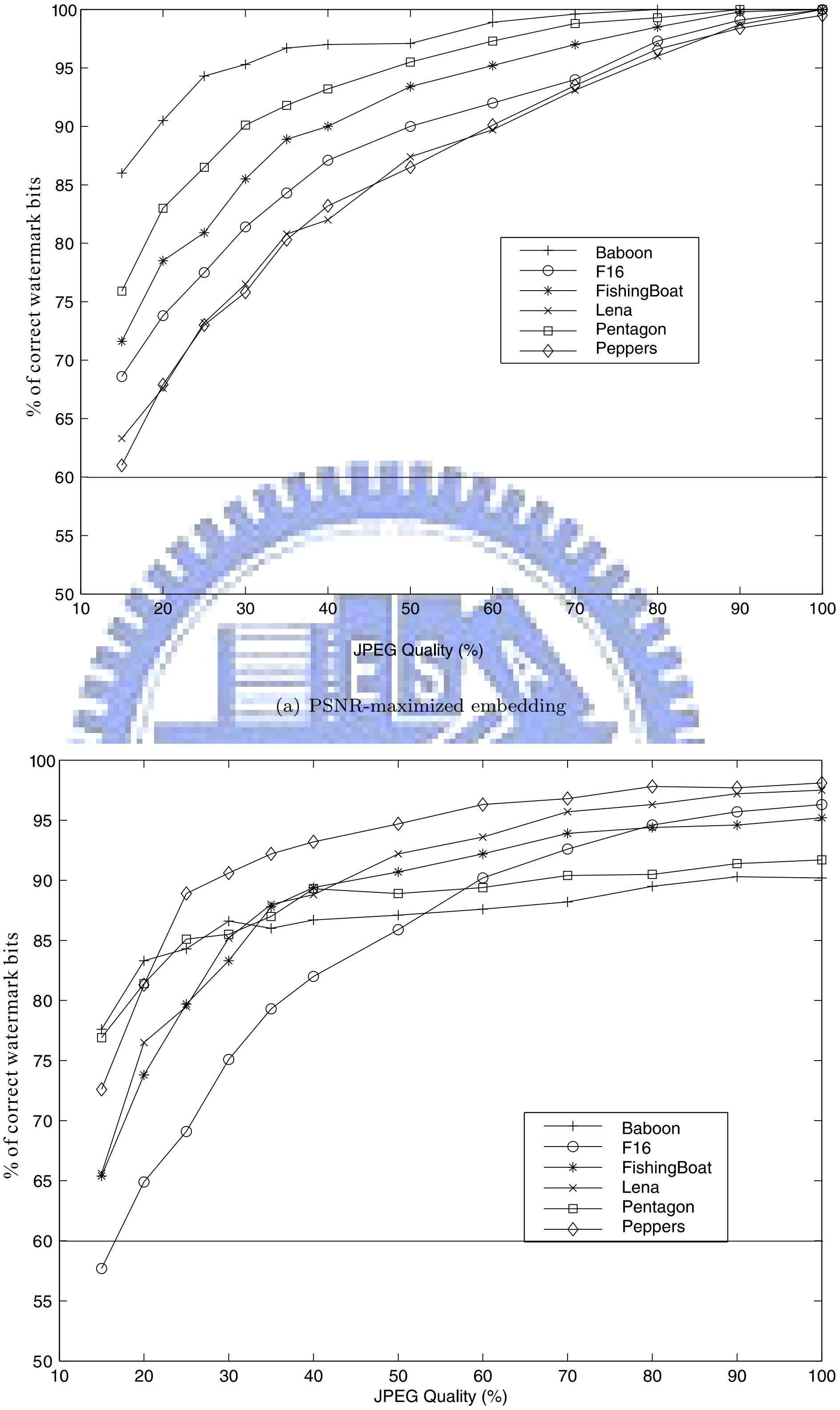
Figure 3.2: Results of applying watermark detection process to 58600 watermarked and non-watermarked images



(a) PSNR-maximized embedding

(b) perceptively adaptive embedding

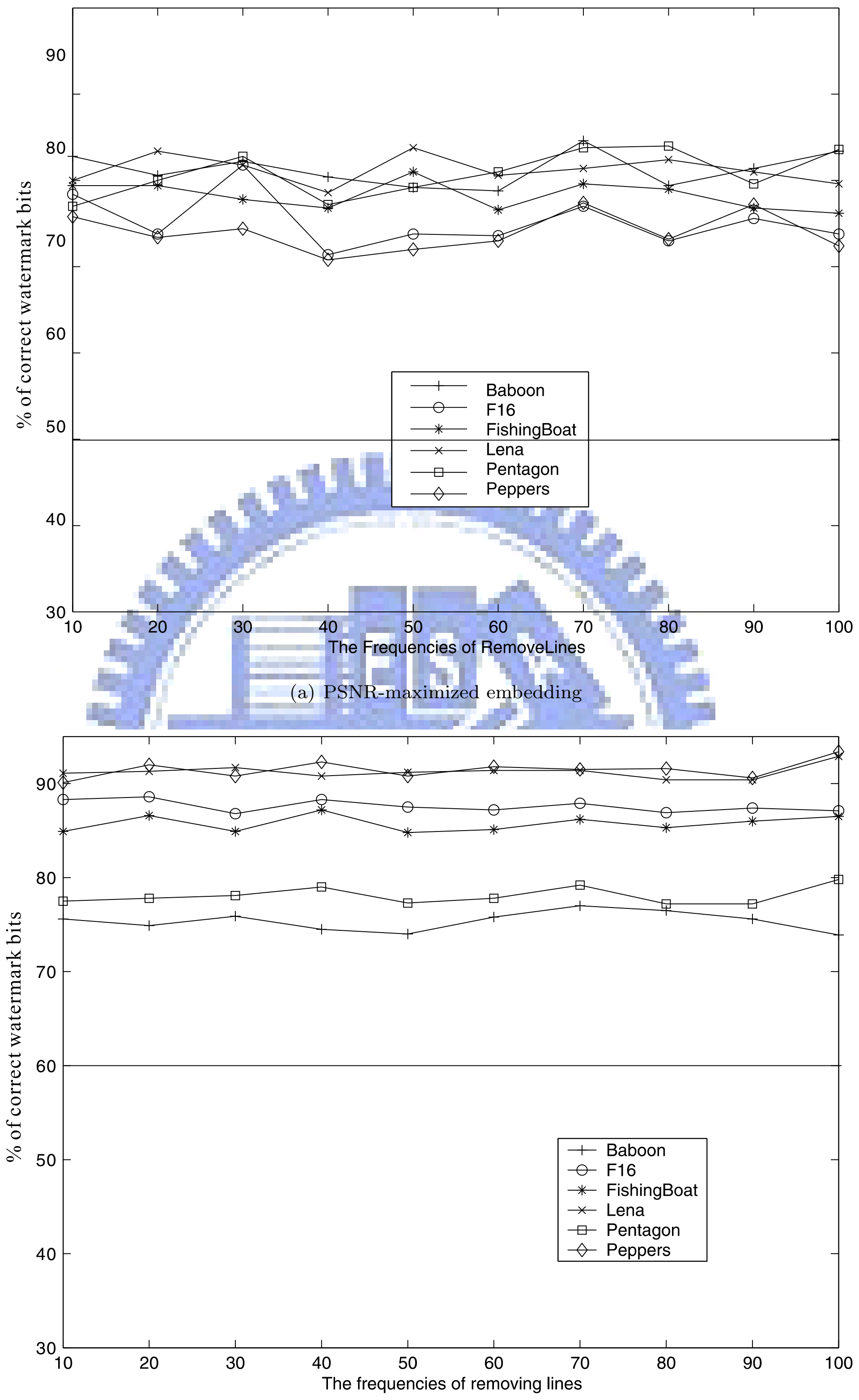
Figure 3.3: Results of applying watermark detection process to six watermarked and non-watermarked test images



(a) PSNR-maximized embedding

(b) perceptively adaptive embedding

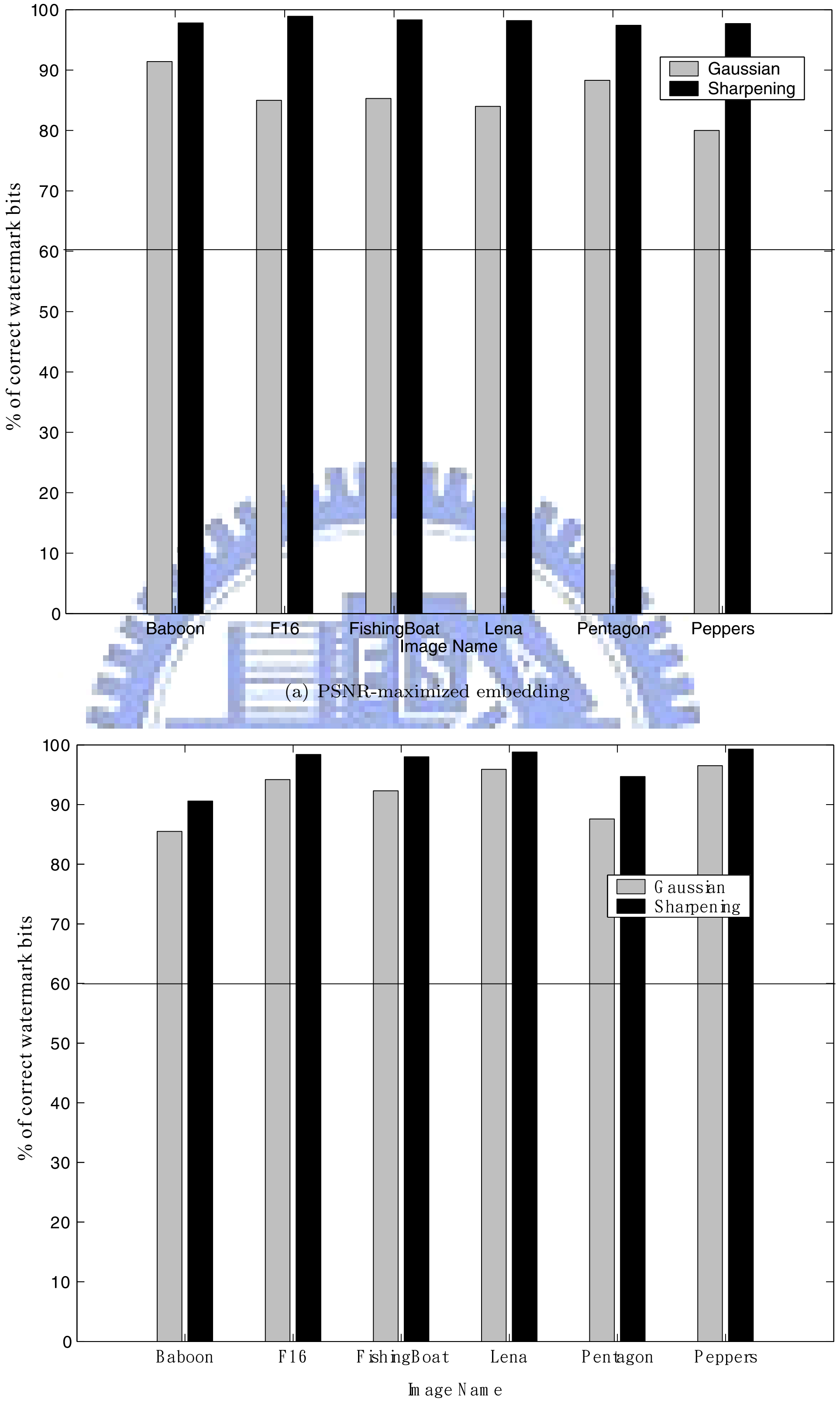
Figure 3.4: Results of watermark detection after JPEG compression



(a) PSNR-maximized embedding

(b) perceptively adaptive embedding

Figure 3.5: Results of watermark detection after line removing attack



(a) PSNR-maximized embedding

(b) perceptively adaptive embedding

Figure 3.6: Results of watermark detection after Gaussian filtering and sharpening

3.4 Experimental results

In this section, results of experiments using Strmark [12] are proposed to demonstrate the robustness of the proposed algorithms. An 1000-bit watermark was generated randomly and used throughout the experiments. To embed one bit of the watermark, twelve coefficients were chosen from $LH2$, $HL2$ or $HH2$ subband. In other words, n was equal to 12 in our experiments. The strength parameter δ was assigned to 0. The method proposed in [13] was used to evaluate JND values in perceptually adaptive embedding algorithm.

To determine the threshold for watermark detection process, 58600 images chosen from Corel Gallery 1000000 were watermarked using PSNR-maximized embedding and perceptually adaptive embedding algorithm. The results are shown in Fig. 3.2. Then, the threshold was chosen such that watermarked and non-watermarked images could be well separated. According to the experimental result in Fig. 3.2, the threshold was assigned to 60% in all following experiments.

To evaluate the robustness of the proposed approach, six popular testing images: Lena, Baboon, F16, Fishing Boat, Pentagon and Peppers were watermarked with the proposed watermarking approaches. Then, four image processing operations, JPEG compression, Gaussian filtering, sharpening and line removing were applied on the watermarked images. The result are shown in Fig. 3.3 - Fig. 3.6. As shown in Fig. 3.3, six watermarked test images can be distinguished from their original versions easily. Although the perceptively adaptive embedding method can not successfully embed all watermark bits due to the adopted JND model, the proposed watermarking schemes are still very effective. The experimental results of watermark detection after JPEG compression are shown in Fig. 3.4. It is obvious that the watermark was still detectable until JPEG quality was lower than 15%. As shown in Fig. 3.5 and Fig. 3.6, similar experimental results can be obtained after line removing, Gaussian filtering or sharpening. These experimental results show that the proposed watermarking approaches are robust on minimizing the perceptual distortion.

3.5 Summary

In this chapter, we propose a strategy that hide watermarks in the relationship between wavelet coefficients. The proposed strategy would minimize the perceptual distortion of embedded images, measured by PSNR or JND. Experimental results illustrate the robustness of the proposed algorithm after common image processing operations such as JPEG compression, Gaussian filtering and sharpening.

Since the appropriate quality of images is different from application to application, it should be able to be pre-specified by the user. The proposed method can also be used to embed watermarks under the constraint of pre-specified image quality after slightly modification.

Chapter 4

Progressive Watermarking for Images on the Internet

4.1 Introduction

The rapid developments in computer and communication technologies have made more and more images and multimedia delivery over Internet. Progressive transmission methods are often used to allow a user to preview an image in advance. According to [14], there are four kinds of progressive image transmission methods, that are (1) Transmission Sequence-Based (TS-based) Method, (2) Successive approximation method, (3) Multistage residual method, and (4) Hierarchical method.

When there is more and more illegal spreading of digital media via Internet, stopping and/or preventing Internet piracy turns into an important affair. Among various media protection methods, digital watermarking [2] has long been an important and attractive digital protection technologies. However, most of digital watermarking methods prevent illegal use or spreading digital contents by checking whether the user owns lawful ownership. Thus, if a method that can detect the source of illegal distribution, so as to stop the transmission of unauthorized digital media will be a much better digital right protection approach. In this chapter, we propose a new wavelet based progressive watermarking system, which addresses the following scenarios:

1. **Domain Generality:** Although the proposed watermarking method is mainly de-

veloped on wavelet transform, the major concepts can be applied to other frequency based transform domain, such as DCT or block DCT.

2. **Progressive Detection:** The progressive detection characteristics of the proposed watermarking scheme can be nicely fit into some commonly used compression standards offering progressive encoding/decoding capability, such as JPEG and JPEG2000.

3. **Optimized imperceptibility:** The proposed watermarking method modifies several coefficients of an image at once to embed a binary value (e.g., a '0' or '1'). Thus, optimization methods can be used to select a proper value for each coefficients so that the watermarked can image achieve the best visual quality.

In order to verify the functionality and to evaluate the performance of the proposed watermarking system, two types of experiments were exercised: (1) robustness against image attacks and (2) early watermark detection along with progressive image transmission. Experiment results show that the proposed watermarking scheme has excellent robustness and imperceptibility as most of existing watermark systems, also successful early detection for various compression methods.

The rest of the chapter is organized as follows. Some related researches are discussed in Section 4.2. Section 4.3 describes the proposed watermarking embedding and detection methods. Section 4.4 presents experimental results of robustness against various attacks and earlier detection with respect to commonly used compression methods. Finally, Section 4.5 summarizes our methods and provides a brief concluding remarks.

4.2 Related works

In general, most of watermarking systems require robustness against attacks and imperceptibility of hiding watermarks. An network based progressive watermark system requires the following additional properties:

1. **Progressive detection:** when partial image contents are received, watermarking detection can start to check whether or not a suspected watermark exits.

2. **Early decision:** watermark detecter can make an early decision of whether a suspected watermark exists or not, without waiting to see the whole image contents.

These two properties provide a progressive watermarking system a great saving of processing resources as well as network bandwidth. Among various existed watermarking embedding methods, three different types are classified in [15]. These embedding methods are called as (1) additive, (2) multiplicative, and (3) quantizing watermarking. In general, these watermarking methods are applied when whole image contents are available, instead of partial image contents. Recently, new methods are proposed to detect watermarks in partial image contents. Aiming to *spectral selection* mode of JPEG compression, Chen et al., [16] proposed to embed additive watermarks to 8 × 8 DCT coefficients, and to have the coefficients in middle bands to be compressed and transmitted at an earlier time than that in high frequency bands. Therefore, the embedded watermarks can be detected in a progressive manner. However, this method does not provide early decision function, which is an important feature for saving computing resource and network bandwidth in progressive watermarking detection methods. In [17], Ashoka Jayawardena et al., proposed to transform watermarks into binary wavelet domain, so that the watermarks can be embedded in JEPG2000 compressed images, and then the watermarked image can be transmitted in multiresolution channels. Thus, when more wavelet coefficients of the watermark are transmitted and received, the more watermark data can be detected in a progressive manner. However, this method lacks of considering the decrease of image quality due to embedded watermarks.

4.3 Proposed watermarking for progressive transmission

By observing the transmission procedures of TS-based and successive approximation progressive transmission [18], we noticed that:

1. For the successive approximation transmission, the approximation value associated with larger wavelet coefficients often transmitted and received in the early part of a

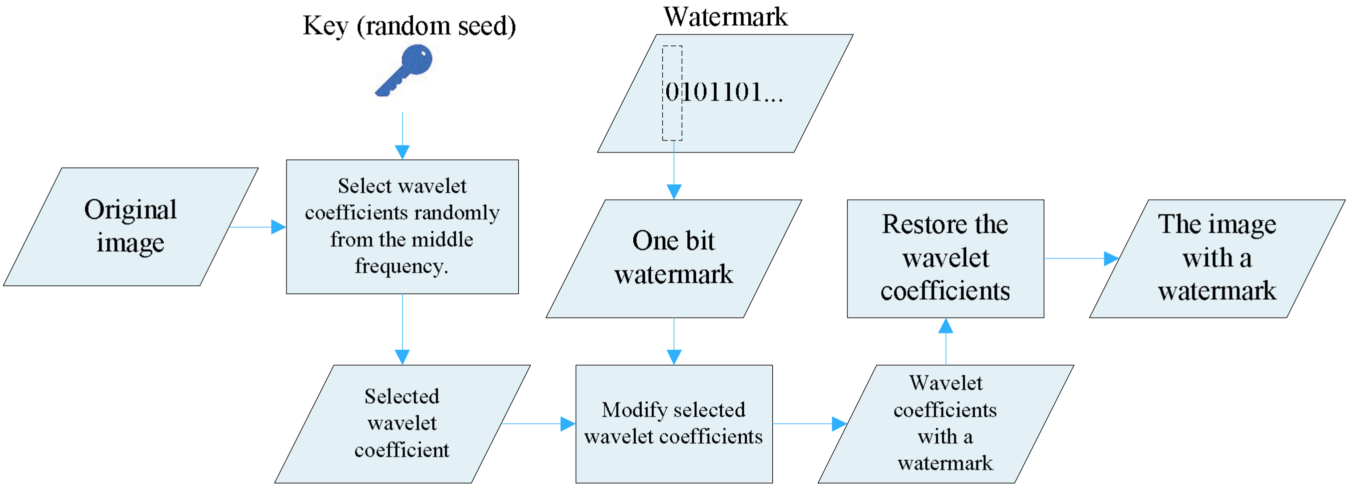

Key (random seed)
Watermark
0101101...
Original image
Select wavelet coefficients randomly from the middle frequency.
One bit watermark
Restore the wavelet coefficients
The image with a watermark
Selected wavelet coefficient
Modify selected wavelet coefficients
Wavelet coefficients with a watermark

Figure 4.1: A flow diagram of the proposed binary valued watermark embedding approach.

compressed data stream;

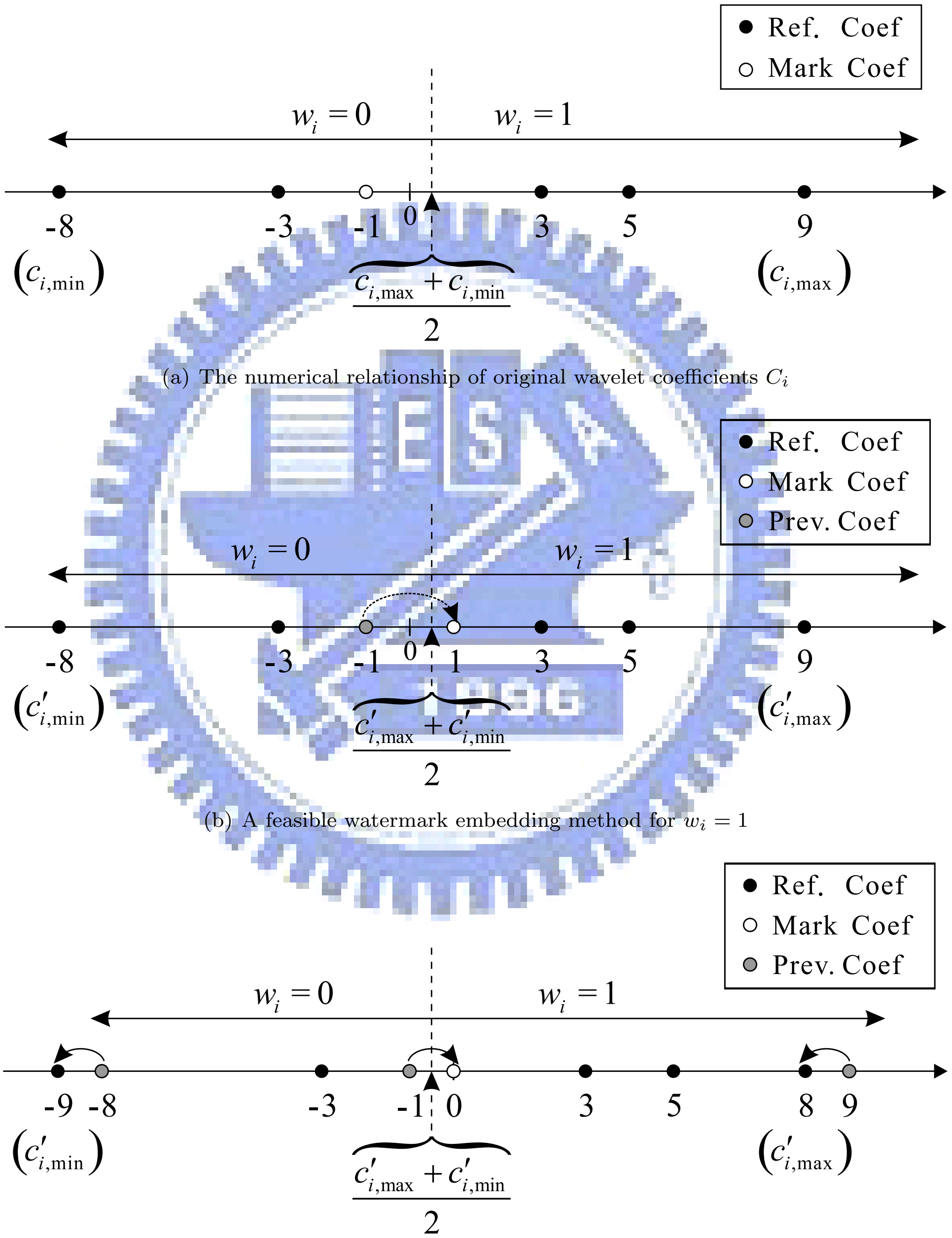
2. For the TS-based approximation transmission, when a watermark was embedded in earlier transmitted data stream, the earlier the watermark will be detected.

As described in Chapter 3 and in [19], we proposed a binary value watermark embedding method, which inserts each bit of a watermark according to the numerical relationship between wavelet coefficients of an image. Thus, embedding watermarks according to the numerical relationship of image data or coefficients seems to propose a new direction of watermark detection before the image data were all received. In the following, we will propose a new progressive watermarking embedding and detection approach.

4.3.1 Watermark embedding

In this section, a binary valued watermarking embedding approach for copyright protection is introduced. As shown in Fig. 4.1, $W = [w_1, w_2, ..., w_i, ..., w_L]^T$ is a binary valued watermark where $w_i \in \{0, 1\}$ for $i = 1, ..., L$ and L is the length of the watermark W. First, a random seed S is selected to generate $N + 1$ of random numbers for each bit w_i in W. These random numbers are used as indices to address $N + 1$ middle band wavelet coefficients $C_i = [c_{i,0}, c_{i,1}, ..., c_{i,N}]^T$ of an image. The first coefficient $c_{i,0}$ is called *mark coefficient*, and the rest coefficients $\{c_{i,1}, ..., c_{i,N}\}$ are called *reference coefficients*. The proposed method of embedding each bit w_i of W into an image is to modify the selected wavelet coefficients C_i into $C_i' = [c_{i,0}', c_{i,1}', ..., c_{i,N}']^T$ according to Eq. 4.1, where $c_{max}' = max(c_{i,1}', ..., c_{i,N}')$ and $c_{min}' = min(c_{i,1}', ..., c_{i,N}')$.

$$\begin{cases} \text{if} w_i = 1 & c_{i,0}' \geq \frac{c_{max}'+c_{min}'}{2}; \\ \text{if} w_i = 0 & c_{i,0}' < \frac{c_{max}'+c_{min}'}{2}. \end{cases} \tag{4.1}$$



(a) The numerical relationship of original wavelet coefficients C_i

(b) A feasible watermark embedding method for $w_i = 1$

(c) Another feasible watermark embedding method for $w_i = 1$

Figure 4.2: An example of the proposed watermark embedding approach.

In the following, C_i' is called *watermarked wavelet coefficients*. Figure 4.2 shows an example of numerical relationship of original wavelet coefficients of C_i, and Figs 4.2(b) and 4.2(c) show two different watermark embedding results for $w_i = 1$. As shown in Fig. 4.2, for $N = 5$, $C_i = [c_{i,0}, c_{i,1}, ..., c_{i,N}]^T = [-1, 5, -3, -8, 9, 3]^T$, where $c_{i,min} = -8$, $c_{i,max} = 9$. Suppose a watermark bit $w_i = 1$ is to be embedded into C_i, then C_i can be modified as either $C'_i = [1, 5, -3, -8, 9, 3]^T$, or $C'_i = [0, 5, -3, -9, 8, 3]^T$. Since there are many choices of C'_i to satisfy Eq. 4.1, thus we propose to use optimization methods [20] to select a C'_i, such that a watermarked image with best visual quality [2] can be achieved.

4.3.2 Watermark detection

In this section, we proposed a new progressive watermark detection method for various progressive image transmission schemes, such as: (1) successive approximation, (2) TS-Based method, (3) hybrid method of (1) and (2).

As shown in Fig. 4.3, the detector first checks to see whether or not the received image is transmitted in a progressive manner. If it is not, then the whole image will be received and then passed to non-progressive watermark detection procedure. If the image is transmitted progressively, then partial image will be passed to progressive watermark detection procedure.

Non-progressive watermark detection:

First, the same random seed S (as stated in Section 4.3.1) are used to generate $N+1$ indices to address $N+1$ middle band wavelet coefficients, $C''_i = [c''_{i,0}, c''_{i,1}, ..., c''_{i,N}]^T$ from the received image to calculate $c''_{max} = \max\left(c''_{i,1}, c''_{i,2}, \cdots, c''_{i,N}\right)$ and $c''_{min} = \min\left(c''_{i,1}, c''_{i,2}, \cdots, c''_{i,N}\right)$. Then, a possible watermark w'_i can be derived from Eq. 4.2:

$$w'_i = \begin{cases} 1 & \text{if} \quad c''_0 \geq \frac{c''_{max}+c''_{min}}{2} \\ 0 & \text{if} \quad c''_0 < \frac{c''_{max}+c''_{min}}{2} \end{cases}. \tag{4.2}$$

Finally, check if w'_i is equal to w_i for $1 \leq i \leq L$. If the number of matched watermark bits are larger or equal to a predetermined threshold T_w, then the received image can be confirmed to contain a watermark W.

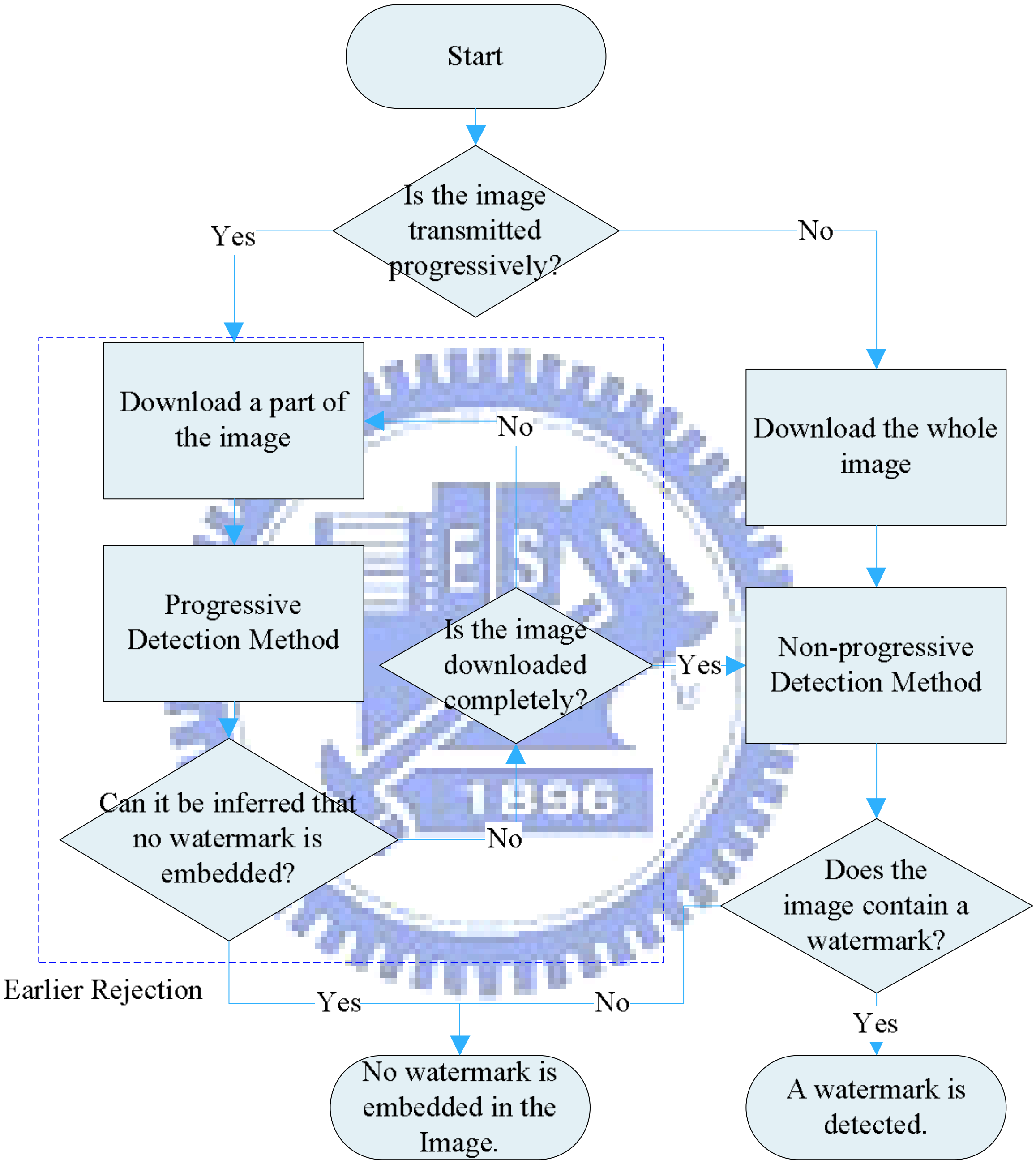

Start
Is the image transmitted progressively?
Yes
No
Download a part of the image
No
Download the whole image
Progressive Detection Method
Is the image downloaded completely?
Yes
Non-progressive Detection Method
Can it be inferred that no watermark is embedded?
No
Does the image contain a watermark?
Earlier Rejection
Yes
No
Yes
No watermark is embedded in the Image.
A watermark is detected.

Figure 4.3: A flowchart of the proposed progressive watermark detection.

Progressive watermark detection:

This method uses partially detected watermark coefficients to estimate embedded watermarks. As shown in Section 4.3.1, the proposed watermark embedding method use the numerical relationship between $c_{i,0}$ and $1/(2 \times (c_{i,min} + c_{i,max}))$ to decide how to modify the value of $c_{i,0}$. Detecting the watermarks can be performed in a similar manner. However, detecting progressively transmitted watermarks w_i needs to find the range relationship from partially received data of $c''_{i,0}$ and $1/(2 \times (c''_{i,min} + c''_{i,max}))$. Suppose the first m-th most significant bits of watermarked wavelet coefficients are received, and the same random seed S is used as the indices to select $N+1$ watermarked coefficients, $C''_i = [c''_{i,0}, c''_{i,1}, \cdots, c''_{i,j}, \cdots, c''_{i,N}]^T$. Let the range of each coefficient $c''_{i,j}$ be (l''_j, h''_j), if the following Eq. 4.3 holds, then the watermark w_i can be estimated:

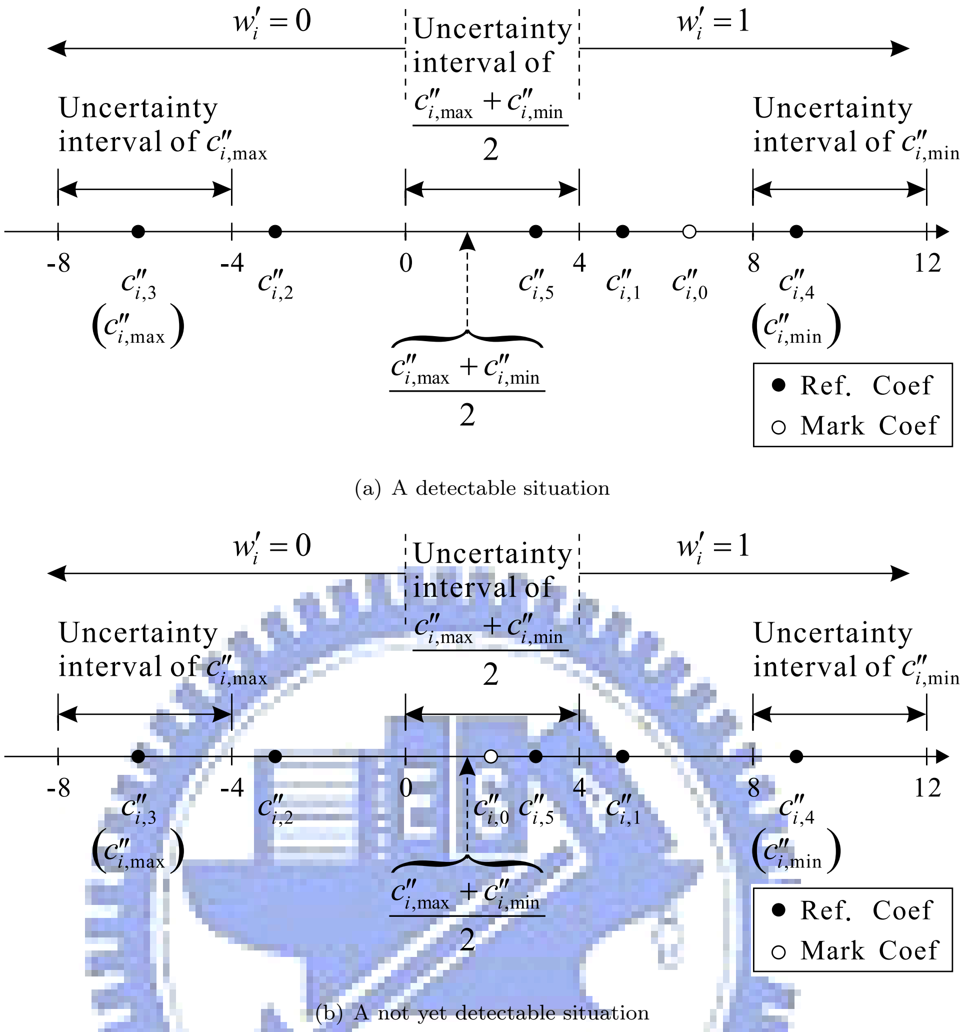
$$w'_i = \begin{cases} 1 & \text{if } \; l''_0 \geq \frac{h''_{max}+h''_{min}}{2} \\ 0 & \text{if } \; h''_0 < \frac{l''_{max}+l''_{min}}{2} \end{cases}, \tag{4.3}$$

where

$$\begin{aligned} l''_{max} &= \max\left(l''_{i,1}, l''_{i,2}, \cdots, l''_{i,N}\right), \\ h''_{max} &= \max\left(h''_{i,1}, h''_{i,2}, \cdots, h''_{i,N}\right), \\ l''_{min} &= \min\left(l''_{i,1}, l''_{i,2}, \cdots, l''_{i,N}\right), \quad \text{and} \\ h''_{min} &= \min\left(h''_{i,1}, h''_{i,2}, \cdots, h''_{i,N}\right). \end{aligned}$$

As shown in Fig.4.4(a), the ranges of $c''_{i,0}$ and $1/(2 \times (c''_{i,min} + c''_{i,max}))$ are not overlapped and the range of $c''_{i,0}$ is to the right of the range of $1/(2 \times (c''_{i,min} + c''_{i,max}))$, thus the value of w_i (=1) can be estimated. However, in Fig. 7(b), the ranges of $c''_{i,0}$ and $1/(2 \times (c''_{i,min} + c''_{i,max}))$ are overlapped to each other, and thus more lower bits of watermarked data are needed to separate the ranges of $c''_{i,0}$ and $1/(2 \times (c''_{i,min} + c''_{i,max}))$.

When the number of matched watermark bits is equal or larger than the predetermined threshold T_w, then the received image can be determined to contain the target watermark, ever through the whole image has not been completely received. On the other hand, if the number of mismatched watermark bits is larger than $L - T_w$, it can be inferred that the image does not contain the target watermark.


$w_i' = 0$
Uncertainty interval of $\frac{c''_{i,\max} + c''_{i,\min}}{2}$
$w_i' = 1$
Uncertainty interval of $c''_{i,\max}$
Uncertainty interval of $c''_{i,\min}$
-8
-4
0
4
8
12
$c''_{i,3}$ $(c''_{i,\max})$
$c''_{i,2}$
$c''_{i,5}$
$c''_{i,1}$
$c''_{i,0}$
$c''_{i,4}$ $(c''_{i,\min})$
$\frac{c''_{i,\max} + c''_{i,\min}}{2}$
● Ref. Coef
○ Mark Coef

(a) A detectable situation

(b) A not yet detectable situation

Figure 4.4: Examples of progressive watermark detection when partial watermarked image data are received

4.4 Experimental results

In order to evaluate the performance of the proposed watermarking system, two types of experiments were exercised: (1) robustness against image processing operations and (2) early watermark detection under the circumstance of progressive image transmission. In the first type of experiments, we use some commonly used image processing operations such as lossy compressions, Gaussian blurring and sharpening [12] to attack watermarked images, and then check whether or not the watermark can be detected. These experiments can briefly demonstrate the robustness of the proposed watermarking method. In the experiments for early watermark detection, watermarked images were firstly compressed by common compression standards such as JPEG and JPEG2000. Then, the experiments evaluate: (1) the percentage of watermark bits detected during progressive transmission, and (2) the amount of image data required to confirm the presence of a watermark. We also conduct the experiments on images without watermark and evaluate the amount of image data required to infer the absence of a watermark (i.e., earlier rejection).

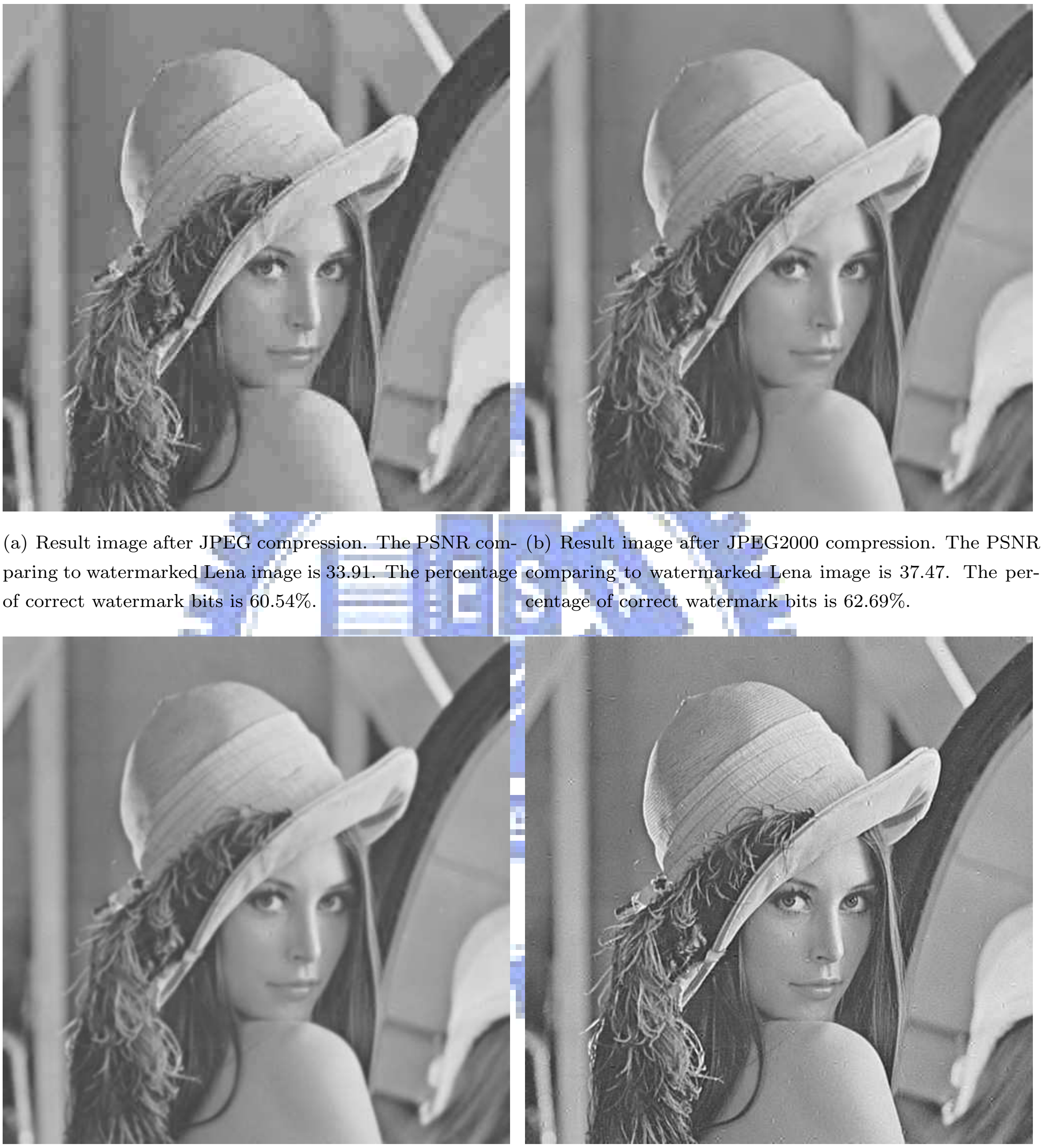
4.4.1 Experimental setting

In the experiments, we randomly generate a binary string containing 1000 bits as a watermark (i.e., L = 1024). All wavelet coefficients used to embed watermark bits are randomly selected from LH2, HL2 and HH2, and six coefficients are used for embedding each watermark bit (i.e., $N = 5$). While the watermark is to be embedded into 8×8 DCT coefficients of images, six coefficients are randomly chosen from AC6 to AC15 of all 8×8 DCT blocks. The watermark strength δ are always set as 1. Six images: Lena, Baboon, F16, Fishing Boat, Sailboat and Peppers are used as test image through all the experiments.

4.4.2 Robustness of the proposed progressive watermarking scheme

To evaluate the robustness of the proposed approach, the six test images were watermarked with the proposed watermarking approach. Then, four image processing operations, JPEG compression, JPEG2000 compression, Gaussian filtering, and sharpening are applied on the watermarked images. The result images are shown in Fig. 4.5 to Fig. 4.10. The experimental results of applying four image processing operations are depicted in Fig. 4.11, Fig. 4.12 and Table 4.1.

As shown in Fig. 4.11, Fig. 4.12 and Table 4.1, even visual quality of the test image is largely decreased by the image processing operations, the watermark embedded in the image can still be detected by the proposed method. Thus, if someone tries to remove the watermark embedded in an image, the image processing operations will cause huge decrease to visual quality of the image such that the image will be of no commercial value.



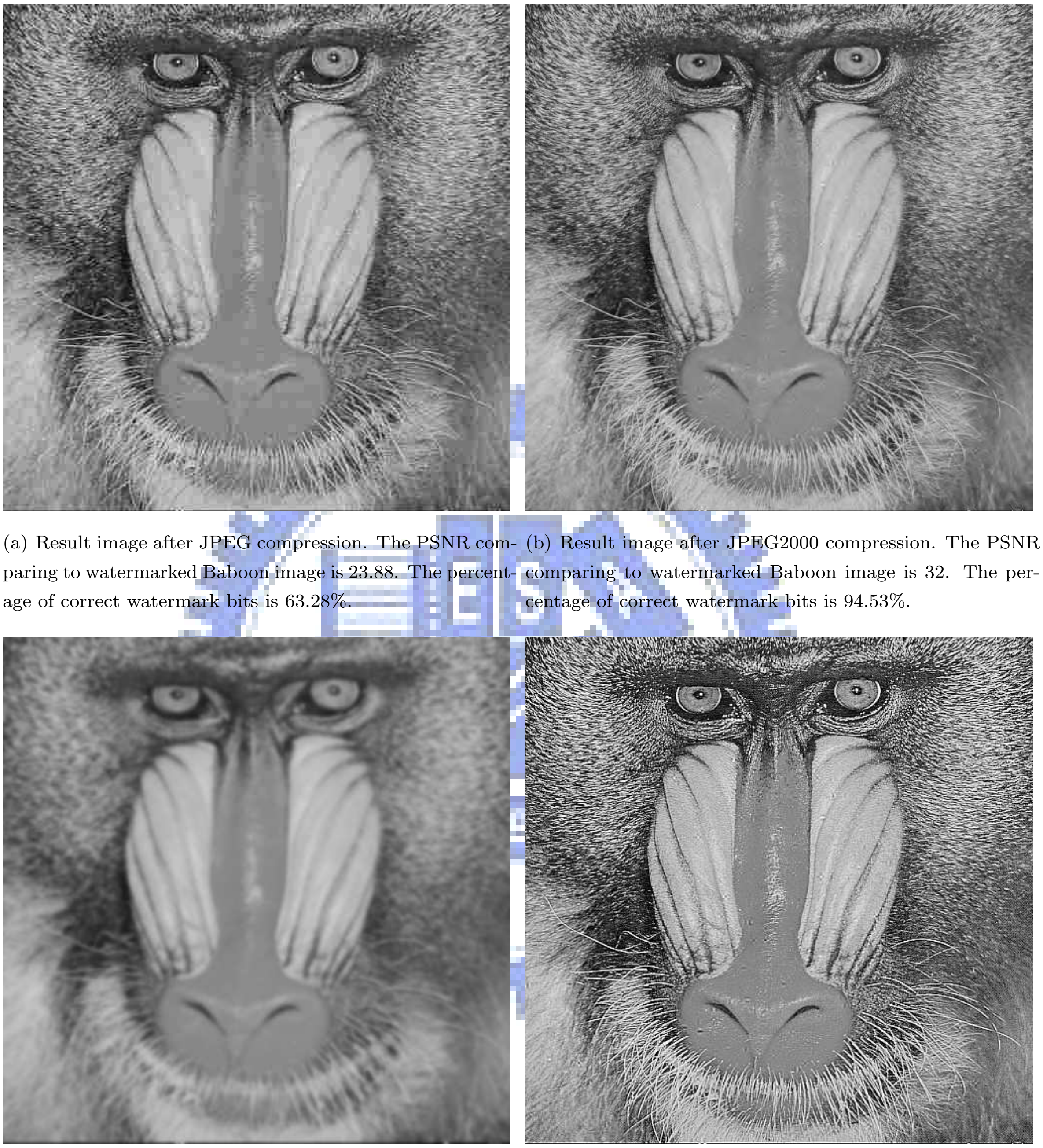
(a) Result image after JPEG compression. The PSNR comparing to watermarked Lena image is 33.91. The percentage of correct watermark bits is 60.54%.

(b) Result image after JPEG2000 compression. The PSNR comparing to watermarked Lena image is 37.47. The percentage of correct watermark bits is 62.69%.

(c) Result image after Gaussian filtering. The PSNR comparing to watermarked Lena image is 31.47. The percentage of correct watermark bits is 75.39%.

(d) Result image after sharpening. The PSNR comparing to watermarked Lena image is 31.74. The percentage of correct watermark bits is 99.9%.

Figure 4.5: The result images of applying four common image processing operations on watermarked Lena image.



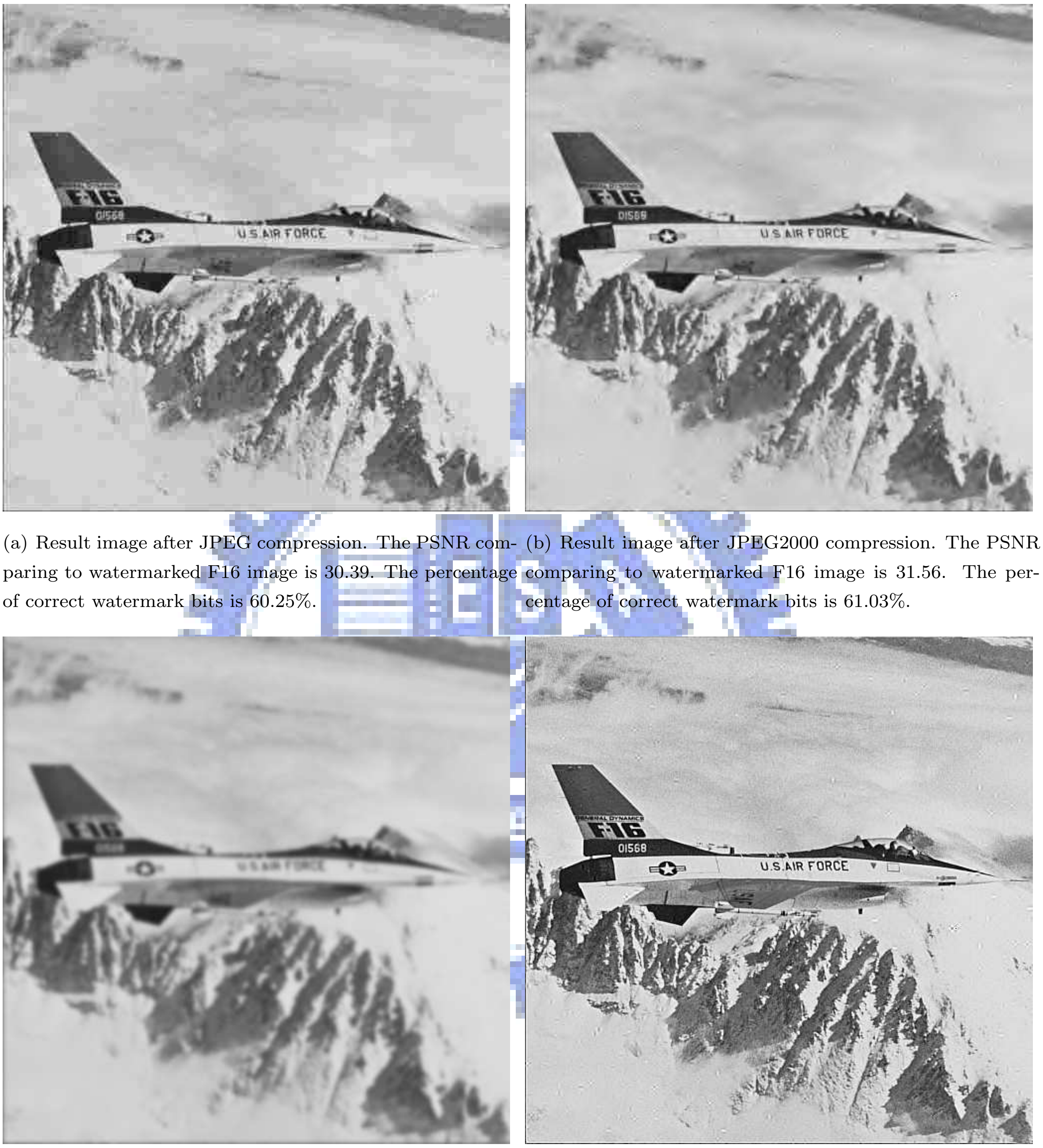
(a) Result image after JPEG compression. The PSNR comparing to watermarked Baboon image is 23.88. The percentage of correct watermark bits is 63.28%.

(b) Result image after JPEG2000 compression. The PSNR comparing to watermarked Baboon image is 32. The percentage of correct watermark bits is 94.53%.

(c) Result image after Gaussian filtering. The PSNR comparing to watermarked Baboon image is 21.37. The percentage of correct watermark bits is 60.64%.

(d) Result image after sharpening. The PSNR comparing to watermarked Baboon image is 17.53. The percentage of correct watermark bits is 94.62%.

Figure 4.6: The result images of applying four common image processing operations on watermarked Baboon image.



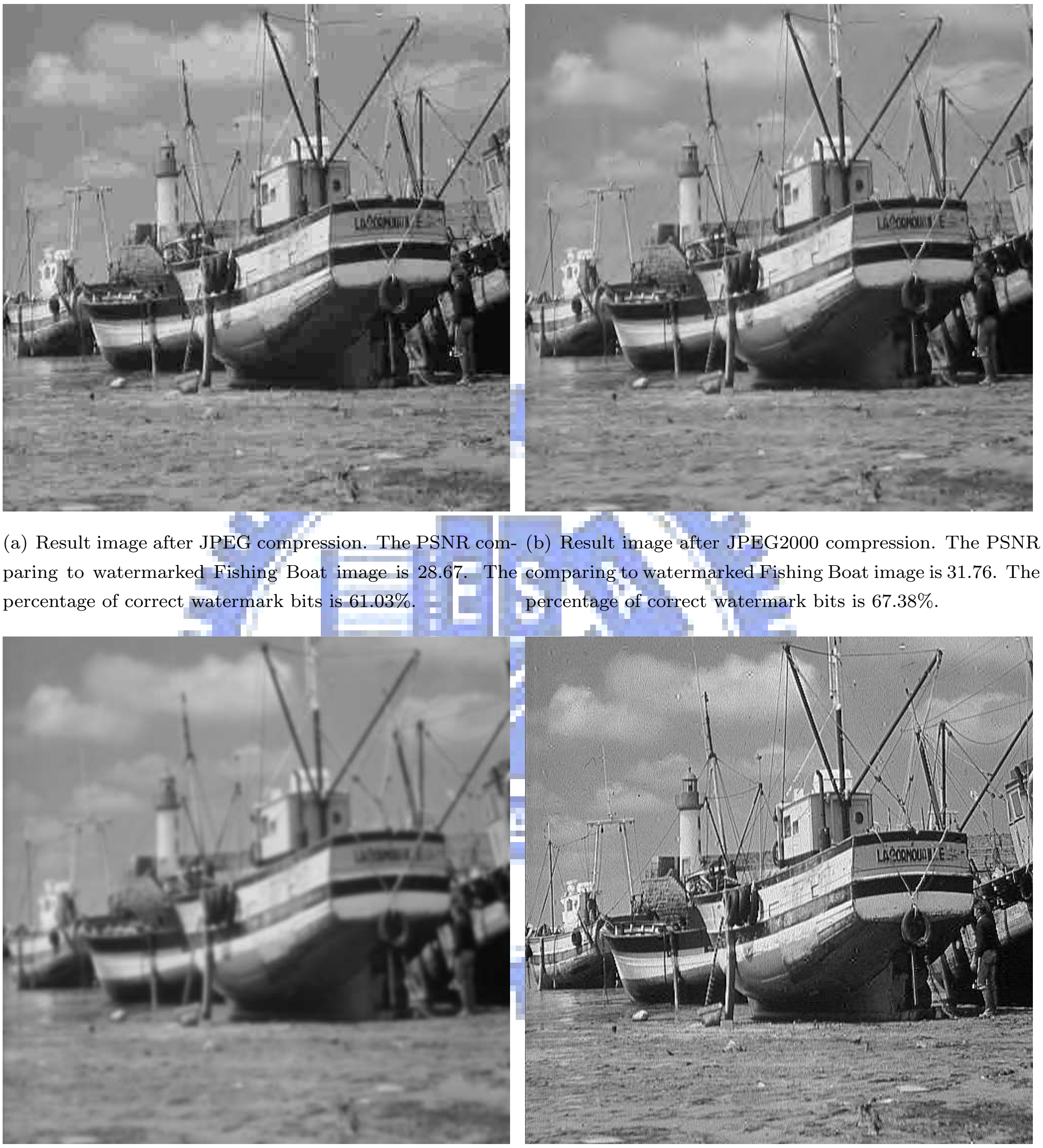
(a) Result image after JPEG compression. The PSNR comparing to watermarked F16 image is 30.39. The percentage of correct watermark bits is 60.25%.

(b) Result image after JPEG2000 compression. The PSNR comparing to watermarked F16 image is 31.56. The percentage of correct watermark bits is 61.03%.

(c) Result image after Gaussian filtering. The PSNR comparing to watermarked F16 image is 26.6. The percentage of correct watermark bits is 68.84%.

(d) Result image after sharpening. The PSNR comparing to watermarked F16 image is 25.42. The percentage of correct watermark bits is 99.21%.

Figure 4.7: The result images of applying four common image processing operations on watermarked F16 image.



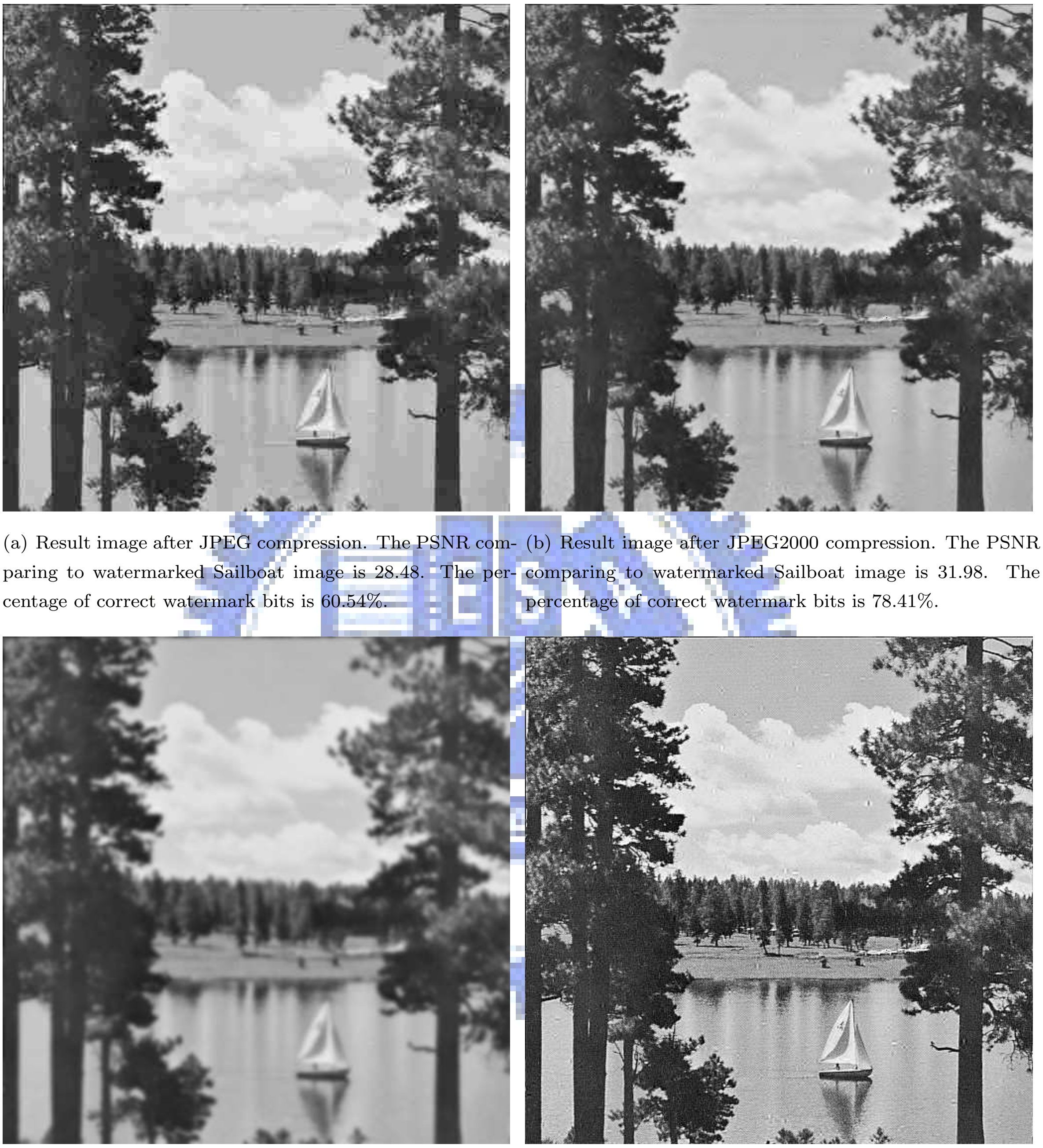
(a) Result image after JPEG compression. The PSNR comparing to watermarked Fishing Boat image is 28.67. The percentage of correct watermark bits is 61.03%.

(b) Result image after JPEG2000 compression. The PSNR comparing to watermarked Fishing Boat image is 31.76. The percentage of correct watermark bits is 67.38%.

(c) Result image after Gaussian filtering. The PSNR comparing to watermarked Fishing Boat image is 25.59. The percentage of correct watermark bits is 66.99%.

(d) Result image after sharpening. The PSNR comparing to watermarked Fishing Boat image is 23.56. The percentage of correct watermark bits is 98.82%.

Figure 4.8: The result images of applying four common image processing operations on watermarked Fishing boat image.



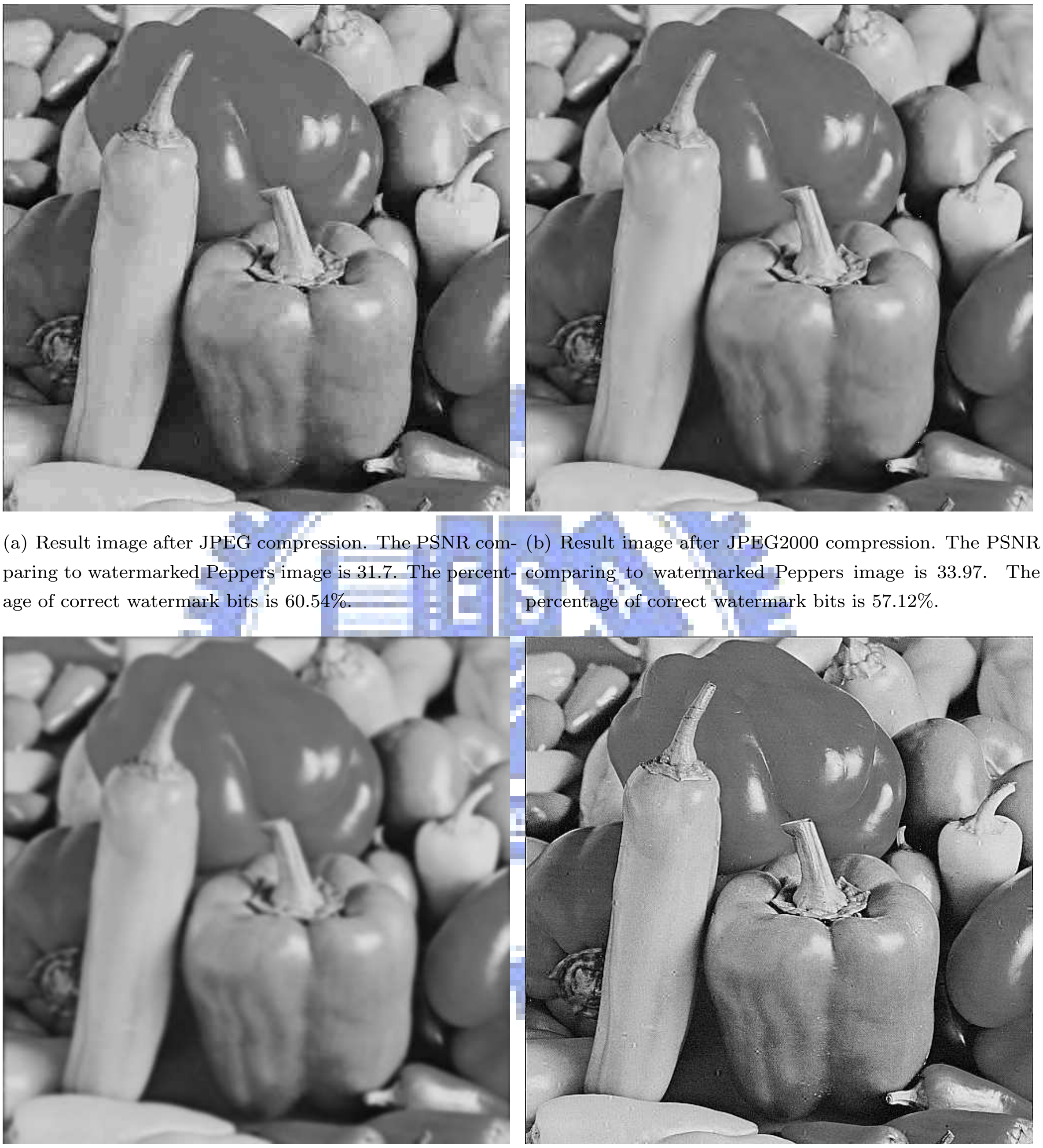
(a) Result image after JPEG compression. The PSNR comparing to watermarked Sailboat image is 28.48. The percentage of correct watermark bits is 60.54%.

(b) Result image after JPEG2000 compression. The PSNR comparing to watermarked Sailboat image is 31.98. The percentage of correct watermark bits is 78.41%.

(c) Result image after Gaussian filtering. The PSNR comparing to watermarked Sailboat image is 25.16. The percentage of correct watermark bits is 64.35%.

(d) Result image after sharpening. The PSNR comparing to watermarked Sailboat image is 23.15. The percentage of correct watermark bits is 99.31%.

Figure 4.9: The result images of applying four common image processing operations on watermarked Sailboat image.



(a) Result image after JPEG compression. The PSNR comparing to watermarked Peppers image is 31.7. The percentage of correct watermark bits is 60.54%.

(b) Result image after JPEG2000 compression. The PSNR comparing to watermarked Peppers image is 33.97. The percentage of correct watermark bits is 57.12%.

(c) Result image after Gaussian filtering. The PSNR comparing to watermarked Peppers image is 28.5. The percentage of correct watermark bits is 71.28%.

(d) Result image after sharpening. The PSNR comparing to watermarked Peppers image is 26.34. The percentage of correct watermark bits is 99.31%.

Figure 4.10: The result images of applying four common image processing operations on watermarked Peppers image.

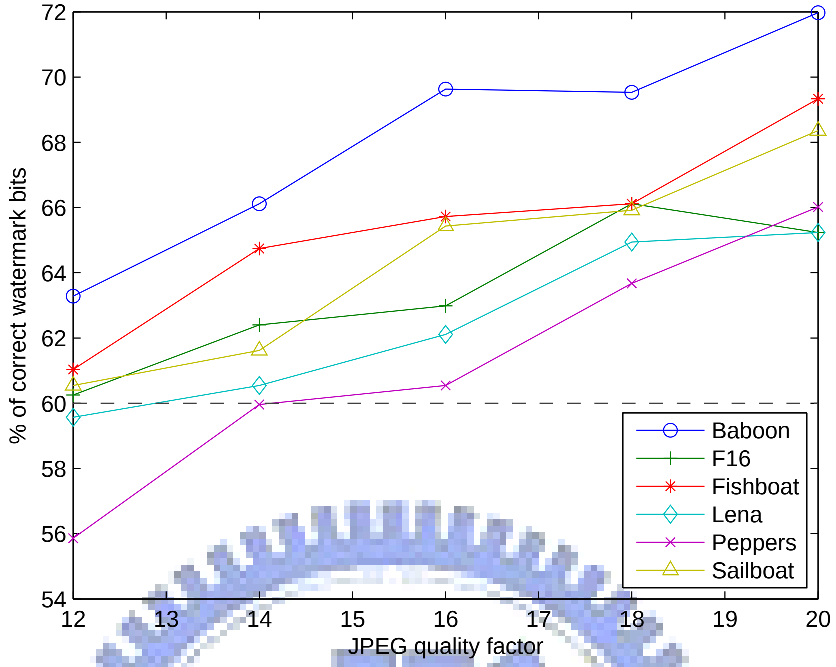


Figure 4.11: The plot of percentage of correct watermark bits versus JPEG quality factor. Six watermarked images were compressed by JPEG.

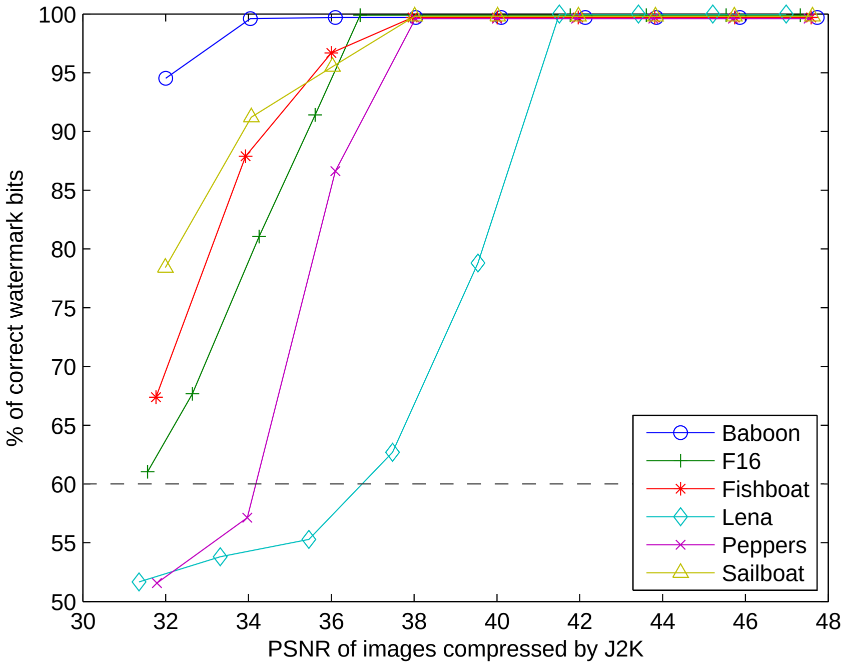


Figure 4.12: The plot of percentage of correct watermark bits versus PSNR. Six watermarked images were compressed by JPEG2000.

Table 4.1: Experimental results of watermark detection after Gaussian filtering and sharpening.

		Baboon	F16	FishBoat	Lena	Peppers	Sailboat
3×3 Gaussian filtering	PSNR	23.36	30.84	28.92	36.52	31.86	28.89
	% of correct bits	76.6	83.7	83.5	92.1	91.5	82.4
5×5 Gaussian filtering	PSNR	21.37	26.60	25.59	31.47	28.50	25.16
	% of correct bits	60.6	68.8	66.9	75.3	71.2	64.3
Sharpening	PSNR	17.53	25.42	23.56	31.74	26.34	23.15
	% of correct bits	94.6	99.2	98.8	99.9	99.3	99.3

These experimental results show that the proposed watermarking method is robust and can be used on the applications of copyright protection.

4.4.3 Earlier detection of the proposed progressive watermarking scheme

In this experiment, we embed watermarks into 8×8 DCT coefficients of the six test images, and then compress these images by common JPEG progressive modes (i.e., hybrid method). Then, the percentage of detectable watermark bits versus the percentage of received image data is evaluated. On the other hand, we embed watermarks into DWT coefficients of the six test images and compress these images by JPEG200. The percentage of detectable watermark bits versus the percentage of received image data is shown in Fig. 4.13.

As shown in Fig. 4.13, if T_w is equal to 0.6, it can be confirmed that the image is contained the watermark when about 15% to 35% image data is received. Moreover, the result shows that earlier detection of JPEG2000 images are significantly superior to JPEG image. This result may change when watermark strength δ or the subband chosen for watermark embedding is different. The larger the δ is, the earlier the embedded watermark can be detected. However, a large δ will decrease visual quality of watermarked images significantly. Embedding watermarks into low-frequency wavelet subband can also reduce the amount of image data required for progressive watermark detection; however, the watermark payload (i.e., coefficients that can be used to embed watermark) will also

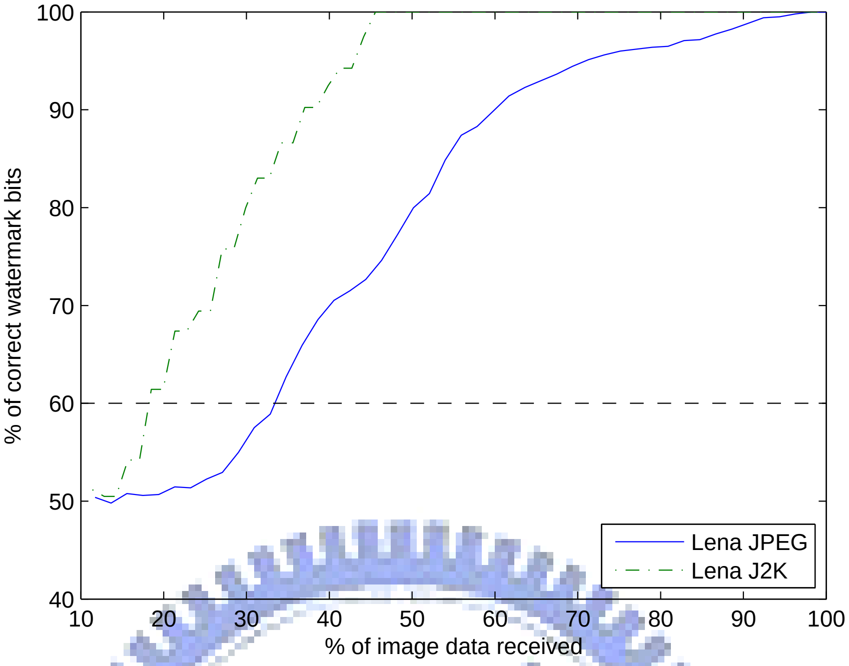


Figure 4.13: The plot of percentage of correct watermark bits versus received image data amount. JPEG hybrid mode and JPEG2000 were used to compress the watermarked image Lena.

deduce.

We also evaluate the performance of earlier rejection with images that do not contain any watermark. The six original test images are compressed by JPEG hybrid mode and JPEG2000, and then the watermark detection process is applied on these images to see how many percentage of image data is required for earlier rejection. In the experiment for JPEG, the watermark is embedded in 8 × 8 DCT domain, whereas the watermark is embedded in DWT domain in the experiment of JPEG2000. The experimental result are shown in Fig. 4.14. Intuitively, if $L - T_w$ extracted watermark bits are incorrect, it can be inferred that an image does not contain a watermark. While the watermark detection threshold T_w is equal to 0.6, the proposed method can infer the JPEG image does not contain the watermark with about 65% image data and the JPEG2000 image with about 30% image data. Again, the result shows that earlier detection of JPEG2000 images are significantly better than JPEG image because JPEG successive approximation mode transmits most significant bits in the order of zig-zag scan but not magnitude of

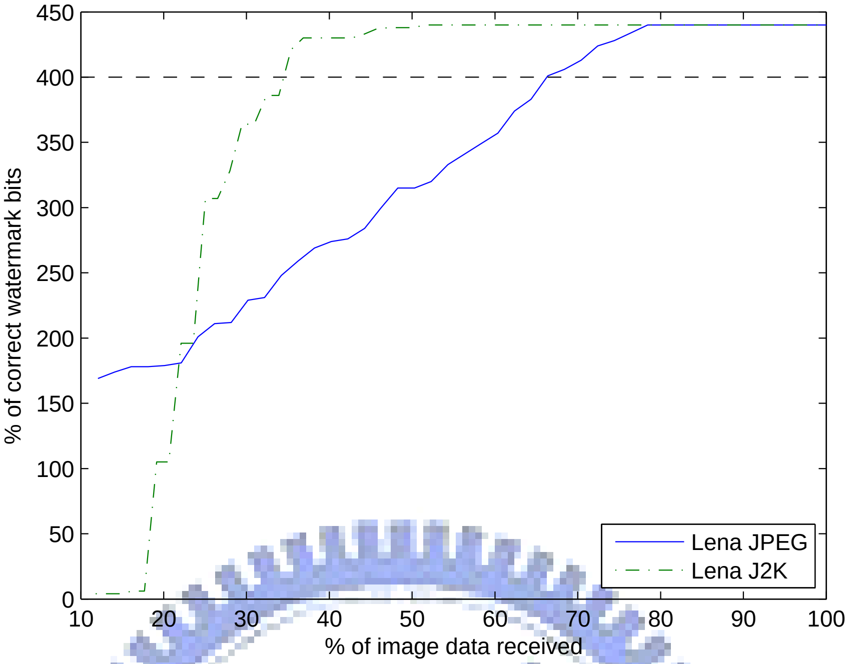


Figure 4.14: The plot of percentage of incorrect watermark bits versus received image data amount. JPEG hybrid mode and JPEG2000 were used to compress the watermarked image Lena.

coefficients.

4.5 Summary

In this chapter, we propose a binary value based progressive image watermarking for digital right protection. First, a new binary valued watermarking for 8×8 DCT or wavelet domain images are introduced, and then a progressive watermark detecting scheme for Internet environment is introduced. Its performance is also evaluated. The significance of the proposed method can protect digital right at distribution side, instead of at the user side. Also when a partial of image is determined to be free of piracy suspect, the progressive watermarking processing can be terminated to save computational resources as well as the network bandwidth.

Chapter 5

Fingerprinting for Multicast video

5.1 Introduction

Tremendous Internet e-learning and entertainment applications contribute to the importance of Internet TV and/or Video on Demand besides text- or image-based services. Nevertheless, the Internet TV and Video on Demand have encountered some difficulties: (1) the dramatically increasing bandwidth required for transmitting video data, and (2) rampant pirates. These are burning issues to the video-based Internet applications.

There is a growing interest in digital right protection (DRP) because of its potential to prevent video data from being pirated [1]. Digital watermarking and fingerprinting are two commonly used techniques for DRP. When an embedded watermark is associated with a particular user, it can be considered as a fingerprint. Once a fingerprinted media has been illegally distributed, the original user could be easily traced from redistributed version. Unfortunately, fingerprint embedding process often makes a signal media copy into many different versions, which have to be transmitted via unicast method. Generally, it is more efficient to transmit a single media via multicast method to massive clients [21, 22]. Thus, it becomes quite important in the field of video-based applications to transmit video data embedded with fingerprints efficiently to all the clients. To tackle the above problem, in this chapter, we propose a new fingerprint method that allows most of video content to be transmitted via multicast and only the very little fingerprint related contents are transmitted via unicast method.

5.1.1 Research goal

As discussed in Section 2.3, a fingerprint constructed with independent Gaussian watermarks possesses a variety of desired properties, such as shorter sequence length and better collusion resistance. In this chapter, we propose a new JDF method. The method first scrambles video data and multicasts video data to all clients, and then uses a secure channel to unicast a designated descramble key to each client. When a client uses the descrambling key to descramble received video, a designated fingerprint will be embedded into the video. Without any type of coding, transmitting descrambling keys to U clients requires $O(U)$ times of bandwidth than a single descrambling key. Thus, we proposed to use a *balanced incomplete block design* (BIBD) based method to reduce the number of descrambling keys up to the order of $O(\sqrt{U})$.

5.1.2 Chapter organization

The rest of the chapter is organized as follows. Some related works are discussed in Section 5.2. Section 5.3 introduces the concepts of the scrambling and multicasting scheme for video delivered to clients. Section 5.4 presents how to multicast a descrambling key to several clients and how to embed a fingerprint in the mean time when a descrambling process is performed. Section 5.5 discusses some practical issues on the proposed methods. Section 5.6 presents the experimental results of the proposed method. Finally, Section 5.7 summarizes the proposed method and draws a brief concluding remarks and future works.

5.2 Related works

Recently, methods of transmitting video data with fingerprints embedded have been seriously studied to tackle this problem. According to the timing when fingerprints are embedded into a video, most of proposed methods can be classified into one of the following cases: (1) transmitter-side fingerprint embedding, (2) receiver-side fingerprint embedding, (3) intermediate-node fingerprint embedding and (4) joint fingerprinting and decryption (JFD). Each of these cases will be briefly introduced and discussed in the following sections.

5.2.1 Transmitter-side fingerprint embedding

In general, these schemes embed users' fingerprints into the video to be multicasted at server side. Then the video is encrypted such that each user can only decrypt his own fingerprinted video. Wu and Wu [23] proposed a technique that multicasts most of the video data and unicasts a portion of the video data with unique fingerprints. When larger percentage of the video is chosen to be fingerprinted, encrypted, and unicasted, the security of transmitted video increases, but the efficiency of the protocol begins to resemble that of the simple unicast model. Boneh and Shaw [9] presented a method to distribute the fingerprinted copies of digital data with encryption schemes. In their approach, only two watermarked versions of video were needed, and these two versions of video data were transmitted in a multicasted manner. However, the bandwidth usage was almost doubled over that of normal multicast. The strategy was also adopted by some other methods on frame [24], packet [25] and segments of video stream [26].

5.2.2 Receiver-side fingerprint embedding

This type of embedding methods, was initially introduced in [27] and more recently discussion were appeared in [28] and [29]. In this scheme, a video is encrypted to produce an encrypted content and then multicasted to clients from the server. At the receiver side, the encrypted video is decrypted and then fingerprinted with an unique mark for each client, immediately. For security, tamperproof hardware must be used in order to protect the purely decrypted media content from eavesdropping. However, tamperproof hardware is difficult to build and is still an interesting open research problem.

5.2.3 Intermediate-node fingerprint embedding

This method proposed to distribute a fingerprinting process over a set of intermediate nodes such as routers [30]. Thus, by tracing the routing paths, the owner of a specific fingerprinted copy can be identified. However, this method creates a different set of challenges, such as vulnerability to intermediate node, compromise and susceptibility to standard network congestion, and packet dropping [31].

5.2.4 Joint fingerprinting and decryption (JFD)

JFD method was proposed by Kunder [32], which integrates decryption and fingerprinting process at the client side. The method allows a server multicasts only one encrypted video to all clients, and unicasts a designated decryption key to a specific client. As soon as the client decrypts the video data with the decryption key, a fingerprint is immediately embedded into the decrypted video. However, Kunder's JFD method did not use independent watermarks as fingerprints.

5.3 Joint independent fingerprinting and decryption

In this section, a new JFD approach using independent Gaussian watermarks as fingerprints is presented. As a result, a client can descramble videos multicasted from a content server with an unicasted descrambling key, and in the mean time an independent Gaussian fingerprint is embedded into the descrambled videos.

5.3.1 Notations and background assumption

The assumption of the proposed JFD scheme is presented firstly. When a server is about to multicast video data to U ($U > 1$) clients, the server scrambles the video data with a scrambling key, K^s, unknown to all clients. In addition, there is an unicast channel between the server and each client, to prevent information from being revealed to other clients. Thus, a client can receive a designated key from the unicast channel for descrambling and is forced to embed a fingerprint to the descrambled video data. Fingerprint detection is assumed to occur off-line at a later time when a pirate version of the video is suspected.

For the video descrambling and fingerprint embedding, the element-by-element multiplication and division between matrices are often used. Let K_1, K_2, $\ldots$, K_v be v

M-by-N matrices. The element-by-element multiplication of these v matrices is denoted as $K_1 \times K_2 \times \ldots \times K_v$. If $K = K_1 \times K_2 \times \ldots \times K_v$, then K is also an M-by-N matrix and $K[m,n] = K_1[m,n] \cdot K_2[m,n] \cdot \ldots \cdot K_v[m,n]$ where $1 \leq m \leq M$, $1 \leq n \leq N$. The element-by-element division can be defined in the same manner. Precisely, The element-by-element division of v matrices is denoted as $K_1/K_2/\ldots/K_v$. If $K = K_1/K_2/\ldots/K_v$, then K is also an M-by-N matrix and $K[m,n] = K_1[m,n]/K_2[m,n]/\ldots/K_v[m,n]$, where $1 \leq m \leq M$, $1 \leq n \leq N$. According to the definition of element-by-element multiplication, we can surely 'decompose' a specific matrix K into v component matrices such that $K = K_1 \times K_2 \times \ldots \times K_v$, or $K[m,n] = K_1[m,n] \cdot K_2[m,n] \cdot \ldots \cdot K_v[m,n]$. The *'decomposition'* operation will be used in descrambling key generation process.

The method to embed fingerprints into video frames is briefly introduced here. For detail description please refer to [33]. Let the frequency domain coefficient matrix of a frame in a video clip be F_c, Eq. (5.1) illustrates the method to embed a fingerprint, which is an independent Gaussian watermark, W^u into F_c:

$$F_c^u = F_c \times (\mathbf{1} + \alpha \cdot W^u), \tag{5.1}$$

where u is the user index, α is the fingerprint strength, which is usually set as 0.1 [33], and $\mathbf{1}$ is an M-by-N matrix with all its elements equal to 1. W^u is also an M-by-N matrix with most of its elements equal to 0. Only L elements of W^u are chosen from a normal distribution, $N(0,1)$.

In [34], a fingerprinting strategy is suggested to protect fingerprints from collusion attack. Therefore, a long video clip needs to be partitioned into segments, in which similar frames are contained. Then, for each client, frames in the same segment should be embedded with an identical fingerprint; on the contrary, frames in different segments are embedded with different fingerprints. In this chapter, it is assumed that the video is preprocessed, so an identical fingerprint is used for the same client.

5.3.2 Video scrambling and descrambling

This section presents the scrambling and descrambleing details of the proposed JFD method. To scramble a video clip, a scrambling key K^s was generated at server side, where K^s is an $M \times N$ matrix and each element in K^s is a real numbers randomly selected from a uniform distribution. For each frame in a video clip, the proposed scrambling processing can be represented by the following Eq. (5.2):

$$F_c^{K^s} = F_c / K^s, \tag{5.2}$$

or

$$F_c^{K^s}(m,n) = F_c(m,n)/K^s(m,n), \quad 1 \leq m \leq M, 1 \leq n \leq N, \tag{5.3}$$

where F_c is the original coefficients and $F_c^{K^s}$ is the scrambled coefficients of a video frame. Therefore, each element in F_c is divided by an corresponding element in K^s after scrambling process. Since all clients receive identical scrambled video clip, thus $F_c^{K^s}$ can be transmitted in a multicasting manner. Meanwhile, the server also prepares U descrambling keys, K^us for all U clients respectively, according to Eq.(5.4):

$$K^u = K^s \times (\mathbf{1} + \alpha \cdot W^u), \quad 1 \leq u \leq U. \tag{5.4}$$

A client u may use its descrambling key K^u received from unicast channel to descramble the multicasted video frame $F_c^{K^s}$, as follows:

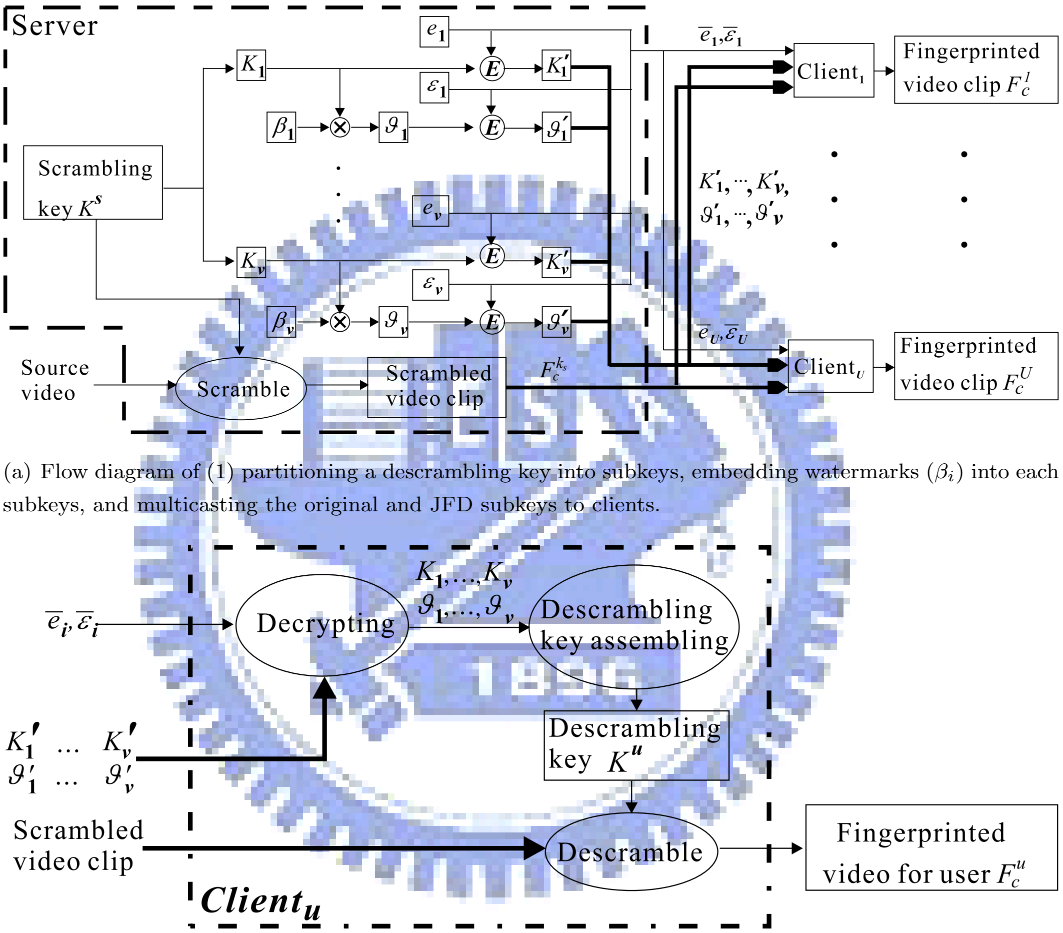
$$F_c^u = F_c^{K^s} \times K^u \tag{5.5}$$

By submitting Eq. (5.1) into Eq. (5.5), a fingerprinted frame can be achieved in the following manner:

$$\begin{aligned} F_c^u &= (F_c/K^s) \times (K^s \times (\mathbf{1} + \alpha \cdot W^u)) && (5.6) \\ &= F_c \times (\mathbf{1} + \alpha \cdot W^u). && (5.7) \end{aligned}$$

Eq. (5.6) shows that a video frame F_c can be embedded with a designated fingerprint in the mean time when the video frame was performed a descrambling process. According to Eqs. (5.5) and (5.6), transmission of scrambled video data by multicasting may save significantly needed network bandwidth, but the needed number of descrambling key is still directly proportional to the number of U users, i.e., $O(U)$. In the next section, we propose a method to further decrease the number of needed descrambling keys to $O(\sqrt{U})$.

5.4 Design and multicast a set of $O(\sqrt{U})$ descrambling keys for U clients



(a) Flow diagram of (1) partitioning a descrambling key into subkeys, embedding watermarks (β_i) into each subkeys, and multicasting the original and JFD subkeys to clients.

(b) Detailed processing of decryption, descrambling and embedding of fingerprints at client side.

Figure 5.1: Overview of partitioning and multicasting a descrambling key from a content server to clients, and embedding a fingerprint to the descrambled video.

This section presents a method of generating and multicasting v sets of descrambling keys, referred to as subkeys, to U clients, where v is in the order of $O(\sqrt{U})$. Figure 5.1 (a) shows an overall diagram of the proposed method. As described in Section 5.4.1, a randomly generated scrambling key is first decomposed into v subkeys. These subkeys are then embedded with Gaussian watermarks to generate v JFD subkeys. These v JFD subkeys along with v original subkeys are then organized into v sets and multicasted to U clients. By using the method described in Section 5.4.2, descrambling keys for U clients can be combined from v sets of subkeys.

5.4.1 Balanced incomplete block designs (BIBD)

The basic concept of multicasting v sets of descrambling subkeys to U clients is mainly motivated from the *balanced incomplete block design* (BIBD) methods [35, 36]. A (v, k, λ) balanced incomplete block design (BIBD) is a pair $(\mathcal{X}, \mathcal{A})$, where $\mathcal{A}$ is a collection of k-element blocks (subsets) of a v-element set $\mathcal{X}$, such that each pair of elements of $\mathcal{X}$ occur together in exactly λ blocks.

According to the design rule of BIBD, for a (v, k, 1)-BIBD, any two subsets in $\mathcal{A}$ can only have at most one common element from $\mathcal{X}$, and the number of k-element subsets in $\mathcal{A}$ can be enumerated by $(v^2 - v)/(k^2 - k)$. In this chapter, we will use (v, k, 1)-BIBD to design decomposing strategies for the descrambling subkeys. In the following, a (9, 3, 1)-BIBD is used as an example to illustrate some major design concepts. A (9, 3, 1)-BIBD is a pair of $(\mathcal{X}, \mathcal{A})$, where $\mathcal{A}$ is a collection of 3-element subsets of a 9-element set $\mathcal{X}$. Thus, for a given set $\mathcal{X} = \{a_1, a_2, a_3, a_4, a_5, a_6, a_7, a_8, a_9\}$, the collection of subset $\mathcal{A} = \{\{a_1,\ a_2,\ a_3\},\ \{a_4,\ a_5,\ a_6\},\ \{a_7,\ a_8,\ a_9\},\ \{a_1,\ a_4,\ a_7\},\ \{a_2,\ a_5,\ a_8\},\ \{a_3,\ a_6,\ a_9\},\ \{a_1,\ a_5,\ a_9\},\ \{a_2,\ a_6,\ a_7\},\ \{a_3,\ a_4,\ a_8\},\ \{a_1,\ a_6,\ a_8\},\ \{a_2,\ a_4,\ a_9\},\ \{a_3,\ a_5,\ a_7\}\}$.

In most application of BIBD, the collection of subsets in $\mathcal{A}$ can be represented in a matrix form, called *incident matrix*. The following is the incident matrix $\mathbf{B}$ for the collection $\mathcal{A}$ of an example (9, 3, 1)-BIBD,

$$B = \begin{bmatrix} 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}. \tag{5.8}$$

In matrix B, each column corresponds to a subset in $\mathcal{A}$. Each ”1” in a column of B corresponds to an element of a subset in $\mathcal{A}$. For instance, the first column in B has 3 ”1s”, at the first 3 rows, which represents $\{a_1, a_2, a_3\}$, and the three 1’s in the 2nd column corresponds to $\{a_4, a_5, a_6\}$.

5.4.2 Multicasting descrambling subkeys with BIBD

This section presents the method of using a $(v, k, 1)$-BIBD to decompose a descrambling key for U clients. For U clients, the values of U, v and k needs to satisfy the following relation:

$$U \leq (v^2 - v)/(k^2 - k). \tag{5.9}$$

Given a (9, 3, 1)-BIBD, the collection $\mathcal{A}$ contains 12 different 3-element subsets, and any two subsets in $\mathcal{A}$ can have at most one common element. Based on this simple concept in BIBD strategy, nine different (finger) patterns can be combined to generate enough different fingerprints for 12 users. In the following, a $(v, k, 1)$-BIBD is used to guide the design of the proposed JFD method. First, we assume that a fingerprint can be composed from a set of watermarks. According to the number of users, a proper $(v, k, 1)$-BIBD is selected first, and v watermarks are generated and correspond to one and only one element in $\mathcal{X}$. Then, each subset in $\mathcal{A}$ corresponds to a designated fingerprint for each client. Hence, for U clients, only v $(v < U)$ watermarks are need to be sent in a multicast manner. Similarly at a client side, a fingerprint can be combined from k $(k < v)$

watermarks. It is easy to see that between any two fingerprints (i.e., a subset in $\mathcal{A}$), there exist at most one common watermarks (i.e., one elements in $\mathcal{X}$). Thus, multicasting v watermarks for the generation of U fingerprints is an efficient approach.

As shown in Figure 5.1 (a), a scrambling key K^s is decomposed into v subkeys K_1, K_2, $\ldots$, K_v. The core idea in the proposed method is embedding an independent Gaussian watermark β_i into each scrambling subkey K_i, to form a JFD subkey ϑ_i:

$$\vartheta_i = K_i \times (1 + \alpha\beta_i), \qquad 1 \leq i \leq v. \tag{5.10}$$

Both K_i and ϑ_i are then encrypted with two encrypting keys e_i and ε_i into K'_i and ϑ'_i respectively. Finally, a total v sets of descrambling subkey pairs $\{(K'_i, \vartheta'_i)|1 \leq i \leq v\}$ are multicasted from the server to all clients. A complete description of subkey pair decrypting, video descrambling, and fingerprint embedding at clients are presented in Figure 5.1(b). First, a combination of v decrypting keys selected from $\{e_i|1 \leq i \leq v\}$ and $\{\varepsilon_i|1 \leq i \leq v\}$ is sent to a designated client u via a secure channel. This combination of v decryption keys consists of two sets of keys $\bar{\varepsilon_u}$ and $\bar{e_u}$. $\bar{\varepsilon_u}$ contains k decryption keys selected from $\{\varepsilon_i|1 \leq i \leq v\}$ and $\bar{e_u}$ contains $v - k$ keys from $\{e_i|1 \leq i \leq v\}$. According to the $(v, k, 1)$-BIBD adopted, if the element (i, u) of the incident matrix is equal to 1, ε_i is sent to client u; otherwise, e_i is sent. Precisely speaking, $\bar{\varepsilon_u} = \{\varepsilon_i|B(i, u) = 1\}$, and $\bar{e_u} = \{e_i|B(i, u) = 0\}$, where B is the incident matrix of the $(v, k, 1)$-BIBD. Next, these v decryption keys are used to decrypt v descrambling subkeys from subkey pairs $\{(K'_i, \vartheta'_i)|1 \leq i \leq v\}$. Since the method based on $(v, k, 1)$-BIBD is used to distribute decryption keys to each client, client u can only decrypt one descrambling subkey from each encrypted subkey pair (K'_i, ϑ'_i), $1 \leq i \leq v$. Finally, these decrypted subkeys are combined to form a descrambling key K^u and can be used to descramble the multicasted video clip. Because some of the decrypted descrambling subkeys were embedded with independent Gaussian watermarks (i.e., JFD subkeys), thus the descrambled video clip will contain fingerprints composed of these watermarks.

In the following, an example of (9, 3, 1)-BIBD is used to illustrate the details of descrambling subkey decrypting and fingerprint embedding process. Suppose a combination of decryption keys composed of $\{\bar{e_2}, \bar{\varepsilon_2}\}$ is unicasted to client 2, according to column 2 of the incident matrix B in Section 5.4.1, the elements in $\bar{e_2}$ and $\bar{\varepsilon_2}$ are $\bar{e_2} = \{e_1, e_2, e_3, e_7, e_8, e_9\}$ and $\bar{\varepsilon_2} = \{\varepsilon_4, \varepsilon_5, \varepsilon_6\}$, respectively. After decrypting process, a descrambling key $K^{(2)}$ designated to client 2 can be obtained:

$$\begin{aligned} K^{(2)} &= K_1 \times K_2 \times K_3 \times \vartheta_4 \times \vartheta_5 \times \vartheta_6 \times K_7 \times K_8 \times K_9 && (5.11) \\ &= K^s \times (1 + \alpha\beta_4)(1 + \alpha\beta_5)(1 + \alpha\beta_6). && (5.12) \end{aligned}$$

By submitting Eq. (5.12) into Eq. (5.1), a fingerprinted frame F_c^2 designated to client 2 can be expressed as:

$$F_c^2 = F_c^{K_s} \times K_2 = F_c \times (1 + \alpha\beta_4)(1 + \alpha\beta_5)(1 + \alpha\beta_6) \qquad (5.13)$$

Let $W^2 = (1/\alpha)(1 + \alpha\beta_4)(1 + \alpha\beta_5)(1 + \alpha\beta_6) - 1$, then

$$F_c^2 = F_c \times (1 + \alpha W^2) \qquad (5.14)$$

As shown in Eq. (5.12), $K^{(2)}$ contains three independent Gaussian watermarks, i.e., β_4, β_5 and β_6, which can be combined to generate a fingerprint (see Eq. (5.14)) for client 2. As described in Section 5.3.1, the descrambling subkeys are obtained by decomposing the scrambling key randomly, so only the random seeds are required to be transmitted for the purpose of saving network bandwidth. Thus, the scrambled video clip and JFD subkeys are the only data needs to be multicasted from server to clients. By comparing with the required bandwidth of JFD subkeys, the bandwidth required to transmit encryption keys $\{\bar{e}_i, \bar{\varepsilon}_i | 1 \leq i \leq v\}$ and the random seeds of subkeys is very small and can be ignored. Thus, the needed transmission bandwidth comes from the v JFD subkeys. According to the relation $U \leq (v^2 - v)/(k^2 - k)$, of a ($v$, k, 1)-BIBD, the needed bandwidth can be estimated in the order of $O(v) = O(k\sqrt{U})$.

5.5 Practical issues

In this section, we discuss some practical issues for the application of proposed methods in real world situation.

Issue 1: In general, the security of the proposed fingerprinting method is highly related to the number of watermarks embedded into media contents. However, embedding too many watermarks in an image or a frame of video may often decrease the visual quality, i.e., lower down the PSNR value.

Issue 2: Recently, most media contents are stored and transmitted in compressed formats. Some lossy compression techniques adopt methods of deleting or quantizing high frequent components to reduce the total data size. Thus, properly deciding the number and location of coefficients for fingerprint embedding are an important consideration for digital right protection.

Issue 3: The two JFD methods presented in Sections 5.3 and 5.4 have their strength and weakness in different circumstances. The major strength of the JFD method proposed in Sections 5.4 is at its multicast features, which can save transmission bandwidth required at server side. However, this method is highly dependent on the BIBD scheme. When U is not equal to $(v^2 - v)/(k^2 - k)$, the multicast capability of the method is not fully utilized. Under these circumstances, we can find v, k and U_ϵ for the given U such that $U = (v^2 - v)/(k^2 - k) + U_\epsilon$ is satisfied, where U_ϵ is the smallest positive number for all possible pairs of v and k in a (v, k, 1)-BIBD. Then, the method proposed in Section 5.4 can be used to multicast the (v, k, 1)-BIBD based descrambling subkeys to $U - U_\epsilon$ clients, and the method described in Section 5.3 to unicast U_ϵ descrambling keys to the rest of clients. It is also possible to separate clients into several portions and multicast descrambling subkeys to each portion with different BIBD.

Issue 4: When many watermarks are embedded into an image or video frame, they may be overlapped with each other. That is, some coefficients may be used to embed more than one watermarks. The overlap will slightly decrease the detection strength of these watermarks. Practically, we can separate the coefficients into v sets for v watermarks to avoid the overlap.

issue 5: If we randomly decompose a scrambling key into v subkeys, we can only represent and transmit $v - 1$ subkeys by random seeds. To tackle this problem, we

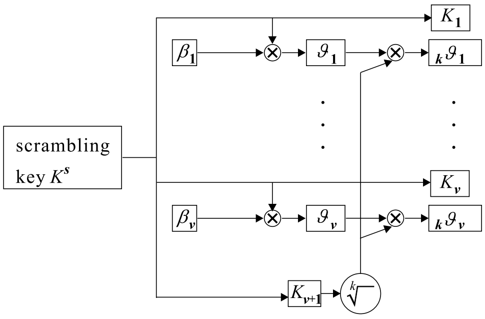


Figure 5.2: Methods to prevent malicious users from using key collusion attacks.

can decompose a scrambling key into $v+1$ subkeys. The first v subkeys are used to generate JFD subkeys as described in Section 5.4.2. As shown in Figure 5.2(b), a k-th root calculation is applied on the $(v+1)$-th subkey, and then the k-th root value of $(v+1)$-th subkey is multiplied with JFD subkeys ϑ_i to produce new JFD subkeys $_k\vartheta_i$. Similar to the derivation of Eq. (5.11), the calculation of descrambling key for client 2 becomes:

$$\begin{aligned} K^{(2)} &= K_1 \times K_2 \times K_3 \times_k \vartheta_4 \times_k \vartheta_5 \times_k \vartheta_6 \times K_7 \times K_8 \times K_9 && (5.15)\\ &= K_1 \times K_2 \times K_3 \times (\sqrt[3]{K_{10}} \times \vartheta_4) \times (\sqrt[3]{K_{10}} \times \vartheta_5) \times (\sqrt[3]{K_{10}} \times \vartheta_6) \times \\ &\quad K_7 \times K_8 \times K_9 \\ &= K_1 \times K_2 \times K_3 \times \vartheta_4 \times \vartheta_5 \times \vartheta_6 \times K_7 \times K_8 \times K_9 \times K_{10} && (5.16)\end{aligned}$$

Thus, a scrambled video clip can be descrambled and fingerprinted with the method proposed in Section 5.4.

issue 6: There are systematic methods for constructing infinite families of BIBDs. For example,$(v, 3, 1)$ systems (also known as Steiner triple systems) are known to exist if and only if $v \equiv 1$ or 3 (mod 6) [35]. Techniques for constructing several kinds of BIBDs can be found in [37].

5.6 Experimental results

In this section, two types of experiments were exercised to demonstrate the performance of the proposed fingerprinting methods. In these experiments, a few watermarks are embedded in a video frame as a user's fingerprint. In the first type of experiment, we evaluate how the number of embedded watermarks may effect the visual quality of fingerprinted images. Usually, more embedded watermarks would provide better protection for a fingerprinted image. By embedding several watermarks at randomly selected coefficients of an image, it is possible that a few coefficients may be repeatedly selected to embed different watermarks. Thus, the second types of experiments will evaluate the robustness of a fingerprint containing multiple watermarks. The watermark embedding method used in the experiment in this section is modified from the method proposed by Cox [33] to fit the 8×8 DCT transformed video frames, which are basic frames for MPEG 1 and 2. A JFD subkey ϑ_i is first calculated from an independent Gaussian watermark β_i according to Eq. (5.4) or Eq. (5.10), then the watermark is embedding to a randomly selected location in the middle frequency coefficient area of 8×8 DCT blocks after descrambling process. To detect a watermark, the 8×8 DCT coefficients in which the original watermark was embedded were extracted, and then Eq. (5.17) is used to recover the watermark $\hat{W}$, Then, $\hat{W}$ and β_i are compared with each other of their similarity according to Eq. (5.18). When the computed similarity value is larger than a predetermined threshold value T_c, then the recovered watermark $\hat{W}$ is considered to be a valid watermark β_i.

$$\hat{W} = \frac{1}{\alpha}(\frac{F_c^u}{F_c} - 1) \tag{5.17}$$

$$Sim(\hat{W}, \beta_i) = \frac{\hat{W}}{\sqrt{\hat{W} \cdot \hat{W}}} \cdot \frac{\beta_i}{\sqrt{\beta_i \cdot \beta_i}} \tag{5.18}$$

In the following experiments, images were extracted from the I-frame of Table Tennis video sequence, obtained from http://media.xiph.org/video/derf/. The strength value α of a watermark is limited between 0.1 and 0.5, the length of a watermark is set to be 1000 real values (i.e., $L = 1000$, and the predetermined threshold T_c for fingerprint detection is 0.7.

In the first types of experiments, images for embedding fingerprints are extracted from the I-frame of Table tennis video sequence. Each extracted 352 × 288 frames are first transformed into 8 × 8 DCT domain. Then, various numbers (from 1 to 20) of watermarks

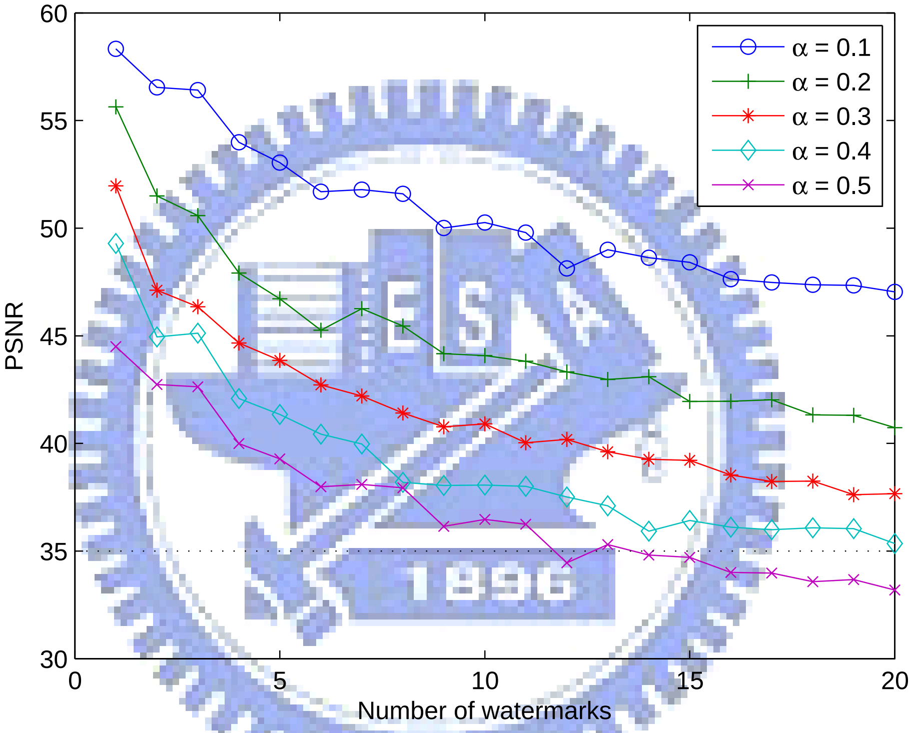


Figure 5.3: The experimental results show that the relationship of the image visual quality (PSNR) vs. the numbers of embedded watermarks.

are embedded into the DCT coefficients of the transformed video frames according to Eq. (5.11). The PSNR values of the watermark embedded images are depicted in Figure 5.3. As stated in [38], when the PSNR value is larger than 30, the visual quality of an image is acceptable. For different embedded watermark strength α (i.e., $0.1 \leq \alpha \leq 0.5$) their PSNRs are higher than 30, which is a generally accepted level of visual quality. According to the proposed fingerprint embedding method, when 19 watermarks are embedded into an image simultaneously at client side, the $(v, k, 1)$-BIBD based method may allow 70 clients to receive a multicasted video clip with a $(21, 3, 1)$-BIBD. This scale of client size seems to be satisfactory for must real world needs.

In the second type experiments, each extracted frame from Table tennis video sequence is embedded with various number (from 1 to 30) of watermarks with strength α from 0.1 to 0.5. Overlapping among embedding watermarks is allowed in order to test the robustness of the proposed fingerprint method. Table 5.1 shows the similarity value between the extracted fingerprints and the original fingerprints for various number of embedded watermarks and various strength α.

As shown in Table 5.1, although watermarks may be overlapped with each other, the number of corrupted fingerprints is very small. Furthermore, when the number of embedded watermarks increases to 20, the average similarity of watermarks is slightly decreased to 0.65. Thus, we would like to claim that the proposed fingerprint embedding method is practical as well as robustness for real world applications.

5.7 Summary

In this chapter, we propose a new video scrambling and fingerprinting approach for digital media right protection. The proposed method contains two parts: (1) video scrambling at server side, and (2) fingerprint embedding at client side. First, a content server scrambles and multicasts video contents to clients. Then by applying a (v, k, 1)-BIBD scheme, the server partitions a the scrambling key into $O(\sqrt{U})$ descrambling subkeys, and multicasts to U clients. Receiving a designated secret key from the content server, each client

Table 5.1: Average similarity values of four testing images.

number of watermarks	$\alpha = 0.1$	$\alpha = 0.2$	$\alpha = 0.3$	$\alpha = 0.4$	$\alpha = 0.5$
1	1.00	1.00	1.00	1.00	1.00
2	0.98	0.98	0.97	0.97	0.97
3	0.95	0.94	0.94	0.94	0.93
4	0.93	0.92	0.92	0.92	0.91
5	0.91	0.91	0.91	0.90	0.89
6	0.89	0.88	0.88	0.87	0.86
7	0.86	0.86	0.85	0.84	0.82
8	0.85	0.84	0.83	0.82	0.81
9	0.83	0.83	0.82	0.80	0.79
10	0.82	0.82	0.80	0.79	0.77
11	0.80	0.80	0.78	0.77	0.75
12	0.79	0.78	0.77	0.75	0.73
13	0.78	0.77	0.76	0.74	0.72
14	0.77	0.76	0.75	0.73	0.71
15	0.75	0.74	0.72	0.70	0.67
16	0.74	0.73	0.71	0.68	0.66
17	0.73	0.72	0.70	0.68	0.65
18	0.72	0.71	0.69	0.66	0.63
19	0.71	0.70	0.68	0.65	0.62
20	0.70	0.69	0.67	0.64	0.61
21	0.69	0.68	0.66	0.63	0.60
22	0.68	0.67	0.65	0.62	0.59
23	0.67	0.66	0.64	0.61	0.58
24	0.66	0.65	0.62	0.59	0.56
25	0.66	0.64	0.62	0.59	0.56
26	0.65	0.63	0.61	0.58	0.55
27	0.64	0.63	0.60	0.58	0.55
28	0.64	0.62	0.60	0.57	0.54
29	0.63	0.61	0.59	0.56	0.53
30	0.62	0.61	0.58	0.55	0.52

combines descrambling subkeys into a descrambling key with fingerprint embedded. By using he descrambling key, a scrambled video becomes a fingerprinted video designated to the user. In general, embedding fingerprints may often add some kinds of noise into the video contents. According to the experiment results, when the fingerprint data is less than 22% of the date size in a image or a video frame, their PSNR can be 35 or higher.

Chapter 6

Application: Web-Based Multimedia News Archive

6.1 Introduction

Due to the recent advances of web technology on multimedia, creating a web-based multimedia news system becomes possible. Such a web-based service can provide a well-organized daily news list, convenient searching mechanism, and rich multimedia contents. And most importantly, a system that can generate contents fully automated. Because of the ease by which multimedia data can be duplicated and distributed, a content server needs effective copyright protection tools. Thus, as an application of the proposed watermarking and fingerprinting approaches, this chapter proposes a fully automated web-based TV-news system. a systematic methodology that can automatically generate semantic labels from news video, and statistical methods to discover hidden information. We intend to expect that the following significance will come to exist.

- Although web-news provides another efficient way to access news, watching TV-news already becomes habit of many people. Beside this, most of web-news system can only provide text-based news.
- There are so many channels providing TV-news. People need more information for searching like-minded channel.

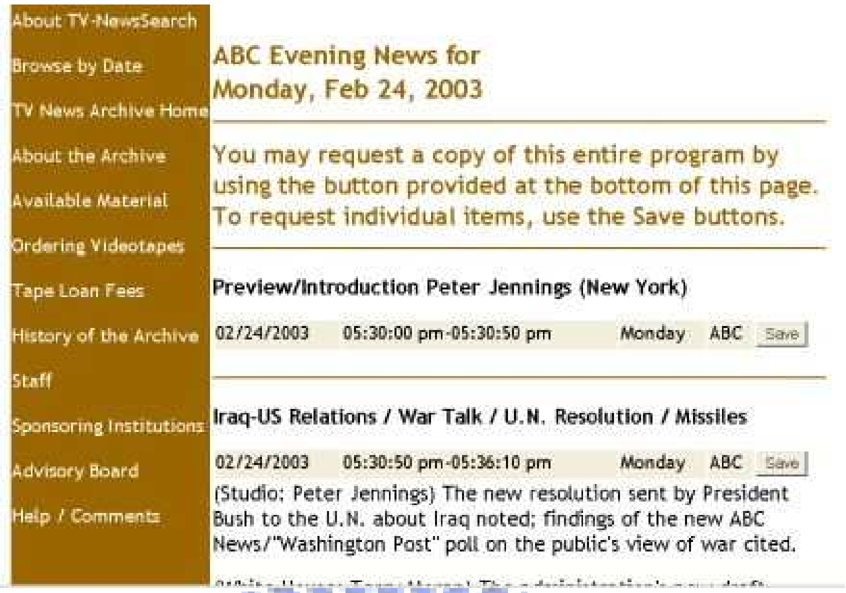
- Although almost every channel announced that they are dispassion, real dispassion is hard to archive with human editing. We need some evaluation to check if the channel is really dispassion.

Moreover, this TV news archive is used as a practical application of the proposed progressive image watermarking and video fingerprinting.

The rest of this chapter is organized as follows. First, some related works will be discussed in Section 6.2. Then, an overview of the proposed TV-news archive is presented in Section 6.3. In Section 6.4, methods of generating necessary semantic labels from the recording TV news video are presented. Section 6.5 focus on describing the information mining from these semantic labels. Section 6.6 introduces the overall concepts of the multimedia TV news archive. Finally, summary and concluding remarks are given in Section 6.7.

6.2 Related works

Among the major sources of news program, TV has clearly had the dominant influence at least since the 1960s. Yet it is easy to find the old newspaper in microfilm in any public library, but it is impossible to find the old footage of television news in the same library. TV news archive has existed in the United States for 35 years. Paul C. Simpson founded the Vanderbilt University Television News archive in 1968. In [39], a team in University of Missouri-Columbia decided to do a content analysis of the three US network coverage of the 1989 Tiananmen Massacre, they located these news items in the Vanderbilt Archive Index as shown in Fig. 6.1. The Vanderbilt archive promptly provided the 11-hour video clips all related to the Tiananmen Massacre. At the same time, the Missourian team also planned to do a comparable study of Taiwanese reportage on Tiananmen Massacre. But the equivalent material of the Vanderbilt archive did not exist in Taiwan then. Therefore, that study only contained the US perspective of the Tiananmen Massacre. In this chapter, we propose an integrated methodology for the construction and information mining on a multimedia TV news archive in Taiwan. As described in [40, 41], a fully automated



About TV-NewsSearch
Browse by Date
TV News Archive Home
About the Archive
Available Material
Ordering Videotapes
Tape Loan Fees
History of the Archive
Staff
Sponsoring Institutions
Advisory Board
Help / Comments

ABC Evening News for
Monday, Feb 24, 2003

You may request a copy of this entire program by using the button provided at the bottom of this page. To request individual items, use the Save buttons.

Preview/Introduction Peter Jennings (New York)

02/24/2003 05:30:00 pm-05:30:50 pm Monday ABC Save

Iraq-US Relations / War Talk / U.N. Resolution / Missiles

02/24/2003 05:30:50 pm-05:36:10 pm Monday ABC Save

(Studio: Peter Jennings) The new resolution sent by President Bush to the U.N. about Iraq noted; findings of the new ABC News/"Washington Post" poll on the public's view of war cited.

Figure 6.1: A sample of evening news index from the Vanderbilt TV news archive.

web-based TV-news system were implemented to achieve the following goals:

1. Academic and applied aspects: This archive will greatly improve the quality of TV news. As Dan Rather, the CBS anchorman, once mentioned that he lives with two burdens -*the ratings and the Vanderbilt Television News Archive*. Therefore, once the archive is there, the researchers and the public will do some content analysis on the TV news, and the journalists will be more careful in what they report.
2. Timing factor: Vanderbilt archive started its project with Betacam videotapes in 1968. There will be a problem of preservation because these tapes deteriorate along the years. Today, we can save all the TV news in hard disc, VCD or DVD.

Infomedia[42] is an integrated project launched in Carnegie Mellon university. Its overall goal is to use modern AI techniques to archive video and film media. VACE-II[43], a sub-project of Infomedia, automatically detects, extracts, and edits highly interested people, patterns, and story evolves and trends in visual content from news video.

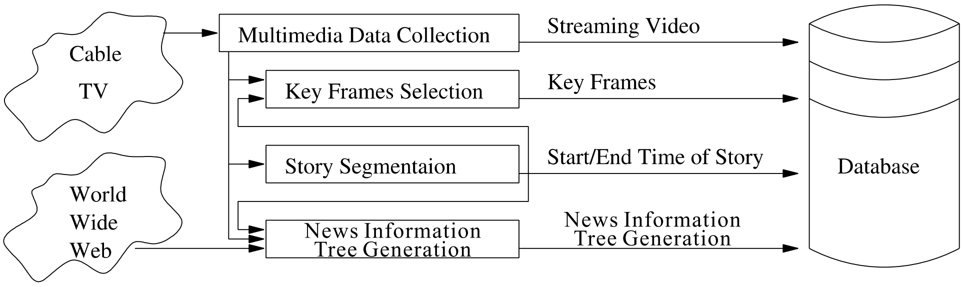


Figure 6.2: Flow chart of automatic news content generation.

6.3 A fully automated web-based TV-news system

A fully automated Web-based TV-News System[40, 41] consists of three modules: (1) TV news video acquisition, (2) news content analysis, and (3) user interface for news query, search and retrieval. In this section, video acquisition method, user interface and some content analysis methods such as shot detection and key frame extraction are presented. The news content analysis schemes will be discussed in Section 6.4 in detail.

6.3.1 An overview of the TV-news system

Using digital multimedia techniques to create TV news programs has been a new trend for news media production system. However, TV news has been broadcasted for years, a lot of TV news contents are saved and preserved in thousands of news videotapes. Thus, an automatic hierarchy news generating system is definitely necessary to produce multimedia contents from these tapes. Although there are difficulties to retrieve contents from video, the maturation of multimedia and pattern recognition techniques signals we are now able to conquer all the problems. In general, shot detection [44], speaker identification[45], video optical character recognition (video OCR)[46, 47], and data mining techniques are needed for video analysis and multimedia content generation.

The flow chart of automatic news content generation is depicted in Fig. 6.2. There are two input sources: Cable TV and World Wide Web. At first, TV-news program video is recorded to produce high-quality video for analysis, and to generate streaming video for web browsing. The news video is fed into modules for story segmentation, key-frame selection and news information tree generating. Then, if the owner needs to protect the content, watermarks are embedded into key frames and the news video are encrypted with scrambling key, as described in Section 5.3.2. Finally, streaming video, key frames and news information tree are stored in a database. We will discuss these technologies in detail in the following sections.

A user could requests through the proposed TV-news service web site to search story by keywords or browse daily news. The detail of web-interface design will be given in Section 6.3.4.

6.3.2 Multimedia data acquisition

In order to automatically generate necessary contents, news video and scripts are collected at the beginning of the work. Then, the captured data is transformed into suitable format before being applied to the following analysis. For general content analysis work, MPEG-1, a well-defined open standard of fair quality video, is the format we need the most. Except MPEG-1 video, high quality and well-recorded images are needed for close caption extraction and video OCR. For this purpose, the captured TV frames are sampled into portable pixel map (PPM) images. Besides, the system encodes captured news video into ASF format for transiting video over Internet of various bandwidths. The advancement of computer hardware makes capturing and encoding video into three different video-formats at the same time to be possible. In addition, encoding video into all desired formats simultaneously brings many of advantages, such as processing time efficient and event synchronizing accurate.

A robot like, web searching software was also developed to automatically fetches news scripts from net-news web sites.

6.3.3 Key frame extraction

To efficiently browse news story without downloading a whole news video, a set of key frames are selected from sampled video images. The main idea is that the system firstly

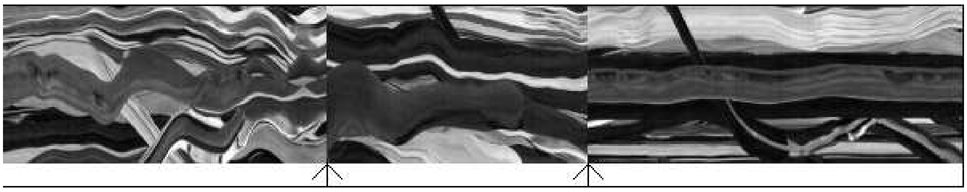


Figure 6.3: An example of spatio-temporal slice. The locations indicated by arrow symbols are just the shot-change locations.

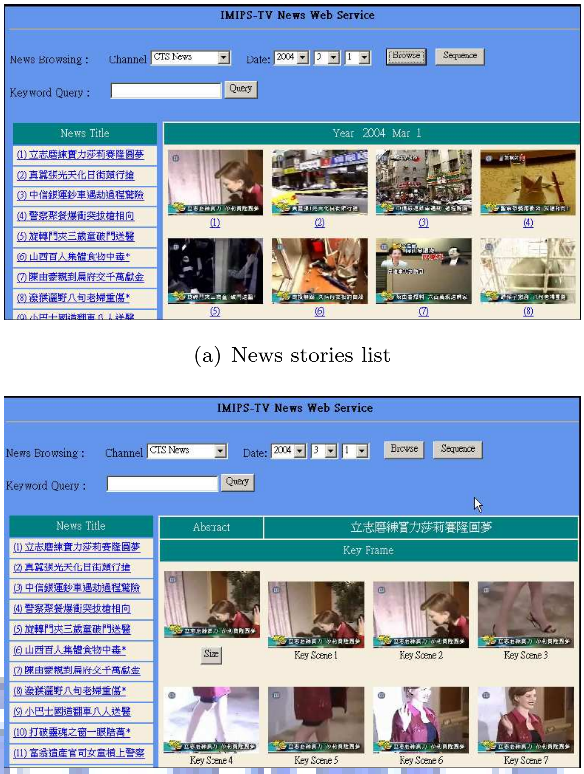
cuts video into several series frame sets, named shots, and then picks frames from each shot. The spatio-temporal slice method, proposed by Wah [44] presents the spatio and temporal relationship of video sequences. Because the shot-change brings clear edges in spatio-temporal slice, the shot change locations can be easily detected by conventional edge detection algorithm. An example of spatio-temporal(ST) slice is shown in Fig 6.3. Two apparent vertical edges divide ST slice into three pieces. These two vertical lines corresponds to the time lines of shot changes.

To catch motion activities of a news story as much as possible, we extract key frames from each shot, and high motion scenes. Then, watermarks are embedded into key frames by schemes proposed in Section 4.3.1. On the other hand, time lines of shot changes are preserved for fingerprinting procedure proposed in Section 5 so that clients' fingerprints can be changed along with a shot change.

6.3.4 Web-based user interface

This section presents the design and implementation of web-base user interface. First, we would like to briefly describe three servers in the web TV-News system. These servers are database server, web server, and media server. The data generated in Section 6.4 are stored in the SQL database. Web server accepts users' requests of looking up contents from database server, and then composes and returns the requested pages to users. When users acquire a news video, web server redirects the request to media server. The media server then supplies required video.

At the prototype web site, users can browse news stories by date, or query desired news story by assigning keywords. After opening the starting page, there are two links



(1) 立志磨練實力莎莉賽隆圓夢
(2) 真囂張光天化日街頭行搶
(3) 中信銀運鈔車遇劫過程驚險
(4) 警察聚餐爆衝突拔槍相向
(5) 旋轉門夾三歲童破門送醫
(6) 山西百人集體食物中毒*
(7) 陳由豪親到扁府交千萬獻金
(8) 潑猴灑野八旬老婦重傷*
(9) 小巴士國道翻車八人送醫

(a) News stories list

立志磨練實力莎莉賽隆圓夢

(1) 立志磨練實力莎莉賽隆圓夢
(2) 真囂張光天化日街頭行搶
(3) 中信銀運鈔車遇劫過程驚險
(4) 警察聚餐爆衝突拔槍相向
(5) 旋轉門夾三歲童破門送醫
(6) 山西百人集體食物中毒*
(7) 陳由豪親到扁府交千萬獻金
(8) 潑猴灑野八旬老婦重傷*
(9) 小巴士國道翻車八人送醫
(10) 打破靈魂之窗一眼賠萬*
(11) 富翁遺產官司女童槓上警察

(b) Key frames and video of story

Figure 6.4: User interface of the proposed web-based multimedia news archive.

for PC users and PDA users respectively to begin the news service. The following explanation is for PC users. The main service page is divided into two partitions - top and bottom frames. Users can specify date and channel on the top frame, and browse news stories list of specified date and channel on the bottom one, as shown in Fig. 6.4(a). Headlines are listed on the bottom-left. And the images displayed on the bottom-right are the representative key-frame of each news story. Users can select stories by clicking on headlines or representative key-frames. The story selected is shown on the bottom frame (see Fig. 6.4(b)). The key frames of this story are presented on the bottom-right all at once. Besides, a embedded window for playing video is also displayed on the top-left corner.

In addition to browsing related news stories by date and channel, users can assign several keywords for related news stories in the database of the proposed web TV-News system. Stories that matched users' requirements are then listed at the bottom-left frame.

6.4 News information tree generation

The most important things in news story writing are that journalists commonly refer to as the 5 W's: who, what, when, where and why. These questions are crucial for catching a reader's attention and introducing the essential facts of the story. Standing on this basic rules, the news archive system introduced in Section 6.6, are further improved to extract more information from a recorded news video. A *news information tree* is suggested to structure the contents of recorded video clips for helping the user focus on specific news information, and information that is a little more general.

6.4.1 News information tree

Figure 6.5 illustrates a hierarchical structure of a news information tree. The hierarchical tree contains five types of video information records: (1) Date (when), (2) Channel (where), (3) Title (what), (4) Content (how), and (5) Commercial. The title record contains the starting time, length, and brief description of the corresponding video clips. The content record can also be further divided into the following sub-records: (a) on-site locations, (b) interview, and (c) tables or quoted word.

6.4.2 Analysis units

Usually, a TV-news program contains the following items: news stories, commercials and weather reports. Complete description of shot detection and scene segmentation can be found in [48]. A flow chart of TV-news program segmentation is shown in Fig. 6.6, and brief introduction are described as follows. Among various scene shots, anchor video clips are detected first. In general, anchor segments are the most appeared video clips, thus we propose to use BIC[49], an unsupervised method, to cluster anchor segments from the other clips. As shown in Fig. 6.7, this method contains the following procedures:

1. The MFCC audio feature sequence X is generate from input audio at first.

2. BIC segments X into X_1,X_2,...,X_n.

3. These segments then are clustered as several clusters C_1,C_2,...,C_m.

4. The cluster containing most clip segments is the set of anchor clips.

After locating each anchor shot, a SVM model based video classifier [50] is used to detect weather report shots. Finally, commercials are detected and separated from on-site news stories. The following feature detecting techniques are integrated to achieve a high performance commercial detector:

- The variation rate of zero crossing rate.
- Short time energy.
- Shot change rate.
- Clip length.

At this stage, anchor's briefing, weather reports, commercial, and the background stories are all separated and identified. For each on-site stories, as shown in Fig. 6.8, we further segment and classify each on-site scene into three categories: locations, interview and tables or quoted words (what).

Figure 6.9 shows how to partition the on-site news story into location, interview, and tables or quoted words scenes. The narration periods can be detected by using speaker identification techniques to distinguish narrator's speech voice from the rest of scenes. Detecting the screen characters regions can find the tables or quoted words scenes. Then, the rest of scenes that are not belonging to interview or tables or quoted words scenes must be the location scenes. In general, on-site narration is not active during the interview scene, as shown in Fig. 6.8, the interview scene can be distinguished from location scene. By using its special characteristics of the character regions, tables or quoted words scene can be distinguished from location scenes.

6.4.3 Semantic labels of units

This section describes how to assign each segmented unit with semantic labels. Basically, the text words for each label are extracting from text streams in close-caption.

Usually, a TV-news program often provides audience a quick overview of each news story in on-screen captions, such as names of location, people, and keywords of events, ... etc. In general, these texts are quiet enough to give enough information for labeling each segmented units.

Figure 6.10 shows how to establish a news information tree. The information tree establishing process contains two phases: story and scene phases. In story phase, all on-screen characters are recognized by video-OCR (optical character recognition) first. Then, the recognized characters or words are used to match with text-based news documents, which are usually retrieved from Internet. The title and contents of best matched text-news document not only fill out the story information record of news information tree, but also used to picking label candidates, including locations, people names, event words, quoted phrases, and tabular data, up for scene phase processing.

In scene phase, picking semantic labels up from label candidates for each scene is done. As shown as Fig. 6.11, the on-screen captions of locality scene provide location name and event descriptions. Therefore, in locality scene, the location and event words of label candidates are searched from on-screen captions to find which ones are exactly appeared.

Figure 6.12 is an example of interview scene. In interview scene, the interviewee's name and their points are always given by on-screen captions. Therefore, in interview scene, we search people name, and events word of label candidates instead.

The example of data chart scene is shown as Fig. 6.13. In general, on-screen captions fill data chart scene. These captions may present quoted sentence or tabular data. Therefore, searching for quoted sentence or tabular data in data chart scene is the major task.

6.5 Data mining on the news information tree

This section presents how and what to mine from a news information tree (NIT). In Section 6.5.1, we propose to mine the favored or preferred news contents of a TV-station. The news information tree can also be used to track the evolution of a series of news stories (see Section 6.5.2). In addition, the mining results from the NIT and the realtime ratings can be combined to provide TV-news commercial buyers a very useful guidance.

6.5.1 Mine the news preference of a TV station

Generally speaking, a TV-station arranges the broadcasting sequence of each story in a news program according to their impact and attractiveness to audience. In fact, a preferred news story often gets more time on the air. By analyzing the sequence order and the length of stories, the preferred or the favored news stories of a TV-station can be roughly estimated or judged. Mining the NIT to extract favored or preferred types of news story from a TV station will help audience to find favor news channel.

The proposed news mining method is described as follows. Given N sets of keywords, $K_1, K_2, ..., K_i, ..., K_N$, which correspond to N news topics (or subjects), let the following delta function $\delta(k, K_i)$ define the relations between a keyword k and a keyword set K_i:

$$\delta(k, K_i) = \begin{cases} 1, & \text{if keyword } k \in K_i \\ 0, & \text{otherwise.} \end{cases}$$

1. Extract keywords $\{k^l_{s_j};\ l = 1, \cdots, L_j\}$ from a scene unit s_j in a news program.

2. For each scene units s_j, compute its association frequency $F(K_i|s_j)$ with respect to a subject K_i,

$$F(K_i|s_j) = \frac{\sum_{l=1}^{L_j} \delta(k^l_{s_j}, K_i)}{\sum_{i=1}^{N} \sum_{l=1}^{L_j} \delta(k^l_{s_j}, K_i)}$$

3. Compute the the association frequency of news program $F_d(t|K_i)$ at time t and day d:

$$F_d(t, K_i) = \begin{cases} F(K_i, s_1), & \text{for } s_1.start \leq t \leq s_1.end \\ F(K_i, s_2), & \text{for } s_2.start \leq t \leq s_2.end \\ \quad\vdots & \qquad\vdots \\ F(K_i, s_M), & \text{for } s_M.start \leq t \leq s_M.end, \end{cases}$$

   where $s_j.start$ and $s_j.end$ are the start and the end time of a scene unit s_j in a news program at time t of the day d.

The associated frequency distribution from one segment of news program is not enough to represent the overall preference or trend of a news channel; thus long term statistics is needed. By accumulating a longer period (say one month) of associated frequency of news subjects, the preference of a channel can be discovered. As shown in Fig. 6.14, keywords that are related to social news, political news, and entertainment news are applied to associate with and to accumulate frequency of news topics. As we can see in this example, the monitored news channel favors social and political news more than entertainment news.

6.5.2 The evolution of a series of news stories

The evolution of a news story can also be mined from the news information tree. By associating the keywords of a specific event with recorded news scenes over a period of days, then the accumulated association frequency of matched scene units presents an overall developing and progressing of the specific news stories. Figure 6.15 shows a sort of life-cycle of a particular news events. In addition, the spreading of the specific events to other areas, e.g., cities, counties, countries, etc., can also be retrieved from the associated names of locations in the matched scene units. For example, one can query a news story by using a particular people's name, then the person's daily schedule and/or whereabouts can be retrieved from the the recorded NIT.

6.5.3 The mining on TV commercial

Beside background stories, commercial records are also valuable information. Huang et al., [48] proposed commercial detecting and identifying methods in TV video clips. When a commercial frame contains image keywords in a video frame, video OCR techniques can be used to extract keywords to label the corresponding video clips. Otherwise, keyblock-based image retrieval methods [51] may be utilized to represent and to identify each commercial clips. However, manual annotation is needed to label the keyblock. By gathering statistical information of these labels and keywords in news programs, cross relationship between TV commercials, realtime ratings, and news stories can be observed and analyzed to achieve a useful marketing database. Two example areas, *customer* *modeling* and *cross-selling*, in database marketing are discussed in the followings.

Customer modeling The basic idea behind customer (i.e., the commercial buyers and news audience) modeling is to improve audience response rates by targeting prospects that are predicted as most likely to respond to a particular advertisement or promotion. This is achieved by building a model to predict the likelihood that groups of news audience will respond based on news type, viewing time and news channels as well as previous viewing behavior. In addition, by targeting more effectively to prospects and existing commercial buyers, TV station operators can improve and strengthen customer relationships. The customer can perceives more value in TV news and commercials (i.e., both commercial buyers and news audience receive only products and/or services of interest to them).

Cross selling The basic idea behind cross selling is to leverage the existing customer base by selling them additional products (commercial time slots) and/or news services. By analyzing the groups of products or services that are commonly purchased together and predicting each customer's affinity towards different products using historical data, a TV-station can maximize its selling potential to the existing customers. Cross selling is one of the important areas in database marketing where predictive data mining techniques can be successfully applied. Using historical purchase data of different products from the customer database along with news type, viewing time and news channels, commercial buyers can identify their products that are most likely to be of interest to targeted news audience. Similarly, for each type of product (i.e., commercial or groups of commercials), a ranked list of different types of news or groups of audience, that are most likely to be attracted to that product. Then, arrangement of commercials with matched types of news to achieve a high likelihood of audience response rate.

6.6 TV-news archive

A fully automated Web-based TV-News System[40, 41] consists of three modules: (1) TV news video acquisition, (2) news content analysis, and (3) user interface for news query, search and retrieval. Figure 6.16 depicts the overall architecture and interaction of these three modules. The major tasks of the acquisition module are to record TV news programs in a proper video format, and to fetch related news text contents from Internet webs. Content analysis module segments the recorded news video into story based units, and extracts news title and keywords from each story unit. Providing a friendly querying and browsing environment for retrieving interested news stories is the most important task of the user interface module.

The overall content processing and analyzing are briefly described in the followings. At the beginning, a TV news program is captured and encoded into stream video format. The recorded streaming video is named and tagged first, and then stored in database. In the meantime, a shot detector is used to segment the streaming video into scene based shots for key-frame extraction and generation. Within a shot, speaker identification techniques are then applied to detect anchor frames. The close captions in the anchor frames are then extracted and recognized by using video OCR techniques[41] as candidates for the title and keywords of each story units. The extracted keywords can then be used in matching with (1) the Internet news stories to construct links between TV news stories and Internet news, and (2) the users' query text words for retrieving their interested news stories.

6.7 Summary

This chapter addresses techniques and possible applications of fully automated information mining on a multimedia TV-news archive. The proposed automated information mining contain the following processes: (1) segmenting a TV-news program video recording into scene clips, (2) using video OCR to extract and recognize close-caption and/or image characters into keywords for each scenes, (3) using keywords to generating semantic labels for each scenes, and (4) segmenting commercial video clips from news clips. Information associated with various labels and scenes (e.g., the starting and ending time of a scene) are stored in the proposed *news information tree*. Performing statistical analysis on the data items in the news information tree can reveal hidden information, like popular channels and evolution of some hot news stories. These information can help general multitude in finding their favored or desired news-channel, searching focal point person, tracking hot news stories, ..., and so on.

The proposed web based TV news archive were also used as a test bed of the proposed image watermarking and video fingerprinting methods. Watermarks can be embedded automatically in key frame extraction procedure, and information on shot change can be used as signals such that the server can change clients' descrambling key (and also the fingerprints) to increase the security [34]. In multimedia applications, computational efficiency is one of the important issues. Our testing shows that the proposed watermarking and fingerprinting methods are efficient to real world multimedia applications.

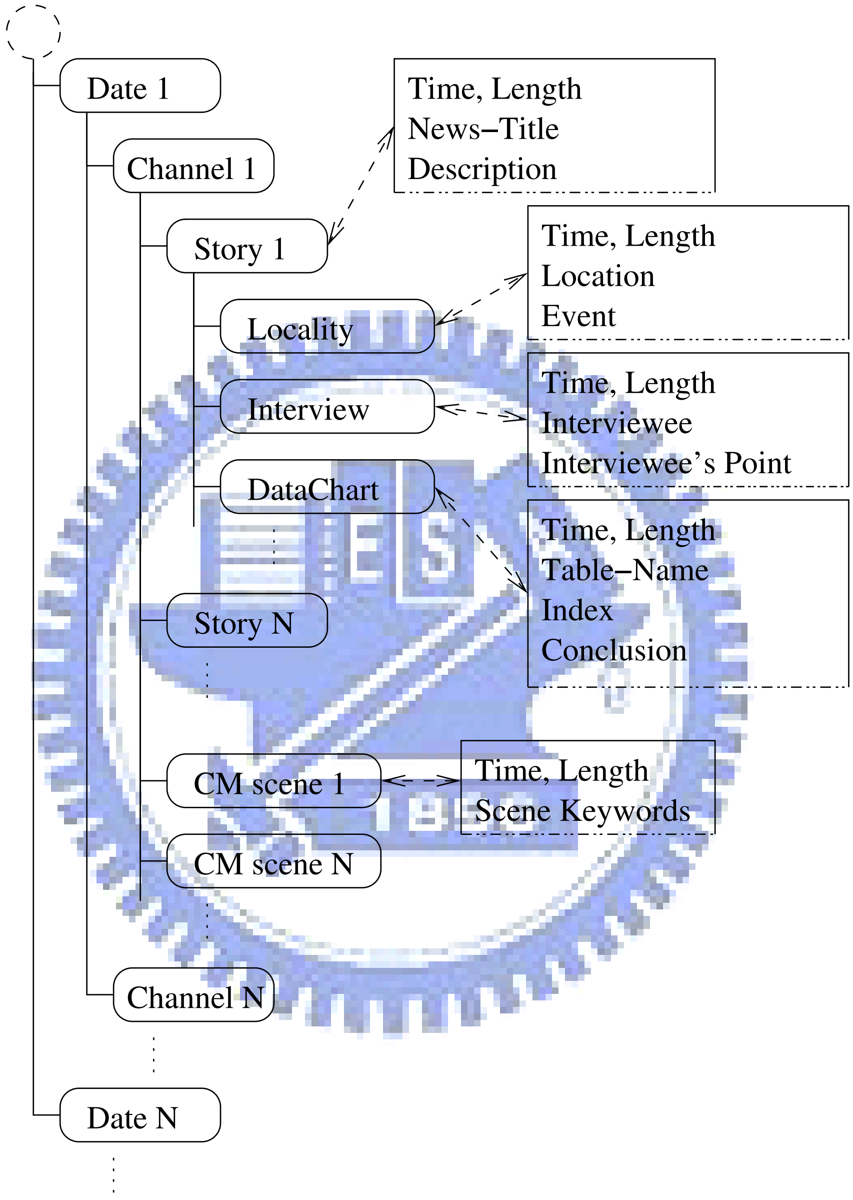


Figure 6.5: The data structure of a news information tree.

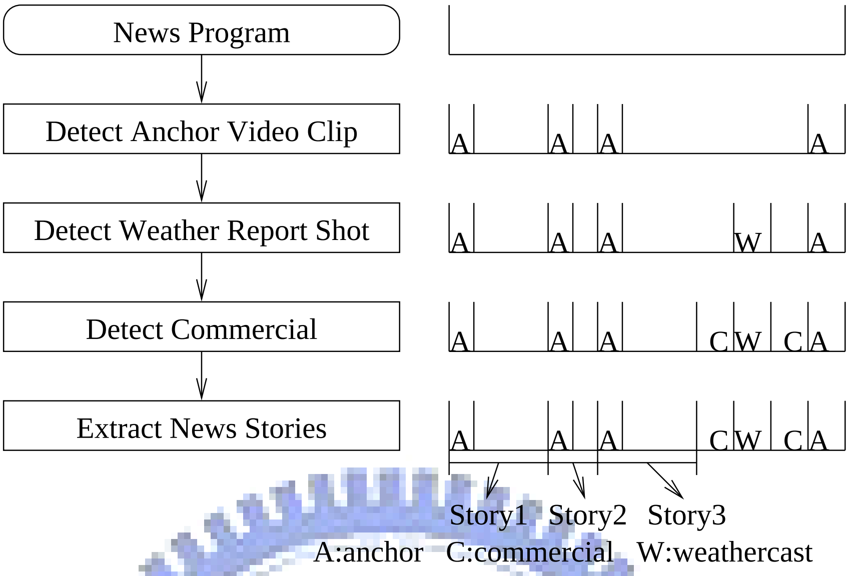


Figure 6.6: The flow diagram of the proposed news story analysis and information extraction processes.

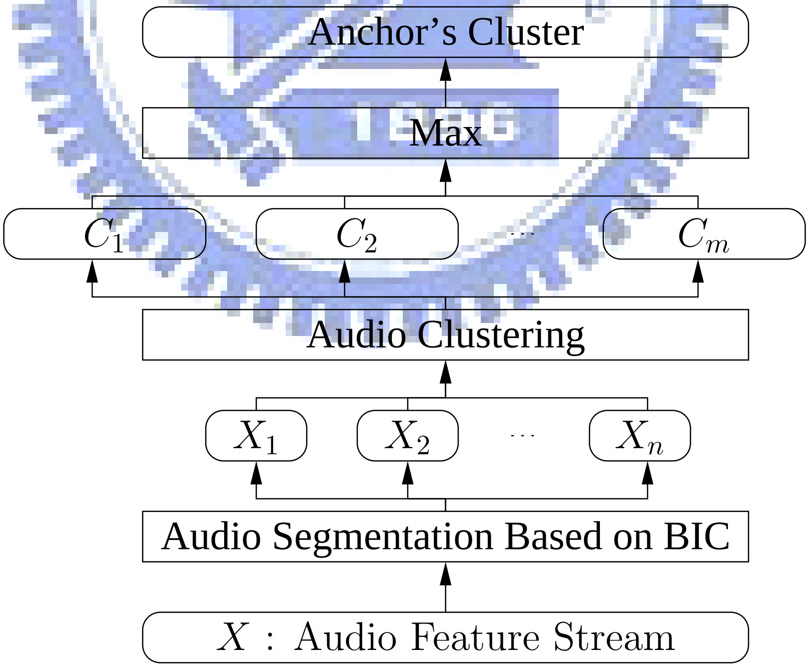


Figure 6.7: The flow diagram of a BIC-based audio segmentation method.

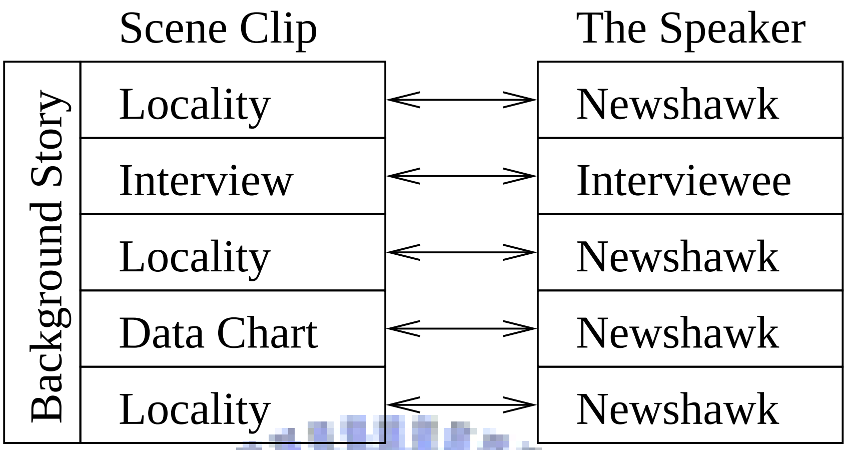


Figure 6.8: The general structure of a news story. On-site scene story contains three major news contents: locations, interview and tables or quoted words.

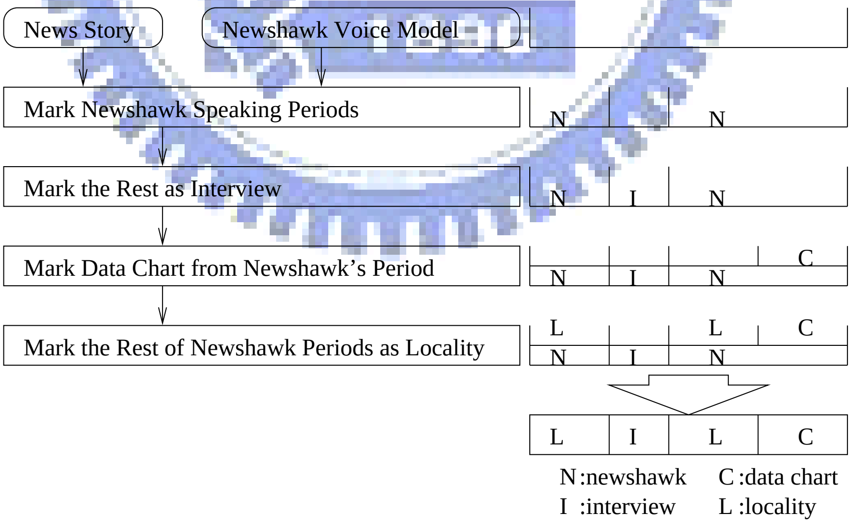

News Story
Newshawk Voice Model
Mark Newshawk Speaking Periods
N
N
Mark the Rest as Interview
N
I
N
Mark Data Chart from Newshawk's Period
C
N
I
N
Mark the Rest of Newshawk Periods as Locality
L
L
C
N
I
N
L
I
L
C
N :newshawk
C :data chart
I :interview
L :locality

Figure 6.9: On-site scene segmentation flow.

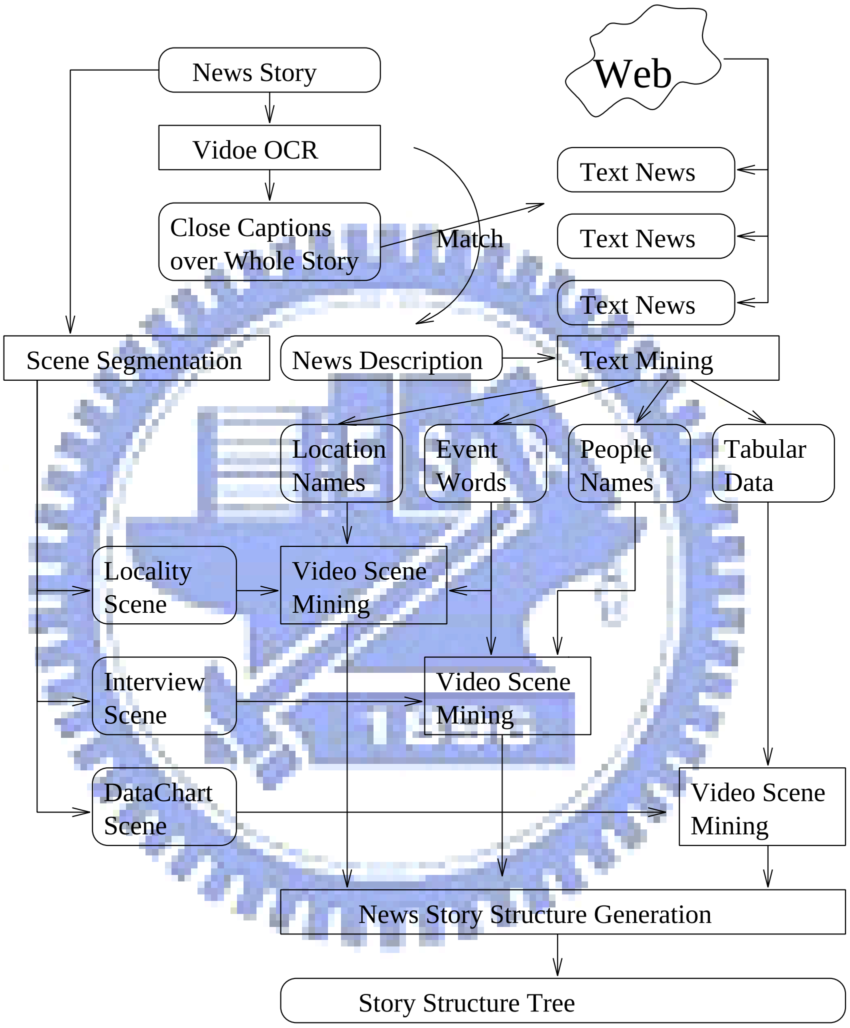


Figure 6.10: Information flow of the generation of a news information tree.

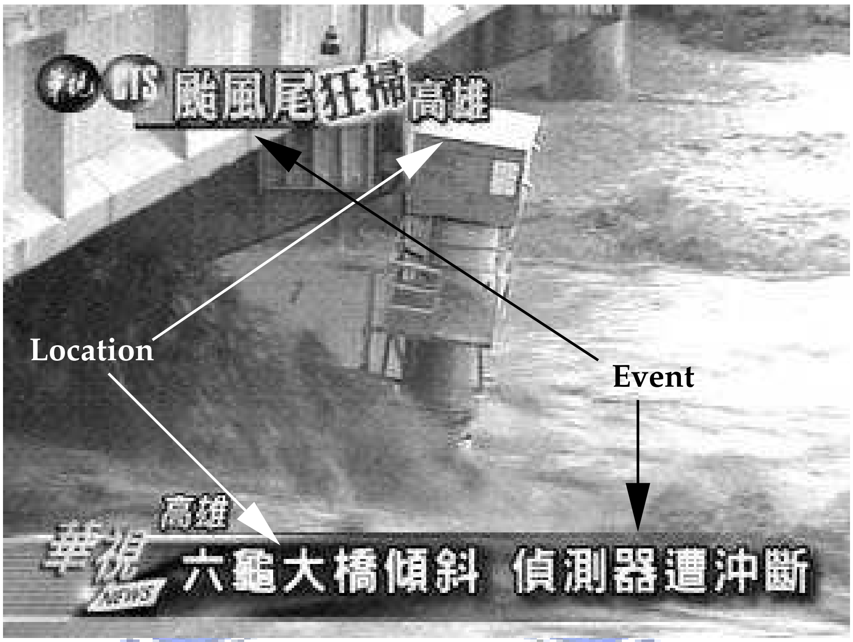


Figure 6.11: An example of locality scene frame. The locality scene is used to show where and what the news occurred. Thus, the location information and event description can be retrieved from the close-captions of a locality scene.

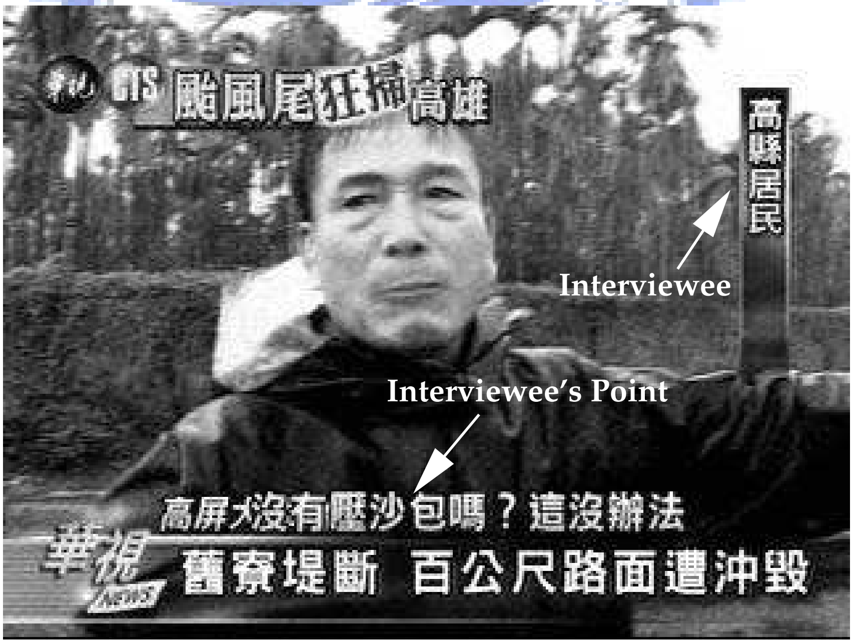


Figure 6.12: An example of interview scene frame. The interview scene is used to present the news persons' point of view. Thus, the interviewee's name and their opinion can be extracted from the screen characters or closed captions.

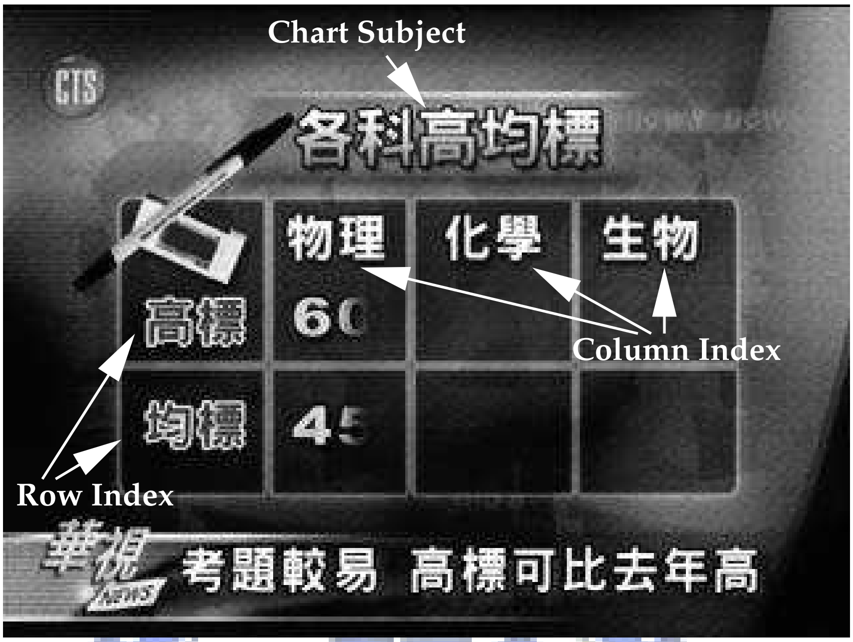


Figure 6.13: An example of data chart scene frame. The data chart scene is used to present information in a organized manner. Additional information is also available from the on screen characters.

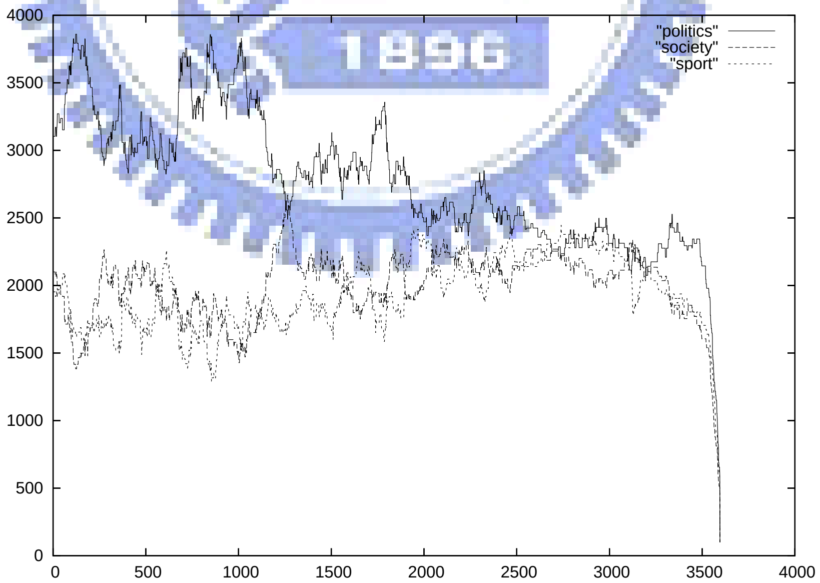


Figure 6.14: Three sets of (representative) keywords are used to associate the appearing frequency of *social*, *political* and *sport* news in a news program.

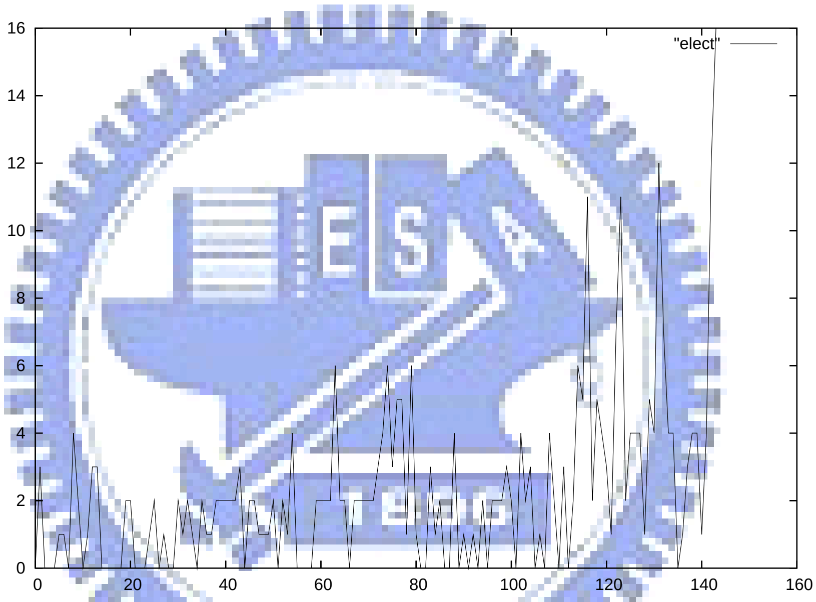


Figure 6.15: The life-cycle of a specific news events along with a period of time.

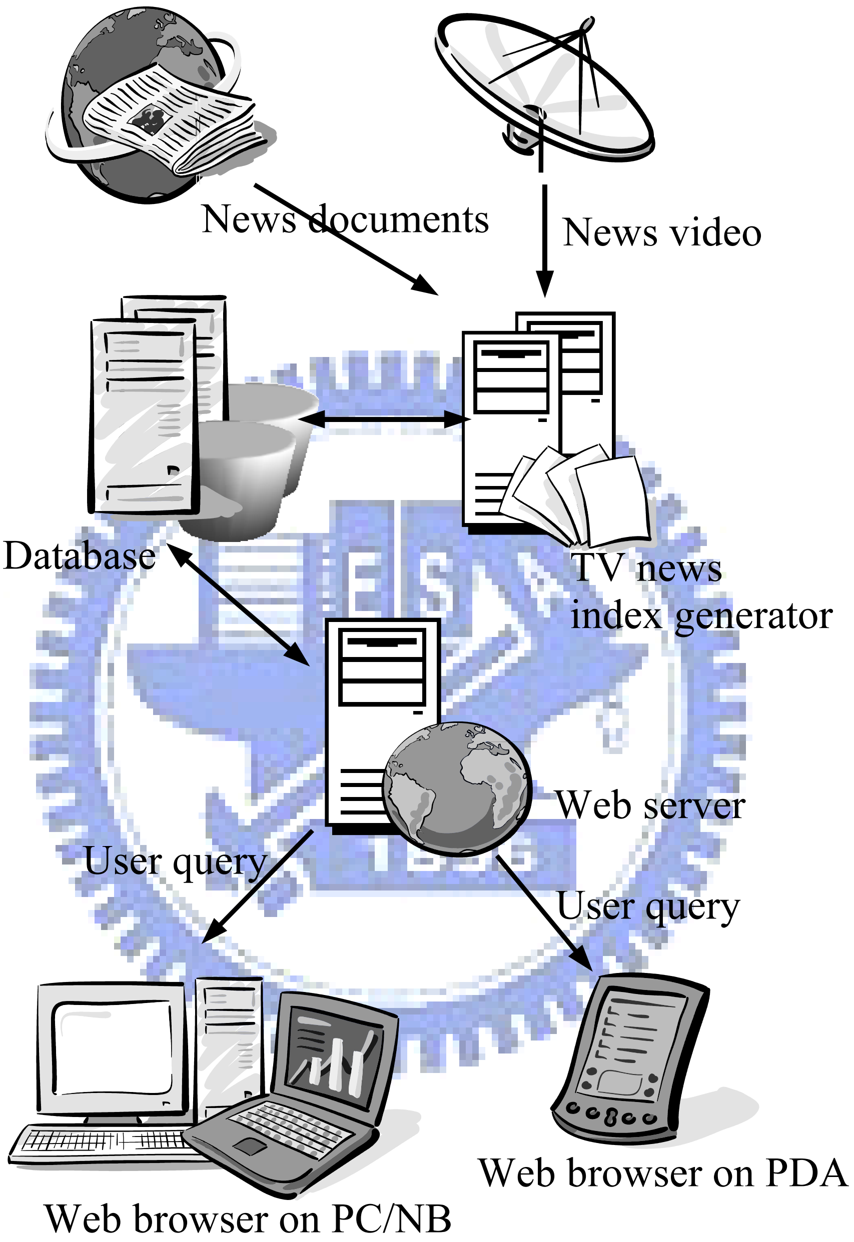


Figure 6.16: The overall architecture and information processing flow of the proposed fully automated web-based TV-news system.

Chapter 7

Conclusions and future works

7.1 Conclusions

In this dissertation, progressive image watermarking schemes and video fingerprinting schemes for a web-based multimedia news archive were proposed. Progressive transmission of images is very useful and widely used in many applications, especially in image transmission over the Internet. In this dissertation, we first propose a progressive image watermarking scheme. In this scheme, the watermark is embedded in such a way that we can retrieve part of it even when the watermarked image is still being transmitted. As transmission progresses, the retrieved watermark has a decreasing bit error rate. Our proposed methods can not only confirm the watermarked image progressively, but also intelligently select watermark modification values. The significance of the proposed method can protect digital right at distribution side, instead of at the user side. Also, when a partial of image is determined to be free of piracy suspect, the progressive detection process can be terminated to save computational resources as well as the network bandwidth.

We also propose a new video scrambling and fingerprinting approach for digital media right protection. The proposed method contains two parts: (1) video scrambling at server side, and (2) fingerprint embedding at client side. First, a content server scrambles and multicasts video contents to end users. Then by applying a (v, k, 1)-BIBD scheme, the server partitions a descrambling key into $v = O(\sqrt{n})$ descrambling subkeys, and multicasts to n users. On receiving descrambling subkeys from the content server, each user combines these subkeys into a descrambling key embedded with a fingerprint. By using he's descrambling key, a scrambled video becomes a fingerprinted video designated to the user. In general, embedding fingerprints may often generate some kinds of noise to the video contents. According to the experiment results, when the fingerprint consisting of less than 15 watermarks or watermark strength α is less than 0.4, the PSNR of video frames can be 35 or higher. This is visually acceptable.

Finally, an integrated information mining techniques for multimedia TV-news archive is addressed. The utilizes techniques from the fields of acoustic, image, and video analysis, for information on news story title, newsman and scene identification. By using acoustic analysis, a news program can be partitioned into news and commercial clips, with 90% accuracy on a data set of 400 hours TV-news recorded off the air from July 2003 to August of 2004. By applying speaker identification and/or image detection techniques, each news stories can be segmented with an accuracy of 96%. On screen captions or subtitles are recognized by OCR techniques to produce the text title of each news stories. The extracted title words can be used to link or to navigate more related news contents on the WWW. In cooperation with facial and scene analysis and recognition techniques, OCR results can provide users with multimodality query on specific news stories.

The proposed web based TV news archive were also used as a test bed of the proposed image watermarking and video fingerprinting methods. Watermarks can be embedded automatically when key frames are extracted, and information on shot change can be used as signals to change clients' fingerprints to increase the security. In multimedia applications, computational efficiency is one of the important issues. Our testing shows that the proposed watermarking and fingerprinting methods are efficient to real world applications.

7.2 Future works

Following the research described in this dissertation, future works can be focused on the following topics:

- A watermark detection system can be developed to actively detect watermarked images on the Internet. Moreover, information about images, such as URL and a few wavelet coefficients can be cached so that a user can *query* his/her watermarked image with the watermark.

- Up to the present, most image watermarking approaches have embedded watermarks in fixed frequency subbands. However, as shown in the experimental results in Section 3.4 and Section 4.4, the proper subbands for watermark embedding are different among images. Thus, based on the proposed methods, an progressive watermarking scheme that can select frequency subbands adaptively should be developed.

- Up to now, Video multicasting have not been broadly used on the Internet. After a commonly used protocol or standard is appear, we will modify the proposed fingerprinting method slightly to fit the multicast method.

- The proposed TV news archive is mainly for Chinese news in Taiwan. In the future, TV news in other languages will be processed, recorded and mined as well.

Bibliography

[1] I. J. Cox, M. L. Miller, and J. A. Bloom, *Digital Watermarking: Principles & Practice.* New York: Morgan Kaufman Publishers, October 2001.

[2] I. J. Cox, M. L. Miller, J. A. Bloom, J. Fridrich, and T. Kalker, *Digital Watermarking and Steganography.* London: Elsevier Science & Technology, November 2007.

[3] E. Koch and J. Zhao, "Towards robust and hidden image copyright labeling," in *Proc. of 1995 IEEE Workshop on Nonlinear Signal and Image Processing*, Halkidiki, Greece, June 1995, pp. 452–455.

[4] C.-T. Hsu and J.-L. Wu, "Hidden digital watermarks in images," *IEEE Trans. Image Processing*, vol. 8, no. 1, pp. 58–68, January 1999.

[5] J. Zhao and E. Koch, "Embedding robust labels into images for copyright protection," in *International Congress on Intellectual Property Rights for Specialised Information, Knowledge and New Technologies*, Vienna, Austria, 21-25 1995.

[6] F. Y. Duan, I. King, L.-W. W. Chan, and L. Xu, "Intra-block max-min algorithm for embedding robust digital watermark into images," *Multimedia Information Analysis and Retrieval*, pp. 255–264, 1998.

[7] W. Bender, D. Gruhl, N. Morimoto, and A. Lu, "Techniques for data hiding," *IBM Systems Journal*, vol. 35, no. 3,4, pp. 313–336, 1996.

[8] J. K. Su, J. J. Eggers, and B. Girod, "Capacity of digital watermarks subjected to an optimal collusion attack," in *Proc. Eur. Signal Process. Conf.*, vol. 4, September 2000, pp. 1981–1984.

[9] D. Boneh and J. Shaw, “Collusion-secure fingerprinting for digital data,” *IEEE Trans. Information Theory*, vol. 44, no. 5, pp. 1897–1905, September 1998.

[10] B. P. Hans-Jurgen Guth, “Error- and collusion-secure fingerprinting for digital data,” in *Prelim. Proc. 3rd Intl. Information Hiding Workshop*, Dresden, Germany, October 1999, pp. 134–145.

[11] S. Voloshynovskiy, S. Pereira, T. Pun, J. . Eggers, and J. K. Su, “New paradigms for effective multicasting and fingerprinting of entertainment media,” *IEEE Communications Magazine*, vol. 43, no. 6, pp. 77–84, June 2005.

[12] M. Kutter and F. A. P. Petitcolas, “A fair benchmark for image watermarking systems,” in *Proc. SPIE Security and Watermarking of Multimedia Contents*, vol. 3657, San Jose, CA, USA, 25–27 Jan. 1999, pp. 226–239.

[13] W. Zhu, Z. Xiong, and Y. Q. Zhang, “Multiresolution watermarking for images and video,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 9, no. 4, pp. 545–550, 1999.

[14] Y.-K. Chee, “Survey of progressive image transmission methods,” *International Journal Of Imaging Systems And Technology*, vol. 10, no. 1, pp. 3–19, 1999.

[15] S. Voloshynovskiy, S. Pereira, T. Pun, J. . Eggers, and J. K. Su, “Attacks on digital watermarks: Classification, estimation-based attacks, and benchmarks,” *IEEE Communications Magazine*, vol. 39, no. 8, pp. 118–126, August 2001.

[16] T. P. chun Chen and T. Chen, “Progressive image watermarking,” in *IEEE Iternational Conference on Multimedia and Expo*, vol. 2, July 2000, pp. 1025–1028.

[17] A. Jayawardena and P. Lenders, “Embedding multiresolution binary images into wavelet domain multiresolution binary watermark channels for copyright enforcement,” in *Proceedings of the Acoustics, Speech, and Signal Processing 2000*, vol. 4, June 2000, pp. 1983–1986.

[18] Z.-N. Li and M. S. Drew, Eds., *Fundamentals of Multimedia.* Prentice-Hall, October 2003.

[19] Y.-H. Chen, J.-M. Su, H. Fu, H.-C. Huang, and H. Pao, "Adaptive watermarking using relationships between wavelet coefficients," in *IEEE International Symposium on Circuits and Systems*, vol. 5, May 2005, pp. 4979–4982.

[20] E. K. P. Chong and S. H. Zak, *An Introduction to Optimization*, 2nd ed. New York: John Wiley and Sons, 2001.

[21] J. C.-I. Chuang and M. A. Sirbu, "Pricing multicast communication: A cost-based approach," *Telecommunication Systems*, vol. 17, no. 3, pp. 281–297, 2001.

[22] H. Zhao and K. Liu, "Bandwidth efficient fingerprint multicast for video streaming," in *Proceedings of the 2004 IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 5, May 2004, pp. 849–852.

[23] T.-L. Wu and S. F. Wu, "Selective encryption and watermarking of mpeg video," in *International Conference on Image Science, Systems and Technology, CISST97*, June 1997, pp. 261–269.

[24] H. hua Chu, L. Qiao, and K. Nahrstedt, "A secure multicast protocol with copyright protection," *ACM Computer Communication Review*, vol. 32, Issue 2, pp. 42–60, April 2002.

[25] R. Parviainen and P. Parnes, "Large scale distributed watermarking of multicast media through encryption," in *Communications and Multimedia Security*, ser. IFIP Conference Proceedings, R. Steinmetz, J. Dittmann, and M. Steinebach, Eds., vol. 192. Darmstadt, Germany: Kluwer, May 2001.

[26] D. Thanos, "Coin-video: A model for the dissemination of copyrighted video streams over open networks," in *Information Hiding: 4th International Workshop.* Pittsburgh, PA, USA: Lecture Notes in Computer Science, April 2001, pp. 169–184.

[27] B. M. Macq and J.-J.Quisquater, "Cryptology for digital tv broadcasting," *Proceedings of the IEEE*, vol. 83, no. 6, pp. 944–957, June 1995.

[28] F. Hartung and B. Girod, "Digital watermarking of mpeg-2 coded video in the bitstream domain," in *Proceedings of the 1997 IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 4, April 1997, pp. 2621–2624.

[29] J. A. Bloom, "Security and rights management in digital cinema," in *Proceedings of the 2003 International Conference on Multimedia and Expo*, vol. 2, July 2003, pp. 621–624.

[30] P. Judge and M. Ammar, "Whim: Watermarking multicast video with a hierarchy of intermediaries," *Computer Networks*, vol. 39, no. 6, pp. 699–712, August 2002.

[31] W. Luh and D. Kundur, *Digital Media Fingerprinting: Techniques and Trends.* CRC, 2004, ch. 19.

[32] D. Kunder and K. Karthik, "Video fingerprinting and encryption principles for digital rights management," *Proceedings of the IEEE*, vol. 92, no. 6, pp. 918–932, June 2004.

[33] I. J. Cox, J. Kilian, F. T. Leighton, and T. Shamoon, "Secure spread spectrum watermarking for multimedia," *IEEE Trans. Image Processing*, vol. 6, no. 12, pp. 1673–1687, December 1997.

[34] K. Su, D. Kundur, and D. Hatzinakos, "Statistical invisibility for collusion-resistant digital video watermarking," *IEEE Transactions on Multimedia*, vol. 7, no. 1, pp. 43–51, 2005.

[35] C. C. Lindner and C. A. Rodger, *Design Theory.* Boca Raton: CRC Press LLC, 1997.

[36] J. H. Dinitz and D. R. Stinson, Eds., *Contemporary Design Theory: A Collection of Surveys.* New York: Willy, 1992.

[37] C. J. Colbourn and J. H. Dinitz, *The CRC Handbook of Combinatorial Designs*. Boca Raton: CRC Press, 2006.

[38] T.-S. Chen, C.-C. Chang, and M.-S. Hwang, "A virtual image cryptosystem based upon vector quantization," *IEEE Transactions on Image Processing*, vol. 7, no. 10, pp. 1485–1488, 1998.

[39] S. Huffman, T.-E. Yang, L. Yan, and K. Sanders, "Genie out of the bottle: Three u.s. networks report tiananmen square," in *Proceedings of the annual meeting of Association for Education in Journalism and Mass Communication*, Minneapolis, Minnesota, USA, 1990.

[40] Y. Xu, Y. Chen, C. Tseng, P. Lai, R. Hsieh, Y. Lu., Y. Shen, and H.-C. Fu, "Multimedia tv news browsing system," in *Proceedings. IEEE International Conference on Multimedia and Expo*, Taipei, Taiwan, ROC, June 2004.

[41] P.S.Lai, L.Y.Lai, T.C.Tseng, Y.H.Chen, and H.-C. Fu, "A fully automated web-based tv-news system," in *Proceedings of PCM2004*, Tokyo, Japan, Dec. 2004.

[42] "Informedia." [Online]. Available: http://www.informedia.cs.cmu.edu/

[43] "Informedia:vace-ii." [Online]. Available: http://www.informedia.cs.cmu.edu/arda/vaceII.html

[44] N. C. Wah, *Analysis of Spatio-Temporal Slices for Video Content Representation*. PhD Thesis, Hong Kong University of Science & Technology, 2000.

[45] S.-S. Cheng, Y. hong Chen, C.-L. Tseng, H.-C. Fu, and H.-T. Pao, "A self-growing probabilistic decision-based neural network with applications to anchor/speaker identification," in *Proceedings of the Second International Conference on Hybrid Intelligent Systems (HIS02)*, Santiago, Chile, 2002.

[46] T. Sato, T. Kanade, E. Hughes, and M. Smith, "Video optical character recognition for digital news archive," in *Proceedings of Workshop on Content-Based Access of Image and Video Databases*, Los Alamitos, CA, 1998, pp. 52–60.

[47] H.-C. Fu, H.-Y. Chang, Y. Y. Xu, and H.-T. Pao, “User adaptive handwriting recognition by self-growing probabilistic decision-based neural networks,” *IEEE Transactions on Neural Networks*, vol. 11, no. 6, p. 1373, 2000.

[48] P.-S. L. Tzu-Yang Huang and H.-C. Fu, “A shot-based video clip search method,” in *Proceedings. of CVGIP2004*, Taipei, Hualien, ROC, August 2004.

[49] C. Fraley and A. E. Raftery, “How many clusters? which clustering method? answers via model-based cluster analysis.” *Computer Journal*, vol. 41, pp. 578–588, 1998.

[50] S.-Y. Sun, C.L.Tseng, Y.H.Chen, S.C.Chuang, and H.C.Fu, “Cluster-based support vector machine in text-independent speaker identification,” in *Proceedings of International Joint Conference on Neural Networks IJCNN 2004*, Budapest, Hungary, 2004.

[51] L. Zhu, A. Rao, and A. Zhang, “Theory of keyblock-based image rerieval,” *ACM Trans. on Information Systems*, vol. 20, no. 2, pp. 224–257, April 2002.