國 立 交 通 大 學

電子工程學系 電子研究所碩士班

碩 士 論 文

使用視覺上可移除的數位浮水印
實現可調式多媒體階層式保護

Layered Protection of Scalable Media
using Perceptually Removable Watermarks

研 究 生：朱育成

指導教授：杭學鳴 博士

中 華 民 國 九 十 六 年 六 月

使用視覺上可移除的數位浮水印

實現可調式多媒體階層式保護

Layered Protection of Scalable Media using Perceptually Removable Watermarks

研究生：朱育成 Student: Yu-Cheng Chu

指導教授：杭學鳴 Advisor: Dr. Hsueh-Ming Hang

國 立 交 通 大 學

電 子 工 程 學 系 電 子 研 究 所 碩 士 班

碩 士 論 文

A Thesis

Submitted to Department of Electronics Engineering & Institute of Electronics

College of Electrical and Computer Engineering

National Chiao Tung University

in Partial Fulfillment of the Requirements

for the Degree of Master

in

Electronics Engineering

June 2007

HsinChu, Taiwan, Republic of China

中華民國九十六年六月

使用視覺上可移除的數位浮水印實現可調式多媒體階層式保護

研究生：朱育成　　　　　　　　　　　　　　　指導教授：杭學鳴

國立交通大學 電子工程學系 電子研究所碩士班

摘要

隨著網際網路的日漸普及，數位資料變得非常容易取得，因此我們需要一個保護架構來保護智財權。再者，由於行動通訊的發達，加上行動通訊本身頻寬的多樣化，造成了可調式多媒體日趨重要，如此一來，為了保護可調式多媒體，一個可調式多媒體階層式保護的架構被提了出來。這個架構利用加密系統來保護各階層的資料，而解密的金鑰則以浮水印鑲嵌在上一個階層的資料內。

在本論文中，我們主要的目的是驗證可調式多媒體階層式保護的可行性，其驗證的範圍包括了解析度可調以及影像品質可調。為了驗證這兩種可調式媒體，JPEG2000 被選擇為測試平台，這是因為 JPEG2000 本身提供了解析度可調與影像品質可調的檔案格式。為了攜帶解密金鑰，一個視覺上可移除的數位浮水印被設計出來，這個數位浮水印利用鑲嵌高強度的浮水印來達到高資料容量的目的，但是隨之而來的是一個品質較差的鑲嵌後影像，不過一個視覺上不受影響的影像可以在最後被取得。有了視覺上可移除的數位浮水印的幫助，用來保護解析度可調或是影像品質可調多媒體的階層式保護架構可以被完整地實現。

Layered Protection of Scalable Media using Perceptually Removable Watermarks

Student : Yu-Cheng Chu Advisor : Dr. Hsueh-Ming Hong

Department of Electronics Engineering & Institute of Electronics National Chiao Tung University

Abstract

With the widespread use of Internet, digital multimedia data are easy to access. Therefore, a protection scheme is needed to secure the intellectual property. On the other hand, scalable media coding becomes more and more important due to the varying bandwidth in transmission such as mobile communication. Hence, a layered protection scheme of scalable media was proposed. It is a scheme which protects data of each layer by cryptosystem and the decoding key of the current layer is carried by the watermark embedded in the previous layer.

In this thesis, our goal is to design and implement the layered protection scheme for the SNR and the spatial scalable media. Thus, the JPEG2000 still image compression standard, which provides content formats for SNR and spatial scalabilities, is used as the platform in our implementation. To fulfill all the requirements in our system, the perceptually removable watermarking (PRW) is chosen for carrying the decoding keys. The specifically designed PRW embeds the watermark with high intensity to increase the data capacity, and it can be removed at the other end with virtually no loss on image quality. With the aid of the PRW, the layered protection scheme for SNR and spatial scalability can be successfully implemented.

誌謝

感謝杭學鳴老師這兩年來細心的指導，讓我了解到做研究該有的態度以及方法，也感謝實驗室的學長姊和同學，尤其是峰誠學長總是給我一些良好的建議，使我在遭遇瓶頸時能夠順利解決，也謝謝建志幫助我了解 JPEG2000 的架構，讓我能夠將研究精力專注在設計而非摸索測試平台；也謝謝家人以及女友的體諒，讓我能夠專注研究而不會被一些瑣事所打擾。

在這兩年的研究生活中，因為大家的幫忙，我學習到了如何解決問題，也順利完成了學業。在此把本論文獻給所有幫助過我的人。

Contents

List of Tables

List of Figures

Chapter 1 Introduction

With the widespread use of Internet, all digital data are easy to access. Thus, a protection scheme is necessary for the multimedia distribution. Moreover, the scalable media coding becomes more and more important due to the varying bandwidth in transmission such as mobile communication. Thus, we like to propose a scheme for protecting the scalable media.

Scalable coding implies that a multimedia is divided into a base layer and several enhancement layers, and each enhancement layer can be independently used to improve the reconstructed image quality. There are three types of scalability, that is, spatial (resolution) scalability, temporal scalability, and signal-to-noise (SNR) scalability. The spatial scalability means that each layer has a different spatial resolution, and we retrieve the reconstructed image with higher resolution when more enhancement layers are decoded. The temporal scalability is a scheme that enhancement layers increase the overall frame rate of a video. And the SNR scalability allows different layers to have different image quality.

In [22], the layered protection scheme for scalable media was proposed. It is a scheme which protects data of each layer by cryptosystem and the decoding key of the current layer is transmitted by the watermark embedded in the previous layer. Since the verification of the layered protection scheme in [22] is only done for the temporal scalability, there is still a question that whether the proposed scheme will work or not for the SNR scalability and the spatial scalability. Thus, our contribution in this thesis is to design the layered protection scheme for the SNR and the spatial scalabilities.

In our system, the JPEG2000 still image compression standard, which provides content formats for the SNR and the spatial scalabilities, is adopted. And the watermarking technique used for carrying the decoding key is a specifically designed one. It is the perceptually removable watermarking (PRW). With the aid of the PRW, the layered protection scheme for SNR and spatial scalability can be successfully implemented.

This thesis is organized as follows. Chapter 2 describes the JPEG2000 standard, the watermarking techniques, and the concept of layered protection scheme for scalable media. In Chapter 3, the characteristics of different watermarking techniques are analyzed and the correlation-based watermarking is chosen to be the method of data transmission in the layered protection scheme. We try to improve the efficiency of the chosen watermarking technique in Chapter 4 and the perceptually removable watermarking is thus proposed. Chapter 5 shows the simulation results. And Chapter 6 concludes this thesis.

Chapter 2
JPEG2000, Watermarking, and Layered Protection

There are three parts in this chapter, they are JPEG2000 standard, digital watermarking, and the architecture of layer protection of scalable media. At first, we will briefly describe the JPEG2000 standard by describing the features of JPEG2000 and explaining the coding procedure of JPEG2000. After describing the JPEG2000 standard, an overview of digital watermarking will then be introduced. And the architecture of layered protection of scalable media that we will design and implement is mentioned in the end.

2.1 The JPEG2000 Standard

Because the increasing popular use of multi-media technologies is increased, the old JPEG standard used to compress still images cannot fulfill the advanced requirements for image coding today. Thus, a new image compression technique with better compressed efficiency and other useful features is needed. A new standard called JPEG2000 was developed during 1997 to 2000 and achieved the aforementioned demands.

2.1.1 Introduction to JPEG2000

JPEG2000 was designed to address the requirements for a diversity of applications such as internet, remote sensing, medical imagery, and so on. Therefore, JPEG2000 has a number of features and the most important features [2] of JPEG2000 are listed below.

- **Superior low bit-rate performance**

The standard should provide the superior performance at low bit-rates (e.g. below 0.25 bpp) as comparing to the old standard such as JPEG. Finally, JPEG2000 has a compression advantage over JPEG by roughly 20% and a subjective quality benefit. This feature is important in some bandwidth sensitive applications such as network image transmission and remote sensing.

- **Lossless and lossy compression**

JPEG2000 provides both lossless and lossy compression using the same architecture. It is desired to provide lossless compression naturally in the course of progressive decoding, and then we can progressively construct the image from the lossy mode to the lossless mode.

- **Progressive transmission by pixel accuracy and resolution**

Progressive transmission that allows images to be constructed with increasingly pixel accuracy or spatial resolution is important for many applications. This feature makes it possible for different devices to reconstruct image with different resolutions and pixel accuracy. Applications like World Wide Web, image archival and printers are some instances.

- **Region-of-interest coding**

Some parts of an image that are more important than the others. This feature allows users to preserve the fidelity of the interested region by using more bits to code the data in the region-of-interest (ROI) and transmitting the coded data earlier.

- **Random codestream access and processing**

This feature allows user to randomly define the ROI's in the image and code them with less distortion than the rest of the image. Moreover, rotation, filtering, scaling and feature extraction are supported.

- **Robustness to bit-errors**

Proper design of the codestream can aid subsequent error correction systems to alleviate catastrophic decoding errors and make sure that some specific portions of the codestream are correct. This feature can overcome the obstacle of transmission over a noisy channel like wireless channel.

- **Open architecture**

It is desirable to have an open architecture to optimize the system for different image types and applications, so the decoder is only required to implement the core tool set and a parser that understands the codestream. Also, unknown tools could be added to the decoder if necessary.

- **Content-based description**

Since image archival, indexing and searching is an important issue in image processing, content-based description of images might be available as part of compression system.

- **Side channel spatial information (transparency)**

Side channel spatial information such as alpha planes and transparency planes are useful in many applications. One example is the transparency plane use in the World Wide Web applications.

- **Protective image security**

We can use watermarking, labeling, stamping and encryption to protect a digital image. Labeling is already implemented in still picture interchange file format (SPIFF) [1] and JPEG2000 must be easy to achieve the target.

- **Continuous-tone and bi-level compression**

A coding standard which has a unified compression architecture to compress both continuous-tone and bi-level image is desirable. This system is able to compress and decompress each color component with depth of 1 to 16 bits.

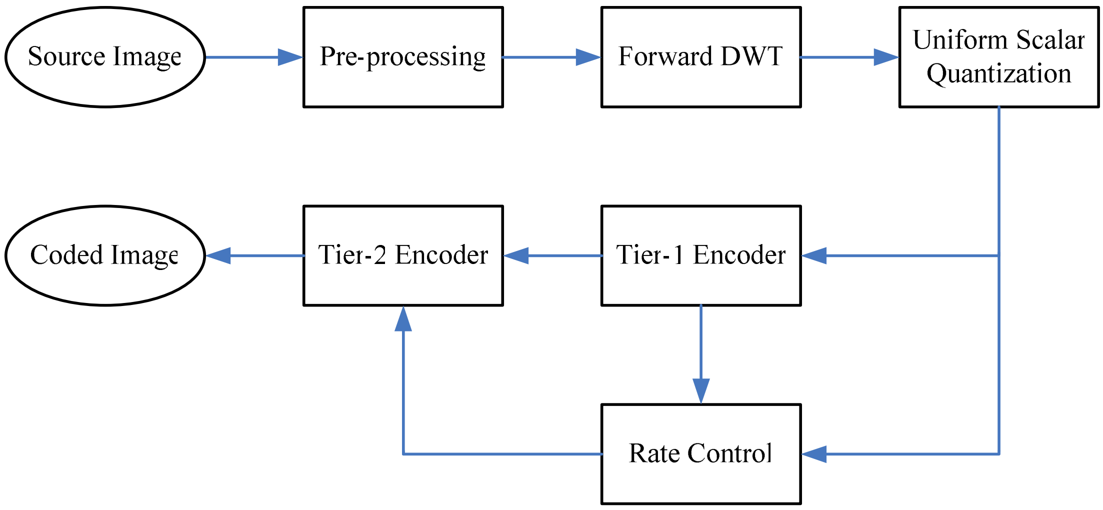


Figure 2-1 Figure 2-1 General block diagram of JPEG2000 encoder [1]

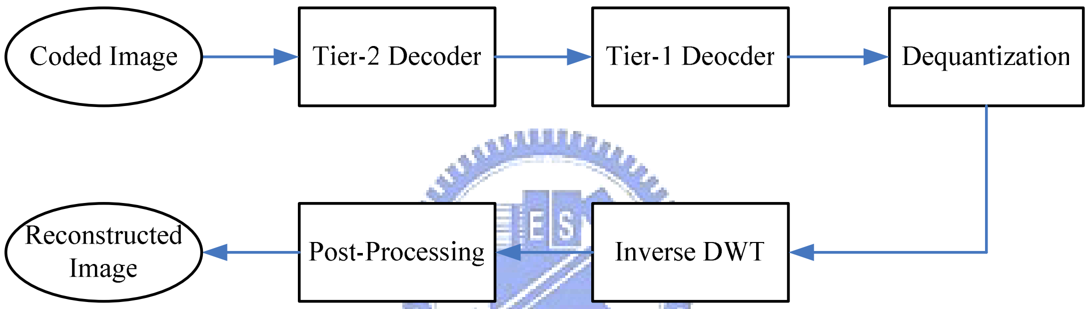


Figure 2-2 General block diagram of JPEG2000 decoder [1]

Figures 2-1 and 2-2 are the block diagrams of the encoder and the decoder in the JPEG2000 standard. Before introducing the functionality of each block in the block diagrams, we should note that the decoder is simply the inverse of the encoder. Thus, although most parts of the JPEG2000 standard are written from the point of view of the decoder, for easy understanding, we will describe the JPEG2000 encoding tools in the following sections.

2.1.2 Pre-Processing

There are three sub-steps in the preprocessing step, which are "Image Tiling", "DC Level Shifting" and "Multi-component Transform" as shown in Figure 2-3. We will introduce each of them as follows.

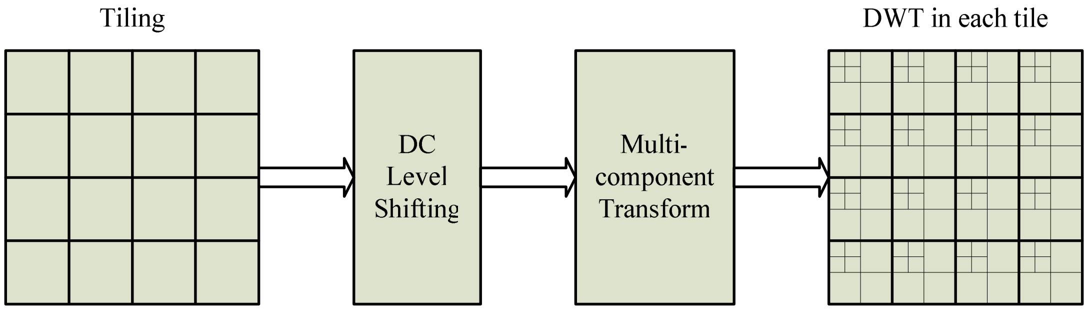


Figure 2-3 Tiling, DC Level Shifting, Multi-component Transform (optional)

2.1.2.1 Image Tiling

Image tiling divides the original image into several rectangular non-overlapping blocks (tiles), and each tile is compressed independently. Since each tile could be processed independently, this step could significantly reduce the use of memory, and we can also decode some specific parts of the image instead of the whole image by this way. Although tiling could reduce the use of memory, the cost is the reduced quality of the compressed image.

2.1.2.2 DC Level Shifting

After tiling image, all pixels in each tile are dc-level shifted by subtracting the same quantity 2^{p-1}, where p is the precision of the corresponding component. It is important to note that the dc-level shifting is only performed on samples that are unsigned.

2.1.2.3 Multi-component transform

Multi-component transform can improve the coding efficiency by efficiently reducing the redundancy between each component of the original image file [3] such as RGB components of the bmp files. JPEG2000 supports two types of transformation, that is, reversible component transformation (RCT) and irreversible component transformation (ICT). RCT can be used for lossy or lossless coding while ICT should only be used for lossy coding. The forward and the inverse ICT transformations are achieved by means of equations (2.1.2-1) and (2.1.2-2) respectively. The other one, RCT, refers to (2.1.2-3) and (2.1.2-4).

JPEG2000 supports multiple-component images and different components need not have the same bit depths. For this reason, the bit depth of each output image component must be identical to the bit depth of the corresponding input image component. This is the only requirement for reversible systems.

$$\begin{bmatrix} Y \\ Cb \\ Cr \end{bmatrix} = \begin{bmatrix} 0.299 & 0.587 & 0.114 \\ -0.16875 & -0.33126 & 0.5 \\ 0.5 & -0.41869 & -0.08131 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \qquad (2.1.2\text{-}1)$$

$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} = \begin{bmatrix} 1.0 & 0 & 1.402 \\ 1.0 & -0.34413 & -0.71414 \\ 1.0 & 1.772 & 0 \end{bmatrix} \begin{bmatrix} Y \\ Cb \\ Cr \end{bmatrix} \qquad (2.1.2\text{-}2)$$

$$\begin{bmatrix} Y \\ U \\ V \end{bmatrix} = \begin{bmatrix} \left\lfloor \frac{R+2G+B}{4} \right\rfloor \\ R-G \\ B-G \end{bmatrix} \qquad (2.1.2\text{-}3)$$

$$\begin{bmatrix} G \\ R \\ B \end{bmatrix} = \begin{bmatrix} Y - \left\lfloor \frac{U+V}{4} \right\rfloor \\ U+G \\ V+G \end{bmatrix} \qquad (2.1.2\text{-}4)$$

2.1.3 Discrete Wavelet Transform

Wavelet transform is a transform that produces the transformed data having both spatial and frequency information, and JPEG2000 used this property to achieve the goal of scalable coding with different resolution and quality.

In JPEG2000, the wavelet transform is used to decompose the tile components into different decomposition levels. Each of them has a number of subbands with vertical and horizontal characteristics of the original image. Due to the less correlated properties of these subband signals, the transformed data can be coded more efficiently than the original data.

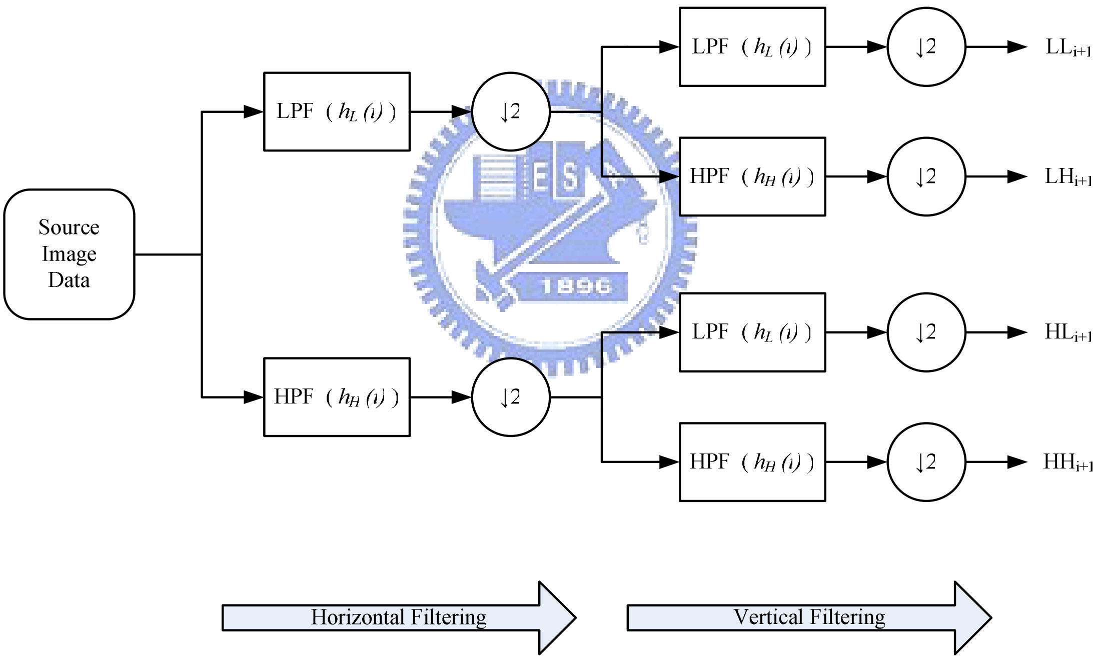


Figure 2-4 2D forward discrete wavelet transform

Usually, the two-dimensional (2D) discrete wavelet transform is accomplished by cascading two one-dimensional (1D) discrete wavelet transform. As shown in Figure 2-4, the two-dimensional discrete wavelet transform is composed of two one-dimensional discrete wavelet transform along the horizontal and the vertical directions, respectively. After horizontal transform, two subbands are formed, and the vertical transform is then applied to each band to produce the four bands in two-dimensional wavelet transform. Figure 2-5 shows the flow of decomposing one subband into four higher-level subbands. It is important to note that for one-dimensional discrete wavelet transform, the low-pass samples represent a down-sampled low-resolution version of the original signal and the high-pass samples represent a down-sampled residual version of the original signal. This property is useful in scalable coding since we can get the low-resolution image by only decoding the LL-band of the transformed data.

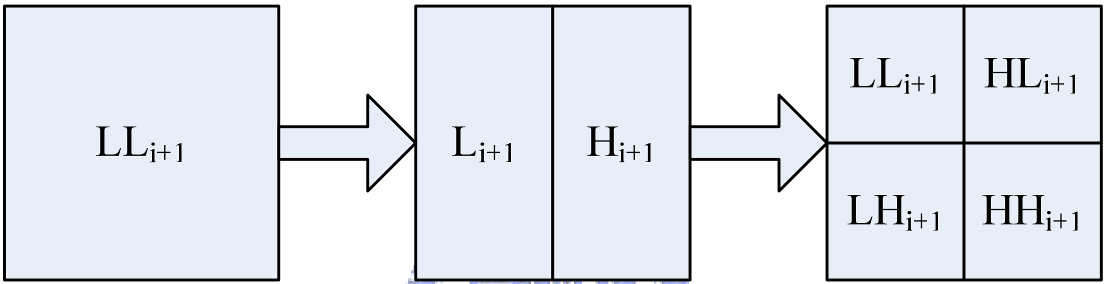


Figure 2-5 2D DWT decomposition

JPEG2000 provides two types of wavelet transform. One of them is the reversible DWT with the Le Gall 5/3 filter; the other one is the irreversible DWT with the Daubechies 9/7 filter. Their coefficients of the analysis and synthesis filter are listed in Table 2-1 and Table 2-2.

Since the RCT may be used for lossy and/or lossless coding, it may only be used together with the 5/3 reversible wavelet transform. And since the ICT may only be used for lossy coding, the 9/7 irreversible wavelet transform may be the only choice for ICT to be used with.

	Analysis Filter Coefficients		Synthesis Filter Coefficients	
i	Low-Pass Filter h_L *(i)*	High-Pass Filter h_H *(i)*	Low-Pass Filter g_L *(i)*	High-Pass Filter g_H *(i)*
0	6/8	1	1	6/8
±1	2/8	-1/2	1/2	-2/8
±2	-1/8			-1/8

Table 2-1 Le Gall 5/3 analysis and synthesis filter coefficients

	Analysis Filter Coefficients		Synthesis Filter Coefficients	
i	Low-Pass Filter h_L *(i)*	High-Pass Filter h_H *(i)*	Low-Pass Filter g_L *(i)*	High-Pass Filter g_H *(i)*
0	0.6029490182363579	1.115087052456994	1.115087052456994	0.6029490182363579
±1	0.2668641184428723	-0.5912717631142470	-0.5912717631142470	0.2668641184428723
±2	-0.07822326652898785	-0.05754352622849957	-0.05754352622849957	-0.07822326652898785
±3	-0.01686411844287495	0.09127176311424948	0.09127176311424948	-0.01686411844287495
±4	0.02674875741080976			0.02674875741080976

Table 2-2 Daubechies 9/7 analysis and synthesis filter coefficients

2.1.4 Quantization

After transformation, all coefficients are quantized. This process is lossy unless the quantization step is equal to one with integer coefficients. Several quantization options are provided by the JPEG2000 standard, but only the uniform scalar quantization, which is the default method in JPEG2000 standard Part I, will be introduced here.

In the integer mode, the quantizer step sizes are always fixed to be one, effectively bypassing quantization and making the transform coefficients unchanged. In this case, lossy coding is still possible, but the rate control is achieved by the other mechanism. In the real mode, the quantizer step sizes are chosen in conjunction with the rate control. Each of the transformed coefficients $a_b(u,v)$ of the subband b is quantized to the value $q_b(u,v)$ according to the formula (2.1.4-1). The step size $\triangle_b$ is defined in (2.1.4-2). It is relative to the dynamic range R_b of the subband b. And the exponent/mantissa pairs (ε_b, μ_b) are signaled in the bit stream syntax.

$$q_b(u,v) = \mathrm{sign}(a_b(u,v)) \left\lfloor \frac{|a_b(u,v)|}{\Delta_b} \right\rfloor \quad (2.1.4\text{-}1)$$

$$\Delta_b(u,v) = 2^{R_b - \varepsilon_b}\left(1 + \frac{\mu_b}{2^{11}}\right) \quad (2.1.4\text{-}2)$$

2.1.5 Embedded Block Coding with Optimized Truncation

Embedded block coding with optimized truncation (EBCOT) [4] is adopted for the entropy coding part of JPEG2000. There are two major coding steps in EBCOT, that is, tier-1 and tier-2 coding, as shown in Figure 2-6. The tier-1 part is the embedded block coding (EBC) which includes the context formation (CF) and the arithmetic encoder (AE). In the tier-1 coding, the encoder divides each subband into code-blocks and all code-blocks are encoded independently to form a block-based embedded bit-stream. The coding method is the bit-plane coding described later in the next section. For each code-block, the embedded bit-stream is composed of numerous coding passes and the outputs of tier-1 are the collection of coding passes of various code-blocks. After the embedded bit-stream is produced, the tier-2 part truncates the bit-stream for best rate-distortion optimization. That is, for fixed quality, the rate is minimized and for a fixed rate, the overall distortion is minimized. We will introduce two tiers in details in the following sections.

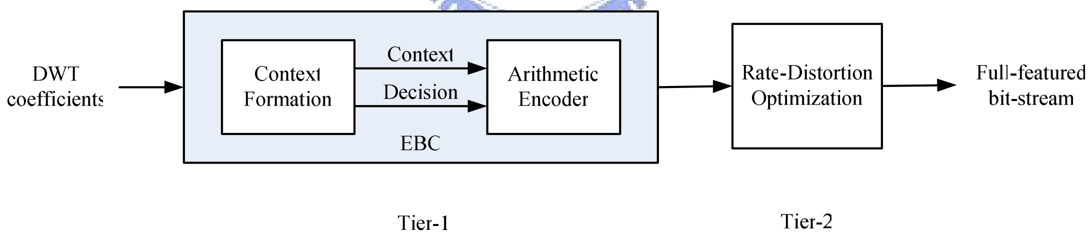


Figure 2-6 Two tiers of EBCOT algorithm

2.1.5.1 Tier-1 Coding

The tier-1 coding is also known as the embedded block coding (EBC). It is composed of the context formation (CF) and the arithmetic encoder (AE), and the code-block is its basic coding unit. The EBC is coding in bit-level by bit-plane coding, that is, the code-block is coded from the most significant bit (MSB) bit-plane to the least significant bit (LSB) bit-plane. Each bit-plane takes three passes and is scanned in a stripe-based method shown in Figure 2-7.

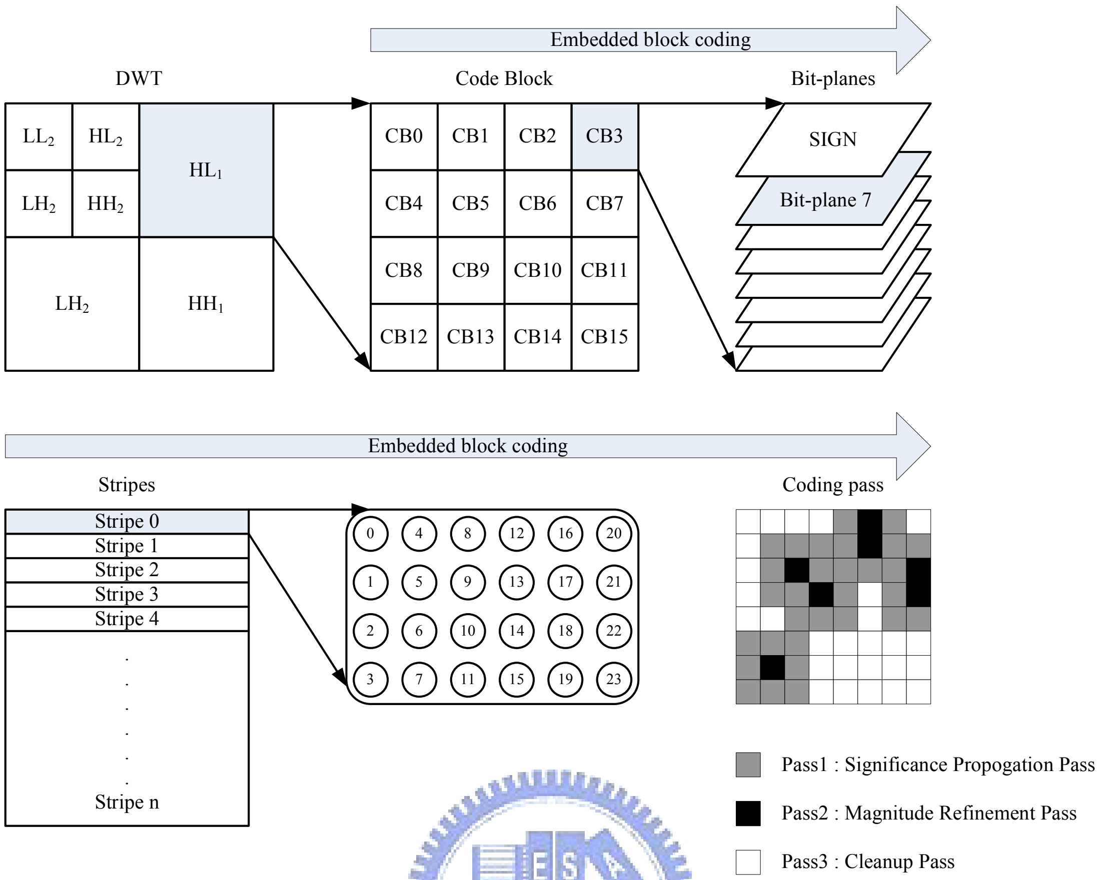


Figure 2-7 Diagram of code-blocks, bit-planes, stripes and coding pass

- Context Formation (CF)

As shown in Figure 2-6, the EBC is essentially a context-adaptive arithmetic encoder. The context formation (CF) generates context-decision pairs for the arithmetic encoder (AE) to adapt the probability of decision. In context modeling, all code-blocks are coded a bit-plane at a time starting from the MSB bit-plane with a nonzero element to the LSB bit-plane. There are three coding passes for a bit-plane, that is, Pass1 (Significance Propagation Pass), Pass2 (Magnitude Refinement Pass), and Pass3 (Cleanup Pass). Each DWT coefficient bit is coded in only one of the three coding passes, and the coding condition is shown in Table 2-3.

Coding Pass	***Coding condition***
Pass1 (Significance Propagation Pass)	Insignificant sample with at least one significant neighbor
Pass2 (Magnitude Refinement Pass)	Significant sample
Pass3 (Cleanup Pass)	All remaining coeeficients

Table 2-3 Coding Pass Classification

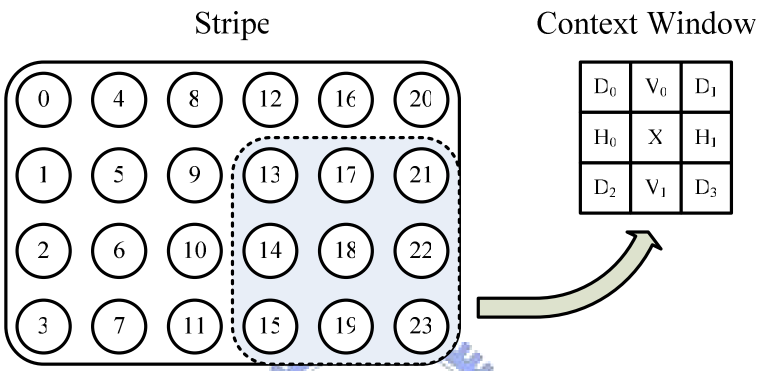


Figure 2-8 Context window and Neighbor states

As the context window shown in Figure 2-8, the 4-connected or the 8-connected neighbors of a sample are chosen to examine the state information used for the context-based arithmetic coding. Various context information (context label) for different passes are shown in Table 2-4, Table 2-5, and Table 2-6, and the “X” in the tables means “don’t care”.

The first coding pass (Pass1) for each bit-plane is the significance propagation pass. In significance propagation pass, a bit is coded if and only if its location is insignificant and at least one of its 8-connected neighbors is significant. Nine context labels in Table 2-4 are created based on the characteristics of the 8-connected neighbors, that is, how many and which ones are significant. The mapping to the contexts also depends on which subband the code-block is in. The significance propagation pass only includes bits of coefficients that were insignificant with non-zero context. All other coefficients are skipped. If the value of this bit is 1 then the significance state is set to 1 and then the sign bit coding must be performed right away. Otherwise, the significance state remains 0. It is important to note that the coding pass always use the most current significance state to encode the bit.

LL and LH sub-bands (vertical high-pass)			HL sub-band (horizontal high-pass)			HH sub-band (diagonally high-pass)		Context Label
ΣH	ΣV	ΣD	ΣH	ΣV	ΣD	Σ(H+V)	ΣD	
2	X	X	X	2	X	X	≧3	8
1	≧1	X	≧1	1	X	≧1	2	7
1	0	≧1	0	1	≧1	0	2	6
1	0	0	0	1	0	≧2	1	5
0	2	X	2	0	X	1	1	4
0	1	X	1	0	X	0	1	3
0	0	≧2	0	0	≧2	≧2	0	2
0	0	1	0	0	1	1	0	1
0	0	0	0	0	0	0	0	0

Table 2-4 Contexts for the significance propagation pass and cleanup pass

As shown in Table 2-5, the sign bit coding uses another context of the neighborhood to determine the context label. Only four neighbors are considered, and each neighbor may have one of three states: significant positive, significant negative, or insignificant. If the two vertical neighbors are both significant with the same sign, or if only one neighbor is significant, then the vertical contribution is 1 if the sign is positive or -1 if the sign is negative. If both vertical neighbors are insignificant or both are significant with different signs, then the vertical contribution is 0. The horizontal contribution is determined in the same manner.

The second coding pass (Pass2) for each bit-plane is the magnitude refinement pass. This pass includes the bits that are already significant (except those that have just become significant in the proceeding significance propagation pass). The context used in this pass is determined by summation of the significance state of the horizontal, vertical, and diagonal neighbors as shown in Table 2-6.

Horizontal contribution	Vertical contribution	Context Label	XOR bit
1	1	13	0
1	0	12	0
1	-1	11	0
0	1	10	0
0	0	9	0
0	-1	10	1
-1	1	11	1
-1	0	12	1
-1	-1	13	1

Table 2-5 Contributions of the vertical and the horizontal neighbors to the sign context

$\Sigma H+\Sigma V+\Sigma D$	First refinement for this sample	Context Label
X	False	16
$\geqq 1$	True	15
0	True	14

Table 2-6 Contexts for the magnitude refinement pass

All the remaining coefficients in the bit-plane are insignificant with the context value of zero during the significance propagation pass. They are all included in the cleanup pass (Pass3). The cleanup pass not only uses the neighbor context, like that of the significance propagation pass from Table 2-4, but also a run-length context. If the four continuous samples in a column and the context labels of the four samples are all zeros, the run-length coding is performed.

- Arithmetic Encoder (AE)

The decision which is produced by the CF is coded during the arithmetic encoding. The AE is an adaptive, binary MQ-coder [5]. The basis of binary arithmetic coding process is a recursive probability interval subdivision of Elias coding. Since the arithmetic encoder is binary, with each binary decision, the current probability interval is divided into two sub-intervals, and the width of the sub-interval is proportional to the corresponding sample probability. The code-stream is modified (if necessary) so that the base (lower bound) of the sub-interval is assigned to the symbol, and if the interval is too small, the sub-interval is renormalized to a larger interval for AE to divide. All the processes described above are shown in Figure 2-9. Besides, a lazy coding is used to reduce the number of symbols that are arithmetically coded. According to this mode, after the fourth bit-plane is coded, the first and second pass are included as raw, while only third coding pass of each bit-plane applies the arithmetic coding.

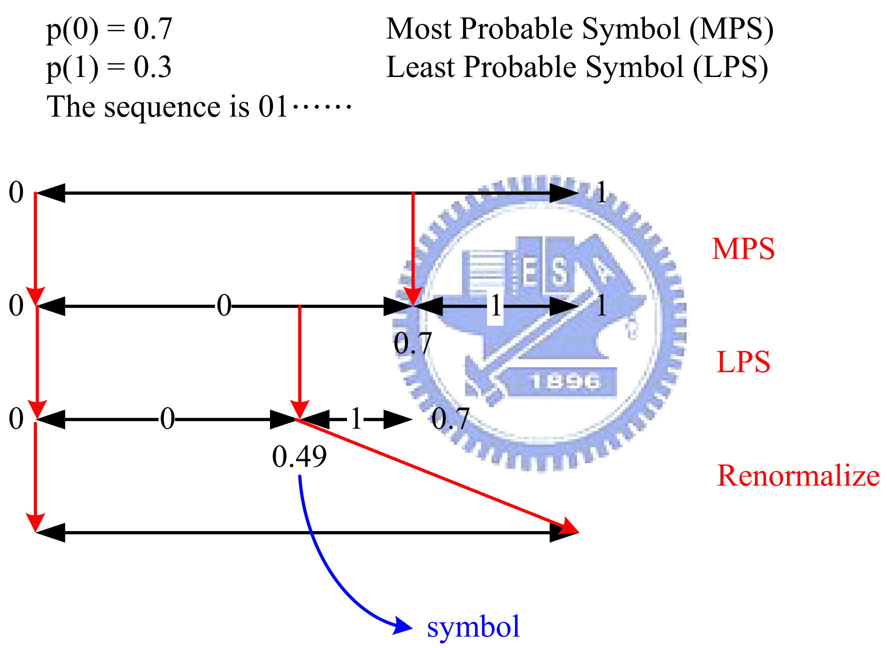


Figure 2-9 Basic operation of the arithmetic encoding

2.1.5.2 Tier-2 Coding

There are two main purposes in tier-2 coding. One is to truncate the bit-stream which is coded in tier-1 coding to achieve the objective quality or rate; the other is to permute the packet which is composed of the three bit-plane coding passes in different progression orders to achieve different kinds of scalability such as resolution scalability, quality scalability (PSNR scalability), and so on.

The input of the tier-2 encoding process is the set of bit-plane coding passes generated during the tier-1 encoding. Each coding pass is a candidate for truncation point of a code-block, and the coding pass information is packaged into packets based on different requirements. For meeting a target bit-rate, the packaging process imposes a particular organization of coding pass data in the output code-stream. The rate-control scheme assures that the output code-stream with desired bit-rate has the best reserved quality compared to all other code-stream at the same rate. The rate distortion optimization (RDO) algorithm will be introduced as follow.

In the encoder, rate control can be achieved through two distinct mechanisms, that is, the choice of quantization step size and the selection of the subset of coding passes to include in the code-stream. When lossless coding is employed, only the second mechanism may be used. The quantization step size must be fixed to one. When lossy coding is applied, both mechanisms may be employed. Since the tier-1 coding needs a lot of computation and changing quantization step sizes leads to redo the tier-1 coding, the first mechanism seems not practical in the tier-2 encoder. The remaining method to control bit-rate is to discard some coding passes to achieve the target rate. Each coding pass gives a load on bit-rate but provides an improvement in quality. Using this information, the encoder could create a bit-stream composed of specific coding passes to minimize the distortion at a fixed rate, and this is the task of rate control.

The rate distortion optimization algorithm is to make a trade-off between bit-rate and distortion. We should minimize the distortion while the rate is fixed or minimize the rate when the distortion is fixed. The solution for this problem is the Lagrange optimization problem [1] as (2.1.5.2-1).

$$\min(R+\lambda D)=\min(\sum_i (R_i^{n_i}+\lambda D_i^{n_i})) \qquad (2.1.5.2\text{-}1)$$

D is the total distortion, R is the total bit-rate, and n_i is the truncation point of the code-block B_i. The Lagrange multiplier (λ) is used to minimize J = R+λD and since that to minimize J is equal to minimize $(R_i^{n_i} + \lambda D_i^{n_i})$ separately. A simple algorithm to minimize J is as follows:

> Set $n_i = 0$
>
> For k = 1,2,3,…
>
> Set $\Delta R_i^k = R_i^k - R_i^{n_i}$ and $\Delta D_i^k = D_i^k - D_i^{n_i}$
>
> If $\Delta D_i^k / \Delta R_i^k > \lambda^{-1}$ then set $n_i = k$

The core concept of the algorithm above is that $R_i^{n_i} + \lambda D_i^{n_i} > R_i^k + \lambda D_i^k$ means k is better than n_i to be the truncation point. Thus the formula "If $\Delta D_i^k / \Delta R_i^k > \lambda^{-1}$ then set $n_i = k$" is formed.

After the rate control process, the packets of data are permuted in different progression orders to achieve the function of scalable coding. As shown in Table 2-7, there are four types of progression order.

Progression order
Layer-resolution-component-position progressive (LRCP)
Resolution-layer-component-position progressive (RLCP)
Resolution-position-component-layer progressive (RPCL)
Position-component-resolution-layer progressive (PCRL)

Table 2-7 The progression order of packets

2.2 The Digital Watermarking

Because the transmission of digital data over the internet is very easy today, there is a need of a technique to protect the intellectual property of digital data. Traditional encryption system satisfies only a portion of this requirement because the clear context of the protected data can be accessed at the paying receiver end. Hence, digital watermarking technique was proposed, which hides the information into the host data without perceptual distortion. There are many kinds of digital data, but we will focus only on the watermark on image data in the following sections.

2.2.1 Introduction to Digital Watermarking

A watermark is required to be robust, invisible, and it conveys as much information as possible. For real-time system, the watermarking system is also asked to have low complexity. Since all the requiring features are conflicting, the design of watermark is just a trade-off issue.

Using the characteristics of the human visual system (HVS) to hide more information with the same perceptual quality is the most popular method to improve the efficiency of watermarking. The domain where watermark is embedded in also affects the robustness. Generally speaking, frequency domain is good for data embedding while the embedded data in spatial domain is easy to be destroyed.

In the following sections, we will introduce the watermarking techniques in detail, such as watermarking process, properties of watermark, categories of watermark, and some examples of watermarking techniques. And the most important model, the human visual system (HVS) model, is described at the end.

2.2.2 The Process of Watermarking

A watermarking system is formed by four basic stages [7]. They are embedding stage, distribution stage, extraction stage, and detection stage.

- **Embedding stage:**

In this stage, the watermark W with the weighting g is added to the host image I as shown in Figure 2-10. The host image data may be in the spatial domain or the frequency domain.

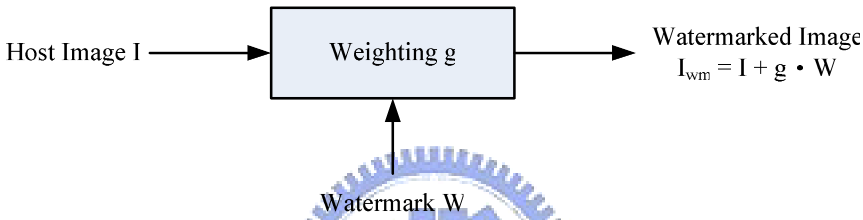


Figure 2-10 Embedding Process

- **Distribution stage:**

In this stage, the watermarked image is distributed. It may be published on a web or sold to a customer. During the transmission and distribution of the data, the watermarked image is affected by some harmful factors, for example, common image processing tools such as lossy compression of the data, transmission error due to noisy-channel, and geometric transformations such as clipping or rotation. All the factors above introduce errors to watermarked image. And then, all manipulation of the watermarked image should be seen as an attack on the information embedded by watermark.

- **Extraction stage:**

There are three types of methods in extracting the watermark from the watermarked image, that is, blind, semi-blind and non-blind extraction. As shown in

Figure 2-11, the watermark is extracted from the watermarked image by the watermark extraction process. The blind extraction is the extraction that does not need side information. For semi-blind extraction method, a little side information such as the image features is needed. And if the original host image is needed, this is the non-blind extraction.

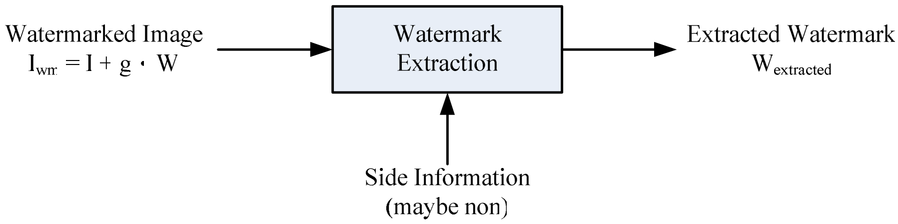


Figure 2-11 Extraction process

- **Detection stage:**

As shown in Figure 2-12, the extracted watermark in the extraction stage is compared to the original watermark. If the extracted watermark is the same as the original watermark or they are similar to a certain degree, we can say that the image is exactly the one we want to identify.

Two types of errors may happen in this process, that is, false positive and false negative. False positive means that a watermark is detected although there is none, and false negative rejects the existence of the watermark even though there is one. These two errors may be caused by the transmission error or the design of watermarks.

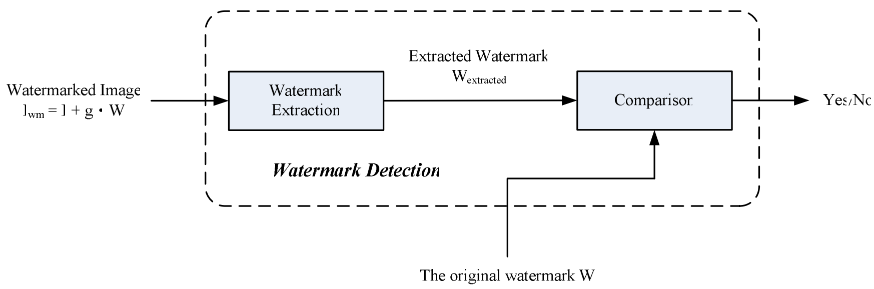


Figure 2-12 Detection process

2.2.3 Properties of Watermark

There are several desirable properties of a digital watermark, and they may be required depending on applications. For example, the fragile watermark is used for data authentication, and the robust watermark is used for copyright protection. The properties of watermark [7] are listed below.

- **Perceptual transparency**

A watermarked image should not be perceptually different from the original one. That is, the watermark should not degrade the perceptual quality of the image to be embedded in.

- **Robustness**

The watermark should be robust means that the watermark is very hard to remove without degrading the quality of the watermarked image. Thus, if a watermark is destroyed, the noisy image with no meaning is then formed unavoidably. Here are some processes the watermark should resist to. One is the common image processing techniques such as lossy compression, which throws away the unnecessary information for perceptual quality of an image to achieve the goal of data compression. Another one is the geometric transformation such as rotation, scaling, or clipping. The third one is the noisy-channel that the watermarked image should pass through. In this channel, noise is added to the watermarked image and results in damaged data. We wish the watermark is resistant to such kind of attack either.

- **Security**

Security means how easy a watermark could be intentionally removed by deletion, modification or burying of the watermark in another illicit one.

- **Data capacity**

Data capacity refers to the capacity for a host image to carry the hidden information. Each image has its own capacity limit for a watermark method. Therefore, the same watermarking architecture should not embed the same amount of information into different pictures.

2.2.4 Categories of Watermark

Base on the properties of watermark, the watermarking techniques can be classified into several categories [8], and some of them are listed below.

- **Perceptible watermarking**

Perceptible watermark refers to the watermark which can be discovered by human eyes. There are two important criteria for perceptible watermark. One is that the watermark should be robust for an authorized person to remove it; the other one is that the watermark should be falsification-resistant. Since it is relatively easy to embed a logo or pattern into the image, we have to ensure that the watermark is embedded by the claimed user.

- **Imperceptible watermark**

The imperceptible watermark is the watermark now in common use. It is invisible but can be extracted by computer through specific algorithm. This makes it possible to build a secret communication channel between one or more recipients.

- **Robust watermark**

Robust watermark refers to the watermark which is resistant to common image processing such as compression or geometric transformation. This means that the watermark will still survive after the destructive procedures described above.

Two major classes of the robust watermark are public and private watermarks [6] [8]. The public watermark needs no side information in the watermark detection stage but the private watermark requires side information. Public watermarks can be accessed by virtually anyone and it provides a side channel for information carrying. Its major drawback is the limited bandwidth for the data. On the other hand, side information required by the private watermark makes a big load on the channel bandwidth and how to securely transmit the side information is a big problem. This is why the private watermark is not popular with most users.

- **Fragile watermark**

As its name implies, the fragile watermark is easy to be destroyed by all kinds of processing procedures. The application of this kind of watermark is the authentication of data, in this application, if the watermark is destroyed, we can say that someone else has modified the data.

- **Semi-fragile watermark**

There exists one watermark scheme that has the characteristics between robust and fragile watermark, it is the semi-fragile watermark. The only purpose of semi-fragile watermark is to differentiate between lossy transformations that are "information preserving" and lossy transformations that are "information altering" [9]. For example, it is important for an authentication to distinguish the changes of the original data is made by the compression process or the other data manipulation. Data compression, which preserves the integrity of contents, is allowed but manipulations which alter the data integrity is forbidden. The semi-fragile watermark achieves this goal because it survives the compression operator but it is broken after the other manipulations.

- **Removable watermark**

The watermarking technique which makes it possible to obtain the original image from the watermarked image is called the removable watermarking. But this kind of watermark violates the desired properties of the robust watermark.

- **Un-removable watermark**

This is the goal that most watermarking algorithms like to achieve. In this scheme, the watermark is tightly combined with the host image and to separate them is almost impossible. Hence, extra information such as the ownership of the products is protected securely.

2.2.5 Watermarking Techniques

We describe several types of watermarking techniques. We will not describe their details but only introduce the principles. The watermarking techniques can be coarsely divided into two classes [10], the correlation-based watermarking, and the noncorrelation-based watermarking.

- Correlation-based watermarking:

The correlation-based watermarking technique directly adds watermark to the host image with an intensity factor g, and the watermark is a pseudorandom noise pattern, which is generated based on a seed and consists of the integers {-1, 0, 1}. As shown in Figure 2-13, the watermark is multiplied by a gain factor g and then be added to the host image to form the watermarked image, that is, $I' = I + k \cdot W$.

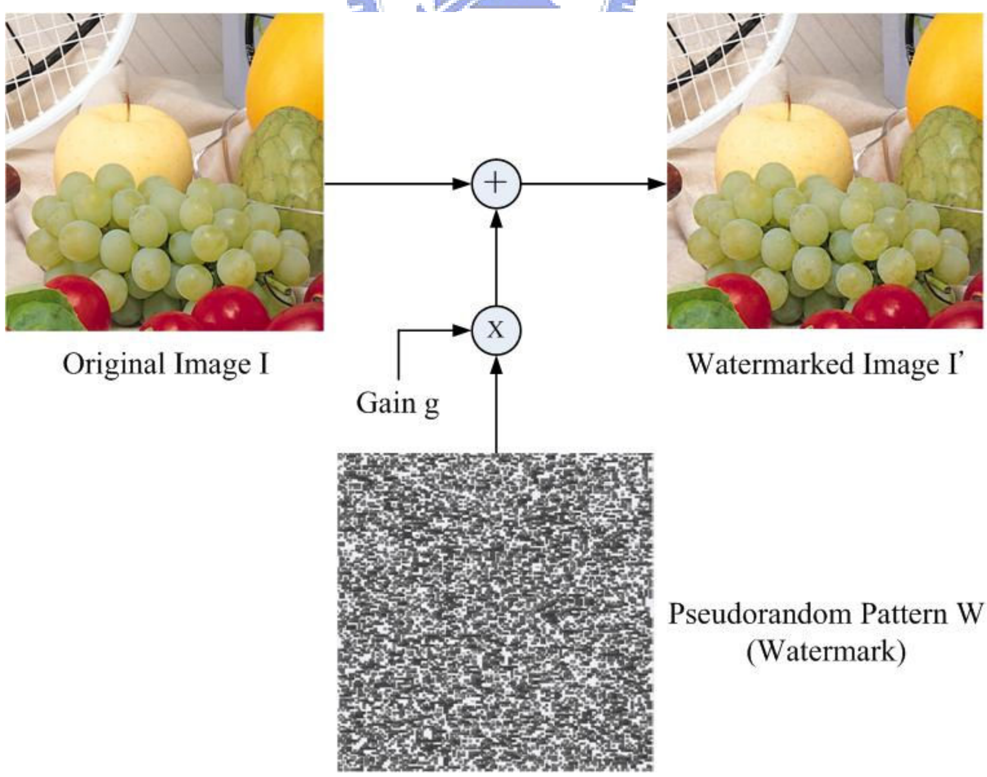


Figure 2-13 Embedding procedure of correlation-based watermark

For watermark detection, we calculate the correlation between the watermarked image and the pseudorandom noise pattern. Since the watermark is pseudorandom, we assume that the correlation between the host image and the watermark is almost equal to zero. Therefore, the correlation between the watermarked data and the watermark is the same as the autocorrelation of the watermark itself. Thus, if we obtain a correlation with high positive value in the watermark detection process, we could say that the image is watermarked; and if we use the wrong watermark pattern to inspect the watermarked image, the correlation value should be very low due to the characteristics of the pseudorandom pattern. Figure 2-14 is a statistical result in [10], the seed is used to generate the pseudorandom pattern, and different seeds result in different patterns, which are almost uncorrelated to each other. The diagram shows that the correlation value between the watermarked image and the pseudorandom pattern is close to zero unless the seed happens to be the seed which generates the watermark embedded in the host image.

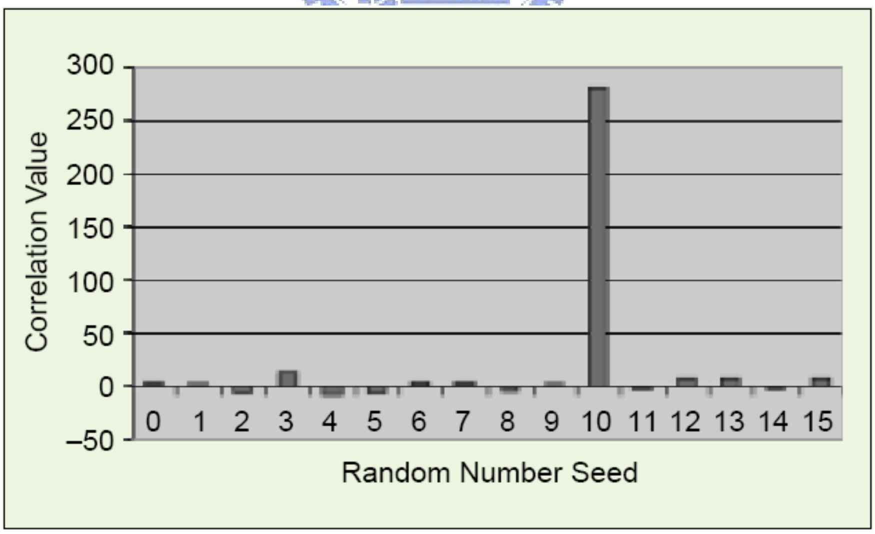


Figure 2-14 Correlation values between the watermarked image and the patterns generated by different seeds and the seed used to generate the watermark is 10

Since the correlation value of a wrong pattern is not exactly equal to zero, a threshold used to decide whether the image is watermarked is required. If the correlation value is below the threshold T, we say that the image is un-watermarked. This detection scheme causes two types of errors named false positive and false negative. False positive means that a watermark is detected although there is none, and false negative rejects the existence of the watermark even though there is one. Normally, a higher gain is leads to a lower the error probability, but it also lowers the watermarked image quality.

In the above method, we can only embed one-bit data in an image by a watermark pattern. The way to embed several bits of data is to divide the host image into several embedding blocks and embed each bit of data to the embedding block, respectively. Figure 2-15 is a simple example of watermarking of four bits in one image, the host image is first divided into four sub-pictures, and the four watermarks are embedded correspondingly into each sub-picture with the corresponding intensity.

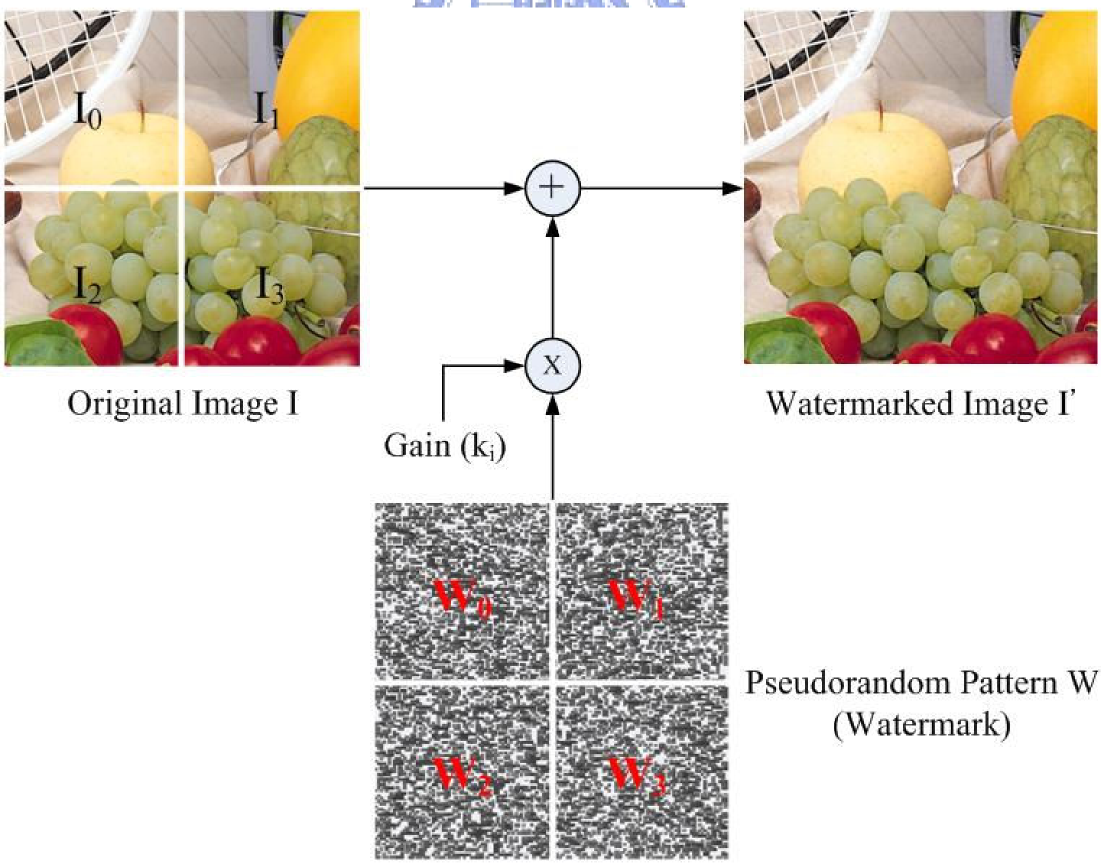


Figure 2-15 Four bits watermark embedding

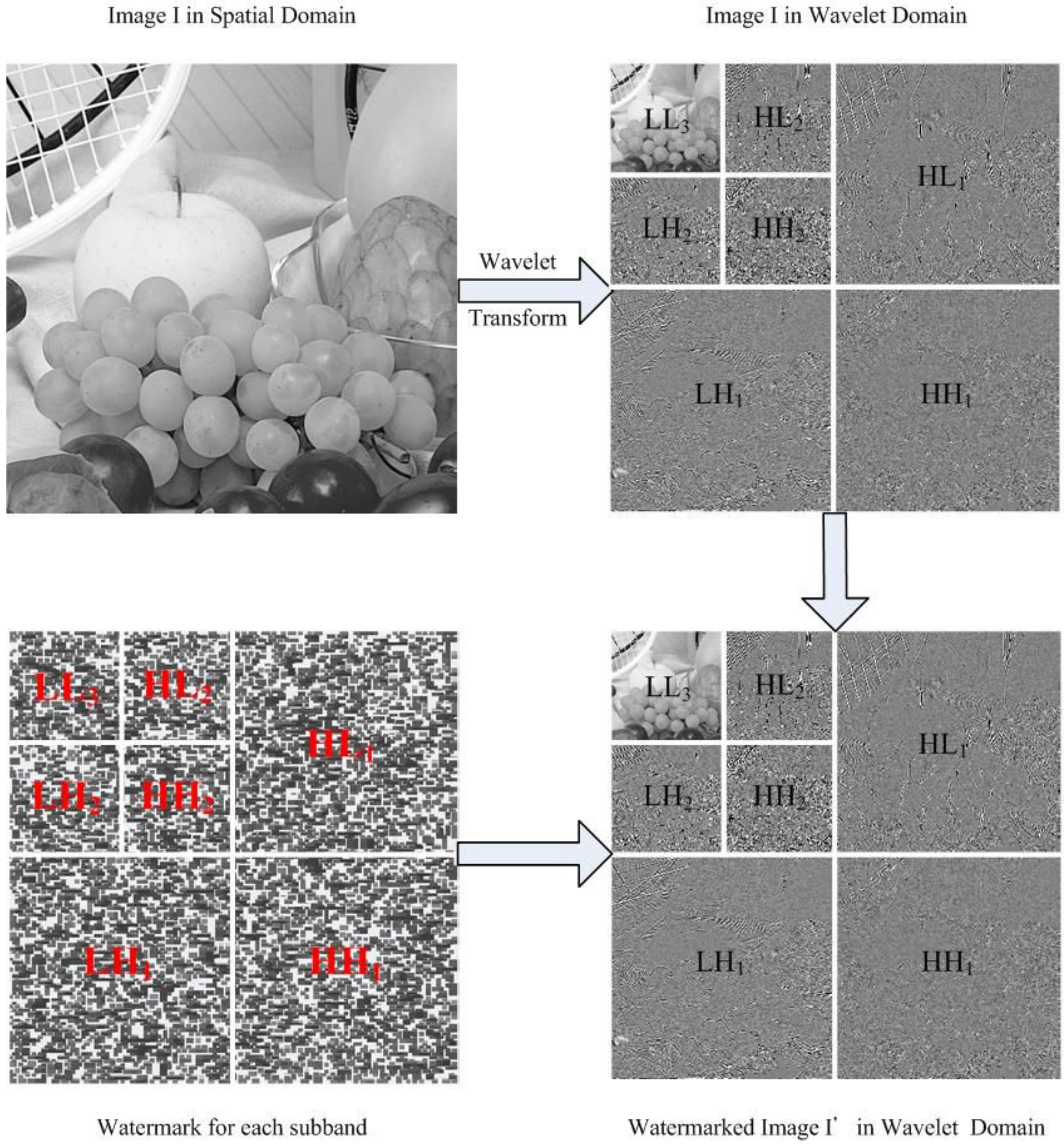


Figure 2-16 Watermarking in wavelet domain

The correlation-based watermarking technique mentioned above is operated under the spatial domain. The watermark can be also embedded in frequency domain such as wavelet transform (WT) domain, discrete cosine transform (DCT) domain, and Fourier transform (FT) domain. Each frequency domain has its own properties, and we can use these characteristics to improve the efficiency of watermarking. For instance, as the characteristics of the human visual system (HVS) model (which will be introduced in later section), human eyes are less sensitive to disturbances in high frequency domain, and thus we can embed a watermark with higher intensity into the high frequency domain to increase the robustness of watermark without seriously damaging the quality of the host image. Figure 2-16 shows an instance of embedding watermark in wavelet domain, the watermark of each subband is independent and is allowed to have different intensity to improve the efficiency of watermarking. Other watermarking techniques which embed the watermark in discrete cosine transform (DCT) domain or Fourier transform (FT) domain have the similar operations, but since they are difficult to display by graph, we neglect them here for convenience.

- Noncorrelation-based watermarking:

There are various types of noncorrelation-based watermarking such as least significant bit (LSB) modification, DCT coefficient ordering, and block-based watermarking.

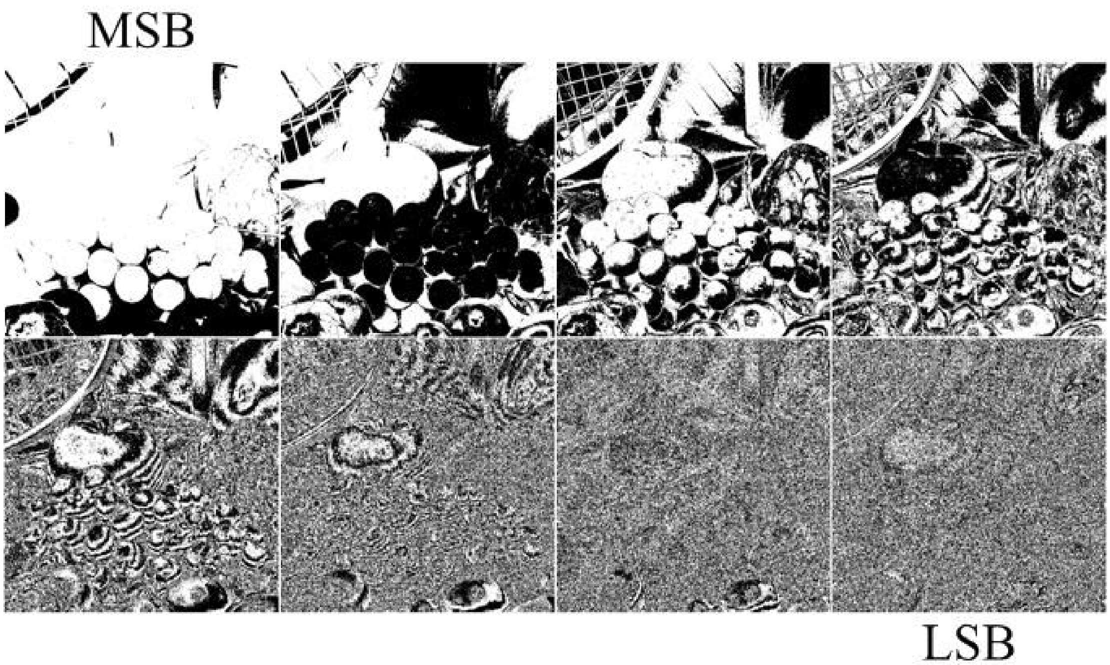
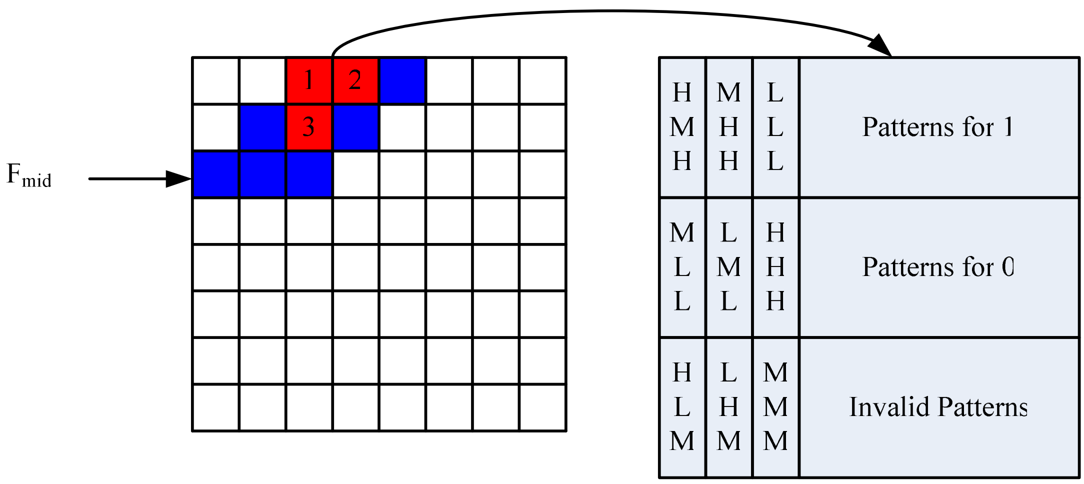


Figure 2-17 Bit-planes of the fruit image

Least significant bit (LSB) modification is the simplest example of spatial domain watermarking techniques. This method directly replaces the LSB bit-plane of the host image by an enormous amount of watermark bits, and of course, it is certainly not a secure scheme because the LSB bit-plane is easy to be interfered by random noise and the watermark is then destroyed. As shown in Figure 2-17, since the LSB bit-plane of the host image has almost no information about the image, replacing the LSB bit-plane by the watermark leads to no perceptual difference between watermarked image and the host image.

DCT coefficient ordering method was proposed in [11] and described more clearly in [10]. This method reorders the DCT coefficients of each block in a specific order which stands for different embedding information. First, the host image to be watermarked is divided into several 8x8 blocks, and DCT is done for each block. Then, two or three DCT coefficients in the mid-band F_{mid} are selected to be quantized using the default JPEG quantization table and relatively low JPEG quality factor. Finally, quantized coefficients are reordered according to a specific order shown in Figure 2-18. For example, if we want to embed a bit with value 0, the third selected coefficient should have the lowest value.



8x8 DCT block with possible locations for embedding a bit

Relations among three selected quantized coefficients
H : high M : meddle L : low

Figure 2-18 DCT coefficient ordering method of watermarking

Block-based watermarking [12] [13] divides the host image into several embedding blocks, and try to tune the mean value of each block by changing some coefficients' value of the block. In [12], the mean of a block with even value represents that the watermark bit of value 0 is embedded while with odd value represents that 1 is embedded. In [13], the mean value of a block with zero value represents that the embedded data is 1 while the embedded data with value 0 makes the mean of block exceed a threshold value P.

Another method is the bit-plane-based method. In this method, the watermark is embedded into a specific bit by directly replacing the bit by the watermark bit. LSB modification method mentioned above is a well known example of this method, but there are still many variations of this method such as the method described in [15]. For this method, the watermark is embedded into bits which may not be in the LSB bit-plane, and use the Torus Automorphisms (TA) technique to protect the security of the embedded information.

2.2.6 Human Visual System (HVS) Model

The HVS model is useful for tuning the intensity of the watermark to improve the efficiency of trade-off between robustness and imperceptibility. The watermarking technique which takes advantage of the HVS model embeds a watermark of higher intensity without affecting the perceptual influence compared to the one which does not use the HVS model.

We can view the HVS model as three independent stages [6], that is, a receiver, the eye and retina, a transmission channel, the optic nerve, and a processing engine, the brain. Since the knowledge of the brain behavior is still unclear, to construct a perfect HVS model is very difficult. In this section, we only try to understand the features of the HVS model for the watermark designer to use in a synthetic way and from engineering perspective.

2.2.6.1 Introduction to the Human Visual System (HVS) model

Wandell [16] uses three successive stages, that is, encoding, representation, and interpretation to divide the human vision. Encoding stage corresponds to the transformation of light into electric signals by the retina. Representation stage is a process of representing the encoded image by different visual pathways tuned to specific characteristics of the visual signal. Interpretation stage is the highest level of the human vision and is located in the brain. We should note that it is very complex and involves so much subjectivity in the interpretation stage, however, the, motion, depth, and color appearance which can be roughly modeled are also belong to this stage.

The retina of human eyes has three kinds of photoreceptors, called red, green, and blue cones. They reach their maximum sensitivity to the light with specific wavelength corresponding to these basic colors. All colors are first encoded as a function of these basic colors, but with the progressing of the research of the human visual system, we could find that a better color space is produced, that is, one achromatic and two chromatic channels. There are many kinds of color spaces of this concept are proposed, such as CIE Luv, CIE Lab, and the YCbCr color space is the most popular one.

The achromatic channel is also called the luminance channel and most modifications of the color images are only done in this domain for simplicity because the characteristics of the chromatic channels are still not unexplored.

2.2.6.2 Visibility Threshold of the Human Visual System

In this section, we introduce the concepts only and skip the detailed formulas. The human visual system (HVS) is not sensitive to the stimuli which is too small and cannot discriminate between signals with an infinite precision. This is also why a lossy compression system could compress an image with no perceptual difference. The definition of HVS sensitivity is first depends on the anatomy of the eye, which leads to the discovery of effects like refraction, diffraction, color aberration, and spatial sampling process. In addition, HVS sensitivity also depends on the characteristics of the visual signal. Unfortunately, it is too complex to analyze the whole HVS by only the method of anatomy. Thus, another method is proposed and worked fine, that is, experiments consisting in the detection and discrimination of patterns by human eyes. Several effects describing the properties of HVS sensitivity are listed below.

- **Just Noticeable Difference**

Just noticeable difference (JND) threshold is a threshold that any noise with value below it can be ignored since the human eye will not discover the difference. JND threshold depends on the features of both the signal and the background pattern.

Weber-Fechner's law states that "if the luminance of a test stimulus is just noticeable from the surrounding luminance, then the ratio of the luminance difference to the surrounding luminance is approximately constant" [17], which implies that a noise is easier to be detected in a dark region. However, the experimental conditions of this law are too simple in comparison to the real conditions of image viewing. To complement Weber-Fechner's law, Albert Munsell proposed another system, the Munsell Renotation System [18]. In this system, the lightness is pointed out to be nonlinearly related to the luminance signal and this law has been used to shape the γ characteristic of displays.

- **Contrast Sensitivity Function**

As the frequency is concerned, JND threshold is also considered in the frequency domain. To determine the JND threshold in frequency domain, extensive psycho-visual measurements have been performed using gratings [19], that is, simple sinusoidal signals of given spatial frequency and orientation with a window extension. Mannos and Sakrison proposed the contrast sensitivity function (CSF) (2.2.6-1) [20]. In (2.2.6-1), L_{max} and L_{min} is the maximal and minimal luminance of the grating, f is the spatial frequency; K0, K1, a, and α depend on parameters such as mean luminance, temporal frequency and orientation. Although CSF may be more precise, it also has many constraints to be satisfied, so it should be use cautiously.

$$C = \frac{L_{\max} - L_{\min}}{L_{\max} + L_{\min}}$$
$$CSF(f) = K_0 \cdot (1 + K_1 f) \cdot e^{(a \cdot f)^{\alpha}} \qquad (2.2.6\text{-}1)$$

2.2.6.3 Visual Masking

When the interference between two or more signals is concerned, their visibility may be increased, or more often, decreased, and this effect is also called the masking effect. All we care about is the ability for the host image to hide other signals, which is important for watermarking system. The masking effect contains the spatial masking, the contrast or pattern masking, and noise masking, and so forth. Since the masking effect is too complex, we will skip the introduction of them, the reader could find more details in [6].

2.3 Layered Protection of Scalable Media

With the widespread use of Internet, all digital data are easy to obtain, thus, a protection scheme is needed for the intellectual property. Moreover, scalable media coding becomes more and more important due to the various bandwidth of transmission such as mobile communication. Scalable coding implies that a multimedia is divided into one base layer and several enhancement layers, and each enhancement layer could independently improve certain quality of the reconstructed image.

There are three categories of scalability, that is, spatial scalability, temporal scalability, and signal-to-noise (SNR) scalability. Spatial scalability means that each layer has different resolution, and we could get the reconstructed image with higher resolution when more enhancement layers are decoded. Temporal scalability is a scheme that enhancement layers increase the overall frame rate of a film. And SNR scalability makes different layers have different amount of quality to improve.

Traditionally, each layer of scalable coding is protected by cryptographic techniques, but the synchronization problem [21] between the encryption key and the encrypted content is still an issue. Therefore, if we could transmit the encryption key with the encrypted content, the synchronization problem will be resolved. Thus, a scheme which combines encryption and robust watermarking is proposed [22]. Before introducing this scheme, a typical MPEG-2 conditional access receiver [23] is introduced.

As shown in Figure 2-19, the digital content is protected by "Control Word" (CW), and the control word can be obtained through a two-step decryption flow, that is, to retrieve CW, the Service Key (SK) should be decoded first, and the decoding of SK is based on the combination of User Key (UK) and the Entitlement Management Message (EMM). User Key may be contained in a Smart Card and the EMM can be used to change the status of the user accessibility of contents, that is, if the subscription is overdue, a broadcaster could disable a user's access by changing the EMM. After decoding the SK, an Entitlement Control Message is required to decode CW. Because of the complicated computation of the Control Word, it is hard to handle the synchronization problem of cryptographic techniques. Thus, a new architecture is proposed [22] and shown in Figure 2-20; in this scheme, the EMM and ECM are embedded into the content and transmitted together, then the problem of synchronization is resolved. We should note that the ECM and EMM in Figure 2-19 are corresponding to the key function key() and the watermark W_i in Figure 2-20, and UK, SK, and CW are corresponding to Gi, Fi, and Ki.

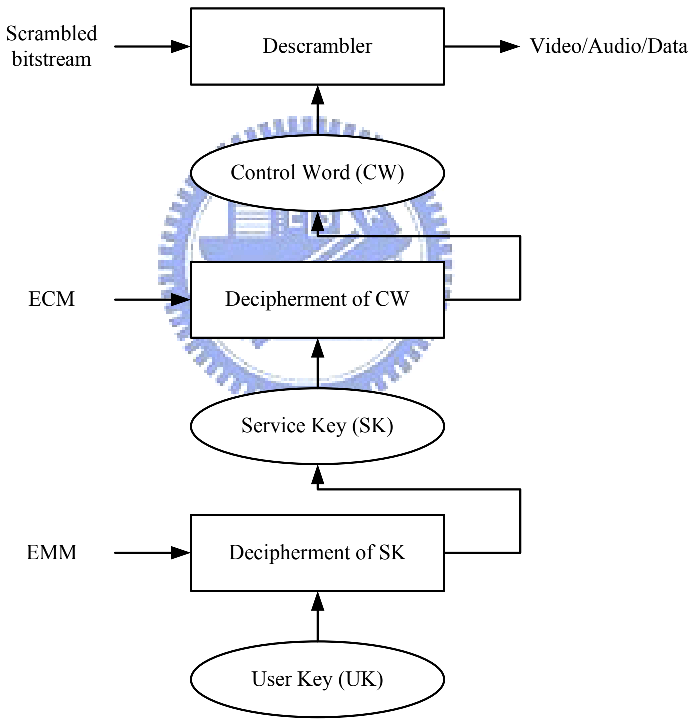


Figure 2-19 A typical MPEG-2 conditional access receiver

In Figure 2-20, X_i is the encrypted enhancement layer which can be decrypted using the decryption key K_i, and E_i is the resulting enhancement layer used to form the ith constructed base layer B_i; W_i is the watermark extracted from the constructed base layer B_{i-1} with extraction parameter P_{i-1}, and F_i is the secret information decrypted by the specific key G_i that can be used to get the decryption key K_i and extraction parameter P_i. In short, the relationships between all coefficients in Figure 2-20 are listed in (2.3-1) to (2.3-6).

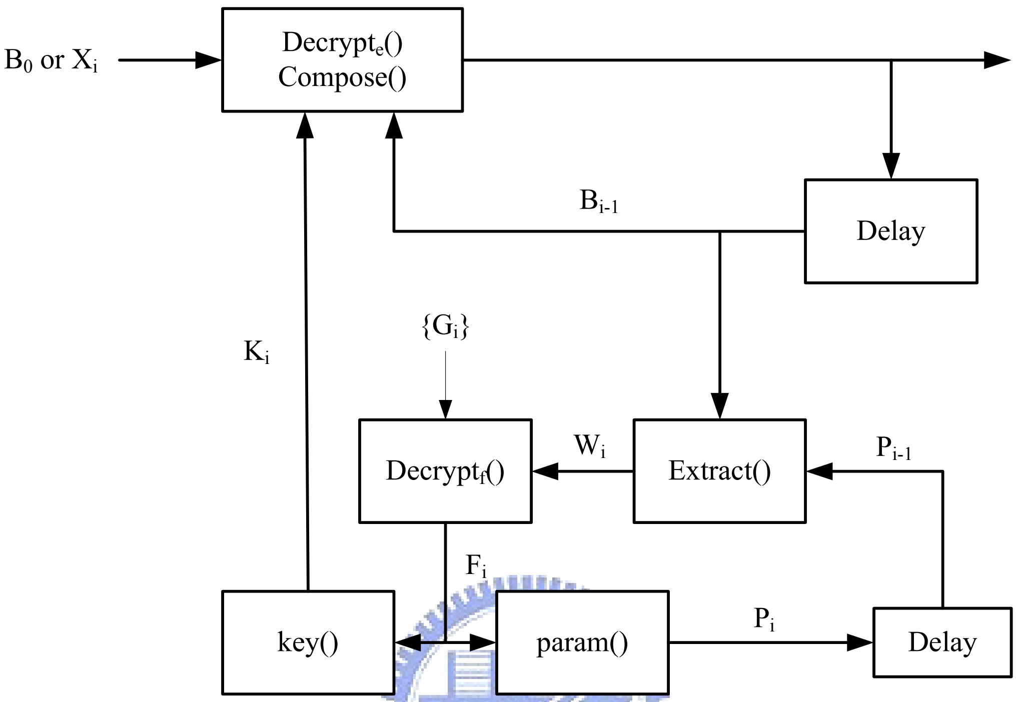


Figure 2-20 Decryption and decoding of layer-protected content

$$E_i = Decrypt_e(X_i, K_i) \qquad (2.3\text{-}1)$$

$$B_i = Compose(B_{i-1}, E_i) \qquad (2.3\text{-}2)$$

$$W_i = Extract(B_{i-1}, P_{i-1}) \qquad (2.3\text{-}3)$$

$$F_i = Decrypt_f(W_i, G_i) \qquad (2.3\text{-}4)$$

$$K_i = key(F_i) \qquad (2.3\text{-}5)$$

$$P_i = param(F_i) \qquad (2.3\text{-}6)$$

Chapter 3 Properties of Various Watermarking Methods

In this chapter, few watermarking techniques will be introduced in detail. We will describe and analyze the characteristics of each watermarking technique and simulate them to examine their features. At the end, we will choose a watermarking technique that is pertinent to the layered protection of scalable media introduced in section 2.3.

3.1 Introduction to Watermarking Techniques

As described in section 2.2.5, watermarking techniques can be divided into two main groups, the correlation-based and the noncorrelation-based watermarking. Here we will introduce two examples in detail and describe their characteristics.

3.1.1 The correlation-based watermarking

The correlation-based watermarking is a simple way to embed watermark. As shown in Figure 3-1, the watermark with a chosen intensity is added to the host image. The procedure of watermark detection is to calculate the correlation between the watermark and the watermarked image. If the correlation value is below a threshold value T, it is claimed that the image is un-watermarked; on the contrary, if the correlation value is higher than T, the image is watermarked.

If we want to embed information into the host image, for example, a one-bit data, then we can embed “0” by adding a negative watermark pattern into the host image and embed “1” by adding a positive watermark pattern into the host image. Thus, if the correlation value is below –T, we say that the embedded data is “0”, and if the correlation value is above T, the embedded data is “1”. Then, if we want to embed several bits into an image, we could divide the image into several blocks and embed one bit into a block. Figure 3-2 is an example of embedding “0110” into the host image. We should note that (k_0, k_3) should be negative and (k_1, k_2) should be positive.

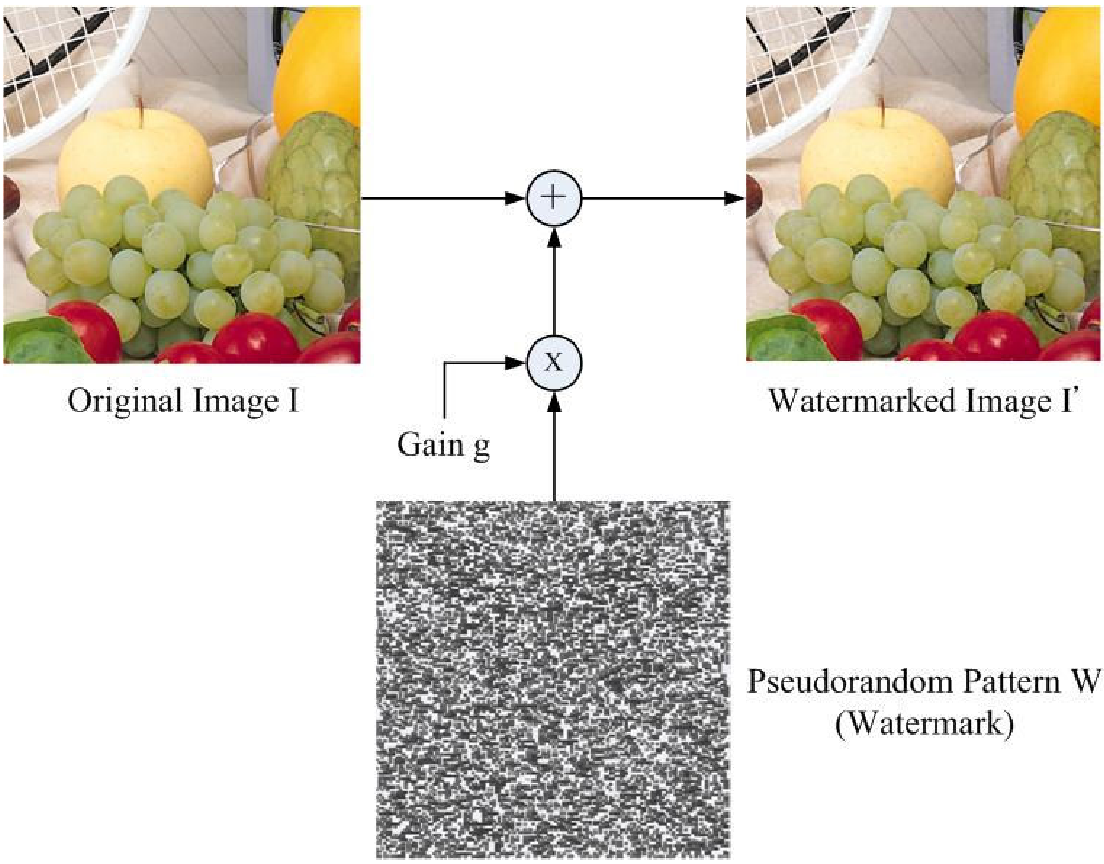


Figure 3-1 Embedding procedure of the correlation-based watermark

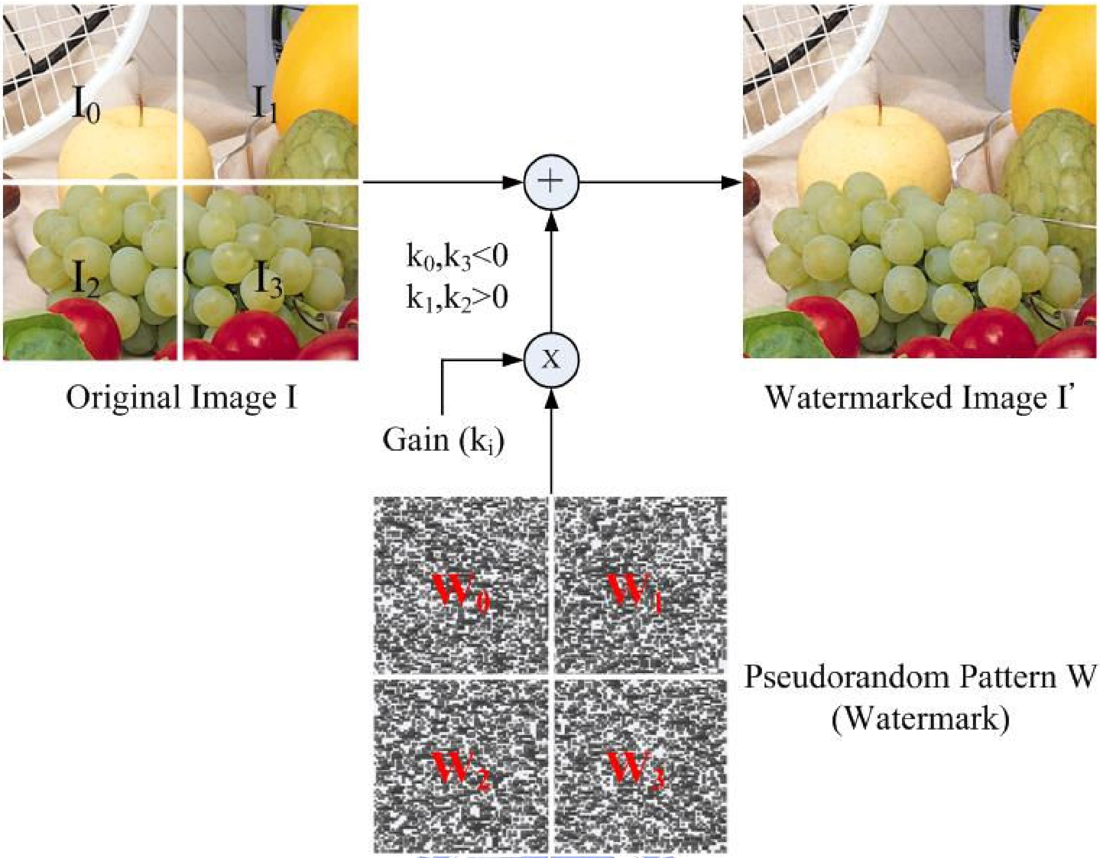


Figure 3-2 Embedding "0110" into the host image

After introducing the basic concept of correlation-based watermarking, we will introduce a watermarking scheme which combines the human visual system (HVS) to improve the embedding efficiency [24]. In this scheme, the embedding position of each bit is first decided, and the HVS model is then used to decide the watermark intensity to improve the efficiency of embedding. The details will be discussed as follows.

- Watermark Embedding

As shown in Figure 3-3, the image to be watermarked is first decomposed through DWT in four levels: I_l^θ is the subband at resolution level l = 0,1,2,3 and with orientation θ = {0,1,2,3}. The watermark consists of a pseudorandom binary sequence (±1) and is embedded into the coefficients of the largest resolution I_0^θ. Also, we can embed the watermark into coefficients of the other resolutions to achieve higher robustness but with a more interfered watermarked image.

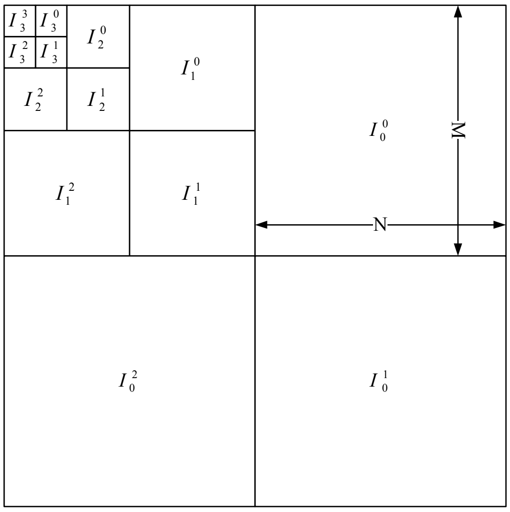


Figure 3-3 An image with four decomposition levels using DWT

In more details, a 1-D pseudorandom binary sequence m_h is rearranged to a 2-D sequence $x^{\theta}(i,j)$ (3.1.1-1) where M and N are the dimensions of the largest subband as shown in Figure 3-3. And $x^{\theta}(i,j)$ is directly added to the host image (3.1.1-2), where $\tilde{I}_0^{\theta}(i,j)$ is the watermarked image, $I_0^{\theta}(i,j)$ is the host image, α is a global parameter accounting for watermark strength, and $w^{\theta}(i,j)$ is a weighting function that allows to match the masking characteristics of the HVS.

$$x^{\theta}(i,j) = m_{(\theta MN + iN + j)} \qquad (3.1.1\text{-}1)$$

$$\tilde{I}_0^{\theta}(i,j) = I_0^{\theta}(i,j) + \alpha w^{\theta}(i,j) x^{\theta}(i,j) \qquad (3.1.1\text{-}2)$$

- Human Visual System (HVS)

The human visual system is proposed by Lewis and Knowles [25] to improve the quality of image compression and is modified by Barni, Bartolini and Piva [24] to refine the efficiency in watermarking application. The HVS model is developed by the observing characteristics [24].

1. The eye is less sensitive to the noise in high resolution bands and the bands having orientation of 45° (i.e. θ= 1); that is, the eye is less sensitive to the noise with higher frequency.
2. The eye is less sensitive to the noise in high or low brightness areas of an image.
3. The eye is less sensitive to the noise in the highly textured areas and near the edges.

Base on the three observations, a HVS model is constructed as expressed by (3.1.1-3), where $q_l^\theta(i,j)$ is the quantization step used for image compression in [25].

$$q_l^\theta(i,j) = \Theta(l,\theta)\,\Lambda(l,i,j)\,\Xi(l,i,j)^{0.2} \qquad (3.1.1\text{-}3)$$

The three terms used to decide the quantization step are described as follows.

The first term $\Theta(l,\theta)$ is associated with the noise sensitivity depending on the band; that is, the band with higher frequency has larger $\Theta(l,\theta)$ as expressed by (3.1.1-4).

$$\Theta(l,\theta) = \begin{Bmatrix} \sqrt{2}, & if \quad \theta = 1 \\ 1, & otherwise \end{Bmatrix} \cdot \begin{Bmatrix} 1.00, & if \quad l = 0 \\ 0.32, & if \quad l = 1 \\ 0.16, & if \quad l = 2 \\ 0.10, & if \quad l = 3 \end{Bmatrix} \qquad (3.1.1\text{-}4)$$

Since the eye is less sensitive to the noise in the high or low brightness areas and the coefficients of the lowest resolution in the DWT domain represent the brightness of certain area in an image, the second term $\Lambda(l,i,j)$ takes into account the local brightness based on the gray-level values of the low pass version of the image. In (3.1.1-5), $L(l,i,j)$ is the gray-level values of the low pass version of the image.

Based on the consideration that the human eye is less sensitive to changes in very dark region as well as very bright region, a modified $L'(l,i,j)$ is used to estimate the strength of tolerable noise.

$$\Lambda(l,i,j) = 1 + L'(l,i,j) \qquad where$$
$$L'(l,i,j) = \begin{cases} 1 - L(l,i,j), & if\ L(l,i,j) < 0.5 \\ L(l,i,j), & otherwise \end{cases} \qquad (3.1.1\text{-}5)$$
$$and \quad L(l,i,j) = \frac{1}{256} I_3^3 \left(1 + \left\lfloor \frac{i}{2^{3-l}} \right\rfloor, 1 + \left\lfloor \frac{j}{2^{3-l}} \right\rfloor \right)$$

Finally, the third term $\Xi(l,i,j)$ gives a measure of texture activity in the neighborhood of the pixel. In particular, this term is composed by the product of two contributions: the first is the local mean square value of the DWT coefficients in all higher subbands, while the second is the local variance of the low-pass subband. The first item correlates to the distance from the edges, whereas the second one represents the texture. These two terms are multiplied together to reflect the initial consideration that the eye is less sensitive to noise in highly textured areas and near the edges.

$$\Xi(l,i,j) = \sum_{k=0}^{3-l} \frac{1}{16^k} \sum_{\theta=0}^{2} \sum_{x=0}^{1} \sum_{y=0}^{1} \left[I_{k+l}^{\theta}\left(y + \frac{i}{2^k}, x + \frac{j}{2^k}\right) \right]^2 \cdot Var\left\{ I_3^3\left(1 + y + \frac{i}{2^{3-l}}, 1 + x + \frac{j}{2^{3-l}}\right) \right\}_{\substack{x=0,1 \\ y=0,1}} \qquad (3.1.1\text{-}6)$$

The fact that $q_l^{\theta}(i,j)$ is chosen as the quantization step for image compression, implies that disturbs having value lower than $q_l^{\theta}(i,j)/2$ are assumed to be un-perceptible. Thus, the weighting function $w^{\theta}(i,j)$ in (3.1.1-2) is set to (3.1.1-7).

$$w^{\theta}(i,j) = q_0^{\theta}(i,j)/2 \qquad (3.1.1\text{-}7)$$

- Watermark Detection

Watermark detection is accomplished without referring to the original image and is called as the blind detection. The correlation between the watermarked DWT coefficients and the watermarking sequence to be tested for presence is computed by (3.1.1-8), and the correlation value is compared to a threshold T_ρ chosen in such a way to grant a given probability of false positive detection. If ρ is below T_ρ, we would say the image is un-watermarked, while the image is said to be watermarked if ρ is above T_ρ.

$$\rho = \frac{1}{3MN}\sum_{\theta=0}^{2}\sum_{i=0}^{M-1}\sum_{j=0}^{N-1}\tilde{I}_0^{\theta}(i,j)\,x^{\theta}(i,j) \qquad (3.1.1\text{-}8)$$

3.1.2 The bitplane-based watermarking

The bitplane-based watermarking is the method which directly replaces the original bits of specific positions by the embedding data. As shown in Figure 3-4, there are eight coefficients with eight bits per coefficient, and each of them chooses a specific position (i.e. the colored area) for data embedding. Since the embedding mechanism is simply replacement, if we know the embedding positions, the embedded data can be easily extracted without any mathematical operation.

8th	1	1	0	1	0	0	1	0
7th	0	1	0	0	1	1	0	0
6th	0	0	1	1	0	1	0	1
5th	0	1	0	0	1	0	0	0
4th	1	0	1	0	0	1	1	0
3rd	1	0	0	1	0	0	0	0
2nd	0	1	1	0	1	1	0	1
1st	0	0	0	1	1	1	1	0

Figure 3-4 Embedding data “11100111” in specific positions

The embedding efficiency of this method depends on the choice of the embedding positions. Thus, how to choose the embedding positions is a big issue. The most common one is the LSB modification described in section 2.2.5 but this method is too fragile and the watermark can be destroyed easily. Another method is proposed by T.S. Chen, J. Chen and J.G. Chen; they embedded the data in the fifth bit of the coefficients and smartly modified the other bits to reduce the quality loss [15]. Thus, the proposed method makes a good balance between robustness and imperceptibility.

3.2 Analysis of Watermarking Techniques

In this section, we will discuss advantages and disadvantages of the watermarking techniques described above.

- Side information

Since the extraction procedure of correlation-based watermarking is only to compute the correlation value between the watermarked image and the watermark, the side information for watermark extraction is only the seed used to generate the watermark pattern.

For bitplane-based watermarking, the embedding positions are needed for watermark extraction. And since that recording the embedding positions needs a lot of extra bits, the side information required for the bitplane-based watermarking is much more than the side information required for the correlation-based watermarking.

- Data Capacity

The data capacity is the data size for an image to hide data imperceptibly. We should note that the correctness of correlation-based watermarking is based on the randomness of the pixels for data embedding in an image; that is, the more random the pixels are, the more correct extraction results the extracted watermark will have. Thus, we use more pixels to embed one-bit data to keep the pixels random enough, and the cost is the lower data capacity.

Since the bitplane-based watermarking directly replaces the bits of the embedding positions by the embedding data, its data capacity is theoretically the same as the number of pixels and is much larger than the data capacity of the correlation-based one.

- Robustness

As most people know, the correlation-based watermarking is a very robust scheme since it is closely combined with the coefficients of the host image. And the bitplane-based watermarking is more fragile and can be destroyed by only re-quantizing the coefficients using another quantization step; that is, if a coefficient is re-quantized, we will find the embedded data in neither the original positions nor the other positions.

- Security

The security refers to how difficult the embedded data are extracted by an unauthorized person. For correlation-based watermarking, if the seed used to generate the watermark pattern is known, there will be no secret of the embedded data. For bitplane-based watermarking, the security is based on the embedding positions. Thus, if the embedding positions are known by an unauthorized user, the embedding data can be extracted directly.

Since the information of embedding positions is often much more than the seed used to generate the watermark pattern, typically the bitplane-based watermarking is more secure. But if we fix the embedding positions in a specific bitplane to reduce the side information in some applications, the bitplane-based watermarking can be less secure but the side information is small.

- Computational Complexity

Because of the extraction procedure of the correlation-based watermarking is to calculate the correlation value between the watermark pattern and the watermarked image, its computational complexity is much higher than the complexity of the bitplane-based watermarking. The extraction procedure of the latter is only examining the embedding positions and directly draws out the embedded data. This drawback of the correlation-based watermarking also makes it hard to implement in a real-time system.

• Removability

Although an imperceptible watermark is acceptable in most applications, there are still some applications such as storage of medical images cannot tolerate any distortion. In these applications, a scheme of removable watermarking is necessary.

As shown in Figure 3-5, to remove a watermark embedded by correlation-based watermarking is simply subtracting the watermark pattern from the watermarked image. On the other hand, because the watermark embedding procedure of the bitplane-based watermarking is the replacement, to remove a watermark without other side information is not possible.

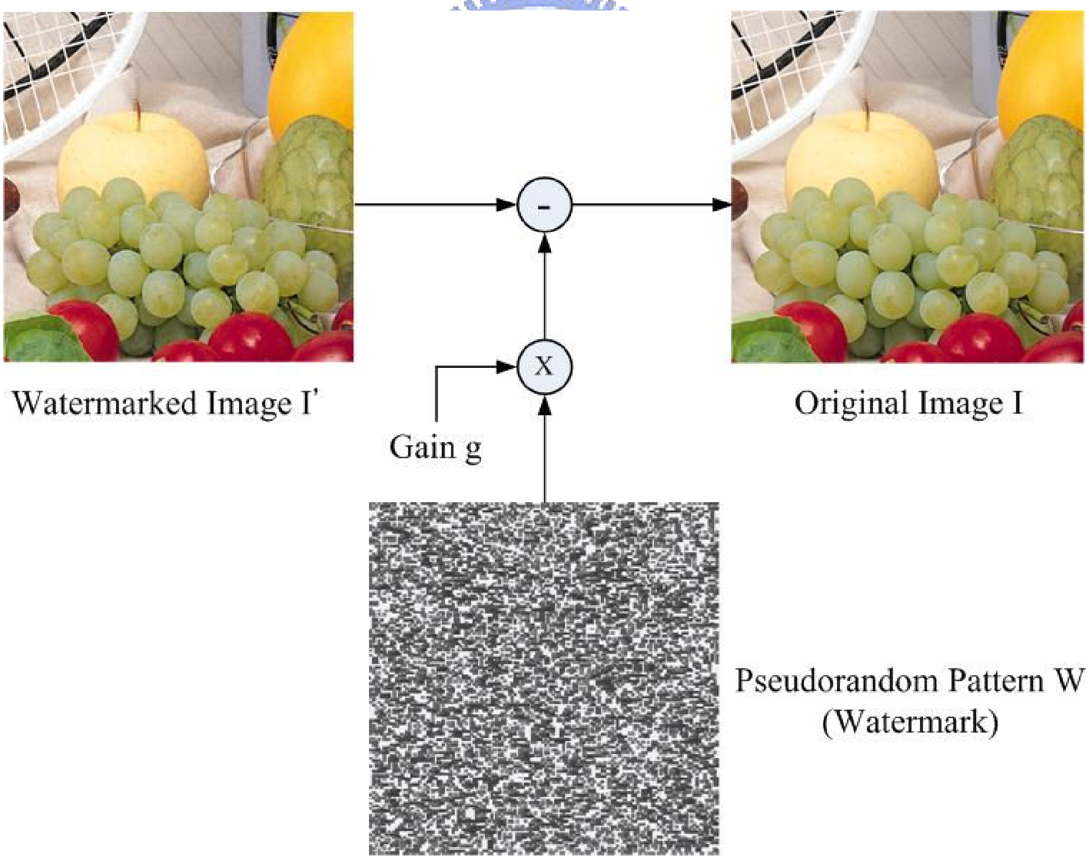


Figure 3-5 Procedure of removing the watermark from the watermarked image

3.3 Simulations

In this section, we run a simple experiment to verify some features of the watermarking techniques we discussed earlier. The watermarks are embedded in the commonly used test images such as baboon, fruit, lena and peppers, which are shown in Figure3-6.

Figure 3-6 Test images

Before listing the results of simulation, the embedding algorithm is introduced. At first, the host image is decomposed into four resolutions shown as Figure 3-3. And areas of the three higher resolutions are then chosen to embed watermarks using the correlation-based and the bitplane-based watermarking techniques.

In our simulations, 160 bits are embedded in the I_0 area, 128 bits are embedded in the I_1 area and 64 bits are embedded in the I_2 area. The strengths of watermarks are tuned to make the embedding imperceptible, and the effect of image compression is also considered; that is, the watermark extraction of a watermarked image passed through a JPEG2000 encoder with compression rate of 20 is conducted. The simulation is done 100 times with different watermarks to be embedded and the mean values of the statistics are shown in Tables 3-1 and 3-2. The bit error rate (BER) is the ratio of the number of bits that the extracted data are different from the embedded data to the total bits of the embedded data. If the bit error rate (BER) is approximate to 0.5, it means that the watermark is almost destroyed.

In Tables 3-1 and 3-2, the bitplane-based watermarking has a much better performance before passing through the JPEG2000 encoder, but the performance decreased substantially after coded by JPEG2000. Thus, it can be said that the bitplane-based watermarking has a much larger data capacity but is much more fragile to the image compression as compared to the correlation-based watermarking. The simulation results match our predictions in section 3.2.

	baboon	fruit	lena	peppers
BER_0	0.000000	0.000000	0.000000	0.000000
BER_1	0.021250	0.000000	0.000000	0.000000
BER_2	0.059531	0.040156	0.017656	0.003594
PSNR	40.056029	41.379366	41.076705	40.336315
After JPEG2000 compression				
BER_0	0.018250	0.000000	0.000000	0.021562
BER_1	0.102188	0.041016	0.003281	0.016328
BER_2	0.090938	0.058906	0.024531	0.021250
PSNR	25.3193	35.1496	36.6823	36.0448

Table 3-1 Results of correlation-based watermarking

	baboon	fruit	lena	peppers
BER_0	0.000000	0.000000	0.000000	0.000000
BER_1	0.000000	0.000000	0.000000	0.000000
BER_2	0.000000	0.000000	0.000000	0.000000
PSNR	47.121854	47.430821	47.490265	47.477924
After JPEG2000 compression				
BER_0	0.487031	0.507734	0.487031	0.499531
BER_1	0.507734	0.499531	0.487031	0.487031
BER_2	0.073750	0.062812	0.002188	0.000625
PSNR	25.3914	35.9127	38.0789	37.5366

Table 3-2 Results of bitplane-based watermarking

3.4 Choice of Watermarking Techniques

Since our scalable protection scheme is based on the JPEG2000 standard, we will discuss the compatibility of the watermarking techniques with the JPEG2000 standard in this section. Although each watermarking technique has its own merits, not all of them are suitable for the JPEG2000 standard and some advantages of theirs may disappear due to the JPEG2000 standard characteristics. Some other implementation considerations are also discussed in this section.

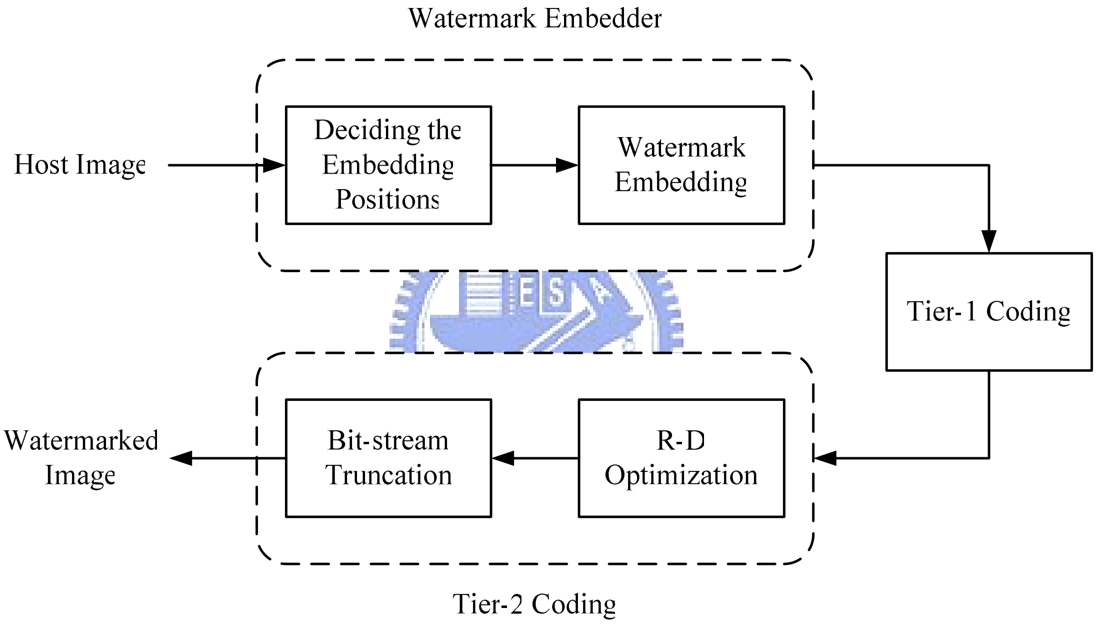


Figure 3-7 Choosing embedding positions before R-D Optimization

In the JPEG2000 standard, there are three coding passes for each bit-plane coding and the coded bit-stream can be rearranged to achieve different types of scalability. That is, bits in the same bitplane may belong to different coding layers. Furthermore, since the tier-2 coding truncates the coded bit-stream to achieve the demanded compression rate or compressed quality, bits to be truncated in a bitplane is unknown until the rate-distortion optimization is done. Thus, bits which will be preserved and bits which will be truncated are unknown at the tier-1 coding stage. For this reason, how to choose the embedding positions for bitplane-based watermarking without perceptually affecting the image quality before entropy coding is a difficult problem. If we choose the embedding positions before the tier-2 coding as shown in Figure 3-7, the embedded information may be truncated due to the rate-distortion optimization process. If we choose the embedding positions after the tier-2 coding as shown in Figure 3-8, the truncation information of the coded bit-stream is sent to the watermark embedder and embedding positions are then decided. After watermark was embedded, the entropy coding is redone. Also, the rate-distortion optimization is redone and the truncation points may be changed. Thus, some embedded bits may be truncated again and the correctness of the watermarking positions is not assured.

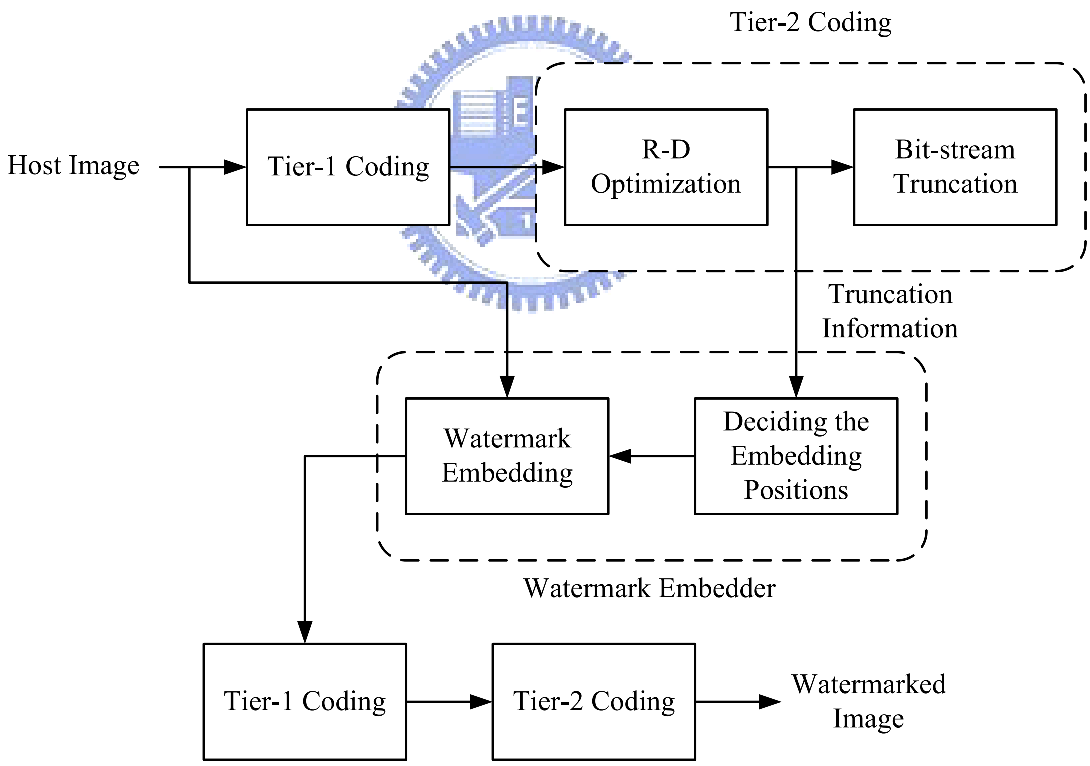


Figure 3-8 Choosing embedding positions after R-D Optimization

Based on the above discussions, the bitplane-based watermarking has the following disadvantages: high implementation complexity, low robustness (correctness), large amount of side information. High implementation complexity is caused by redoing the tier-1 and tier-2 coding. Low robustness is due to the uncertainty of data truncation, which may lead to wrong extraction results without any attack. Large amount of side information is due to the embedding positions. Because of the low correct detection rate, some capacity is sacrificed to enhance the robustness using the error correction coding (ECC). Thus, the only advantage is the simplicity of the watermark extraction, which is the original advantage before combining with the JPEG2000 standard.

For the correlation-based watermarking, the major parameter is the strength of watermarking, which decides the robustness of the watermark. And the embedding process is very simple while the extraction process is more complex. Thus, the correlation-based watermarking has the following advantages as comparing to the bitplane-based watermarking: relatively low implementation complexity, higher robustness, a small amount of side information. Since the complexity of the correlation-based watermarking is mainly due to calculating the correlation operation in the watermark extraction stage, which is much simpler than the tier-1 and tier-2 coding, the correlation-based watermarking has relatively low implementation complexity comparing to the bitplane-based watermarking. Because the truncation of coded bit-stream in the tier-2 coding does not affect the extraction results seriously, a watermark with certain intensity should provide sufficient robustness against attacks. The side information is not high since there are no embedding positions to be recorded.

Since the correlation-based watermarking seems to have more advantages than the bitplane-based watermarking for our purpose, it is thus chosen to be the watermarking method used in the protection scheme of scalable media.

Chapter 4
Design of Watermark

Since the correlation-based watermarking is chosen to be the transmission method used in layered protection of scalable media, the only thing to be done is to try our best to improve the embedding efficiency or to increase the data capacity. There are two types of scalability, resolution scalability and quality scalability (PSNR scalability). And the watermarking technique aiming at each of them will be designed separately based on their own requirements.

In this chapter, the algorithm design flow is described. Firstly, the original watermarking technique which is to be improved is introduced. Second, various methods are proposed to improve the embedding efficiency and the perceptually removable watermark is then presented. At last, the layered protection scheme is a combination of the JPEG2000 standard and the perceptually removable watermark. This scheme allows unauthorized users to preview only the coarse image by decoding the base layer.

4.1 The Original Watermarking

The perceptual watermark is proposed by M. Saito et al. [14] and is the extension of the scheme proposed by Barni et al. [24]. Since the watermarking technique proposed by Barni has been briefly introduced in section 3.1.1, here we only describe the entire watermarking scheme in sketch and point out the difference between Barni and Saito. Similar to JPEG2000, a color image with RGB components are first transformed to the YC_bC_r domain and the watermark is embedded in the Y coefficients. For easy understanding, some previous formulas and figures are repeated here.

- Watermark Embedding

As shown in Figure 4-1, the image to be watermarked is first decomposed through DWT into four levels: I_l^θ is the subband at resolution level l = 0,1,2,3 and with orientation θ= {0,1,2,3}. The watermark is made of a pseudorandom binary sequence (±1) and is embedded into the coefficients of the highest resolution I_0^θ.

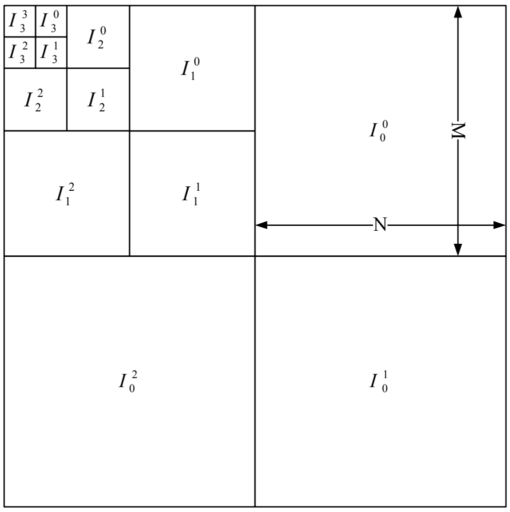


Figure 4-1 An image with four decomposition levels using the DWT

For data embedding, formula (4.1-1) maps a 1-D pseudorandom binary sequence m_h to a 2-D sequence $x^\theta(i,j)$, where M and N are the dimensions of the largest subband. Further, data of n bits is represented by $b_k \in \{1,0\}, k = 0,1,....,n-1$ and a watermarking code is generated by (4.1-3) with $p=\theta MN+iN+j$. At last, the final watermark pattern $w^\theta(i,j)$ is directly added to the host image (4.1-4), where $\tilde{I}_0^\theta(i,j)$ is the watermarked image, $I_0^\theta(i,j)$ is the host image, α is a global parameter representing the watermark strength, and $q^\theta(i,j)$ is a perceptual weighting function based on the HVS model described in section 3.1.1.

$$x^{\theta}(i,j) = m_{(\theta MN+iN+j)} \quad (4.1\text{-}1)$$

$$\sigma_k = \begin{cases} 1 & if \quad b_k = 1 \\ -1 & if \quad b_k = 0 \end{cases} \quad (4.1\text{-}2)$$

$$w^{\theta}(i,j) = x^{\theta}(i,j)\sigma_{p \bmod n} \quad (4.1\text{-}3)$$

$$\tilde{I}_0^{\theta}(i,j) = I_0^{\theta}(i,j) + \alpha q^{\theta}(i,j) w^{\theta}(i,j) \quad (4.1\text{-}4)$$

$$q_l^{\theta}(i,j) = \Theta(l,\theta)\,\Lambda(l,i,j)\Xi(l,i,j)^{0.2} \quad (4.1\text{-}5)$$

- Watermark Extraction

Watermark extraction of the embedded data is accomplished without referring to the original image. The only required information is the seed used to produce the PN sequence m_h. With the correct produced m_h, the correlation values are calculated by (4.1-6). Since the watermarking technique plays a critical role in the gradual data recovery process in our specific layered protection scheme, the extraction result saying that the image is un-watermarked is useless. Thus, the threshold used in [14] is not used and the data extraction scheme is simply expressed by (4.1-7).

$$\rho_k = \sum_{\substack{\theta,i,j \\ p \bmod n = k}} \tilde{I}_0^{\theta}(i,j)\, x^{\theta}(i,j) \quad (4.1\text{-}6)$$

$$b_k = \begin{cases} 1 & if \quad \rho_k > 1 \\ 0 & if \quad \rho_k \le 0 \end{cases} \quad (4.1\text{-}7)$$

4.2 Evolution of the Watermarking Technique

In this section, the architecture of the layered protection of scalable media is described. The watermarking technique is as a tool used for sequential data transmission. Thus, this is necessary to know the information that the watermark carries and the data relationship between layers. After the basic watermark is designed, we improve its embedding efficiency and eventually a perceptually removable watermarking technique is proposed. The proposed technique has several features different from the traditional demands for watermark but it is very useful for our specific purpose, that is, the data transmission.

4.2.1 Layered protection structure

Consider the architecture of layered protection of scalable media, which is mentioned in section 2.3 and re-shown as Figure 4-2. The watermark is used to carry the data recovery information such as the decryption key and the extraction parameter for the next layer.

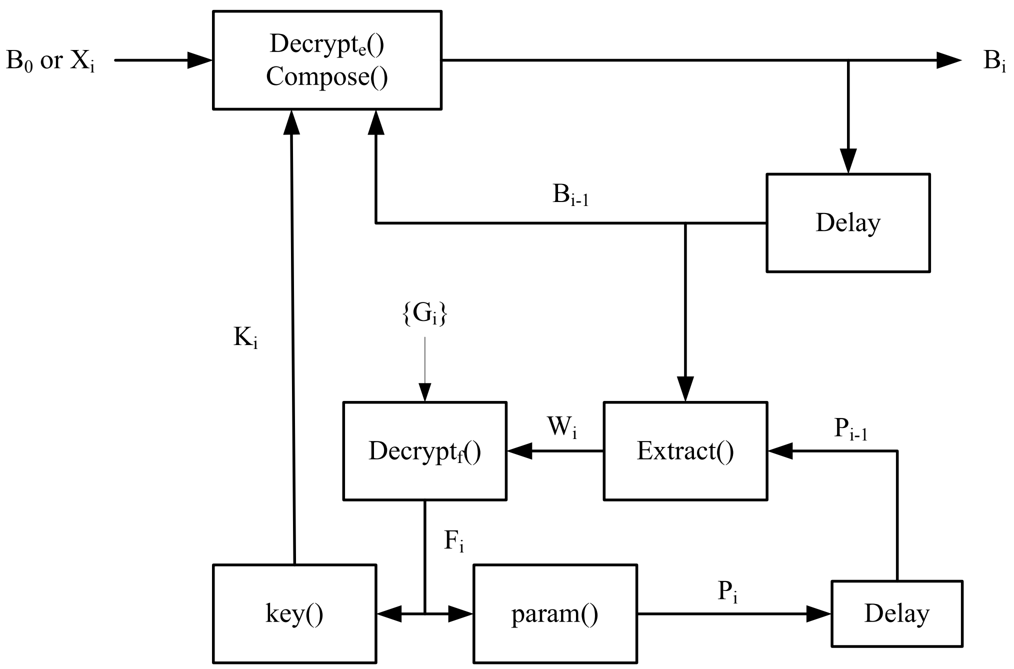


Figure 4-2 Decryption and decoding of layer-protected content

In our application, the data encryption standard (DES) is chosen to protect the context of the first enhancement layer, the advanced encryption standard (AES) is used for protecting the second enhancement layer, and the base layer serves as a preview layer without any protection. Hence, as shown in Figure 4-3, the information that should be embedded in the base layer is 96 bits. Among them, a piece of 32-bits data is the seed used to produce the pseudorandom sequence m_h and the other piece of 64-bits data is the decryption key for the first enhancement layer. Since the first enhancement layer is the last layer for data embedding, it does only contain the decryption key used for AES, which is, a 128-bits key.

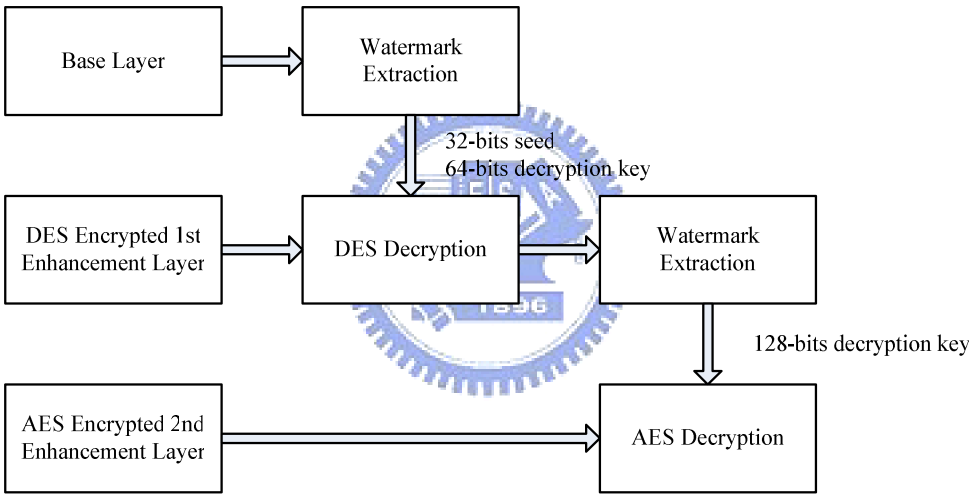


Figure 4-3 Relationship between layers

In the following sections, the resolution-based watermark used for protection of the data with resolution scalability is first discussed. After the resolution-based watermark is completely developed, it is slightly modified to form the PSNR-based watermark which is used for protecting the data with PSNR scalability.

4.2.2 The resolution-based watermarking technique

For data with resolution scalability, as shown in Figure 4-4, the base layer contains the subbands of the lowest resolutions while the enhancement layers are composed of the subbands of the higher resolutions.

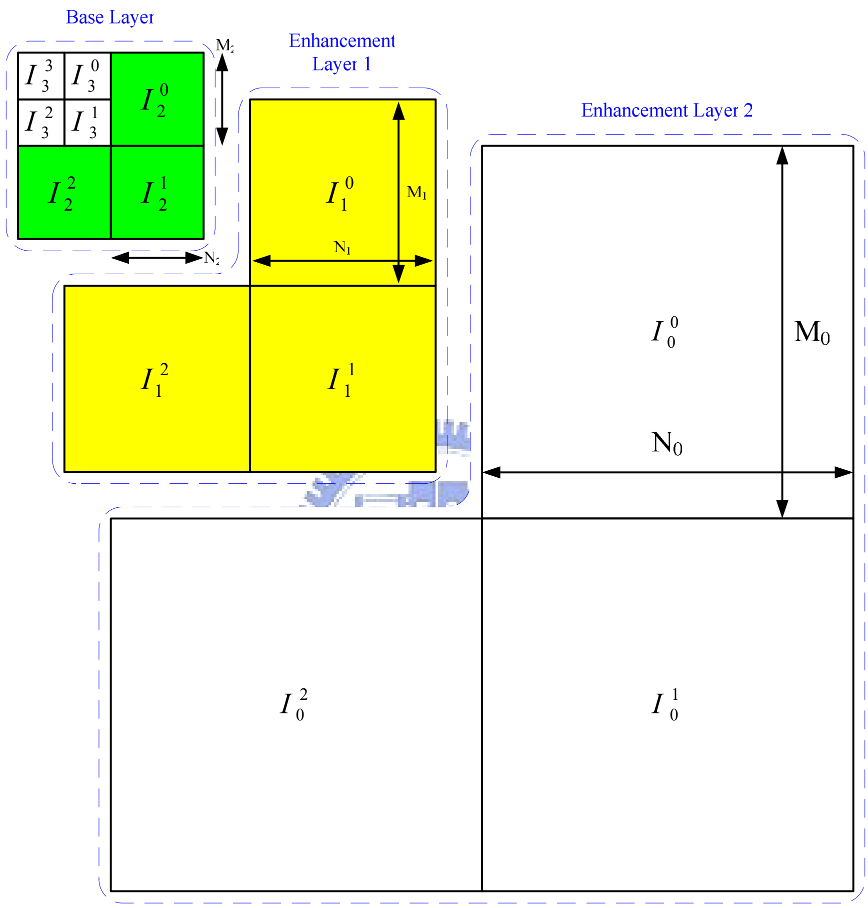


Figure 4-4 Embedding positions of the watermark

The resolution-based watermarking embeds the data of each layer into the corresponding colored area in Figure 4-4. The embedding scheme is almost the same as (4.1-1) to (4.1-5), and the only difference is that M and N are no longer the dimensions of the largest subband but the dimensions of the subband in the embedding layer. Figure 4-5 shows an example of embedding the first bit of a 6-bits data into layer k, and the colored pixels are the positions for embedding the first bit.

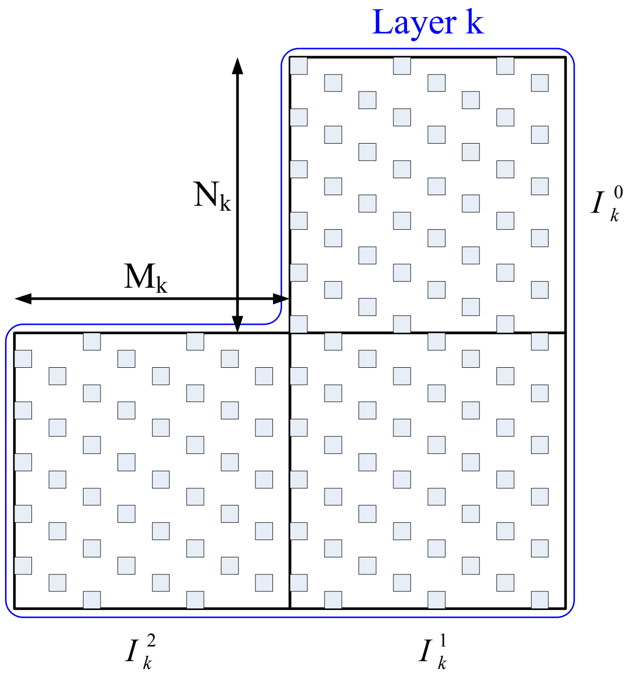


Figure 4-5 Embedding positions of the first bit of a 6-bit data

As shown in Table 3-1, we could find that the data capacity of the correlation-based watermark for a 512x512 image is low. Even though the data to be embedded into the base layer is only of 64 bits, the extracted data may not be totally correct. Therefore, our goal of embedding 96-bits data into the base layer is not possible. It is also very difficult to embed the data of 128 bits into the first enhancement layer for the same reason. All the following sub-sections describe our efforts to improve the data capacity.

4.2.2.1 Selection of the seed

Since the correctness of the extracted data depends on whether the watermark pattern is correlated to the host image, the seed used to produce the pseudorandom sequence is an important factor for improving embedding capacity of watermarking.

How to choose a seed that minimizes the correlation value between the produced pseudorandom sequence and the host image is a difficult problem. Although there are many existing methods for finding the optimum parameter to minimize a function, such as the traditional calculus-based methods (such as gradient decent) and the genetic algorithms (GA) which uses directed random searches to locate optimal solutions in complex landscapes [27], it seems none of them could solve the aforementioned problem because of the irregular cost function associated with the pseudo-random number function.

Because of the specific property of the pseudorandom function that the pseudorandom sequences produced by adjacent seeds are nearly unrelated to each other, a full search among a certain range of seeds may be the only way to identify a good seed for producing the desired watermark pattern. Thus, we set the beginning and the end seeds and examine all seeds in between to find out the best seed for producing the desired watermark pattern.

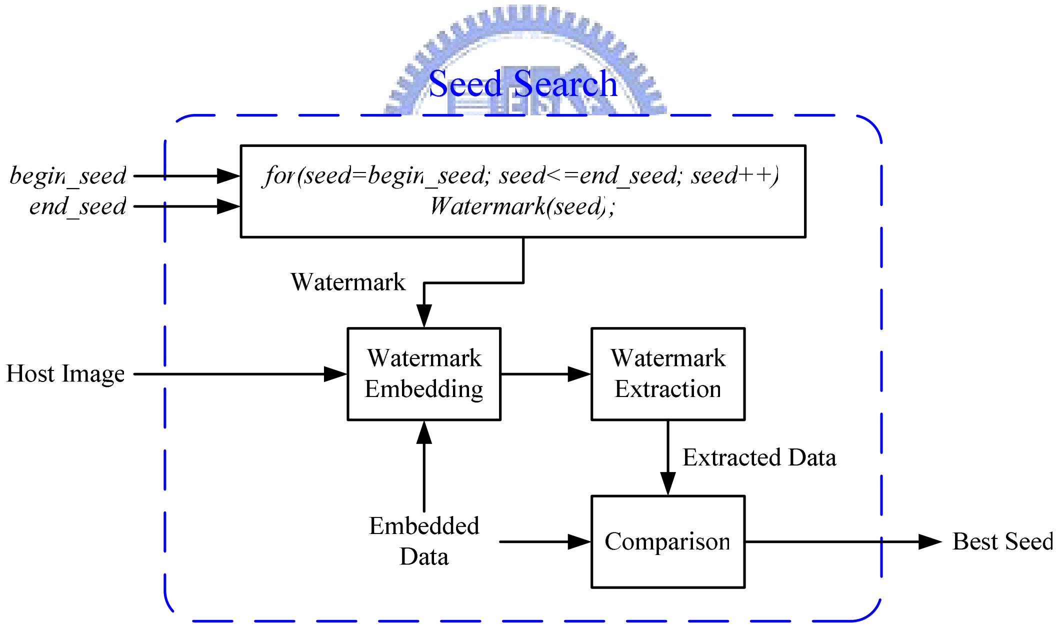


Figure 4-6 Seed search procedure

It is important to note that using the pseudorandom function to produce a watermark which is completely uncorrelated to the host image is impossible [28]. Thus, to further improve the embedding efficiency, we should take the advantage of the characteristics of the host image itself. That is, the watermark pattern does not need to be uncorrelated to the host image but it should make the extracted data as correct as possible. For this reason, we look for the seed that leads to a better extraction performance rather than the seed that produces a watermark pattern less correlated with the host image. It implies that the watermark embedding and extraction stages are pre-done to examine the correctness of extracted data as shown in Figure 4-6.

Table 4-1 shows a comparison of the embedding efficiency between the watermark embedding with and without the seed search procedure. The compression rate is of 20. It is obvious that the embedding efficiency is improved by the seed search procedure but the performance is not good enough. Therefore the other methods need to be employed to improve the watermark extraction performance.

		baboon	fruit	Lena	peppers
Before compression	BER_1	0.044062	0.007734	0.000000	0.000000
	BER_2	0.109479	0.093750	0.042500	0.021042
After compression	BER_1	0.083828	0.028672	0.000000	0.000156
	BER_2	0.135000	0.106250	0.042500	0.022187
	PSNR	25.3177	35.149	36.727	36.044
After Seed Search Procedure is applied					
Before compression	BER_1	0.000000	0.000000	0.000000	0.000000
	BER_2	0.018125	0.008229	0.000000	0.000000
After compression	BER_1	0.059766	0.009687	0.000000	0.003672
	BER_2	0.037500	0.018333	0.002917	0.002917
	PSNR	25.324	35.1643	36.6943	36.0163

Table 4-1 Results of correlation-based watermarking

4.2.2.2 The Error Correction Code (ECC)

In 1948, Shannon shows that errors in the received data induced by a noisy channel can be reduced to any desired level by a proper channel code [29]. In Shannon's channel coding theorem, it is said that every channel has a capacity C and there exist codes of rate R (R<C) that, with the maximum likelihood decoding, have an arbitrarily small decoding error probability.

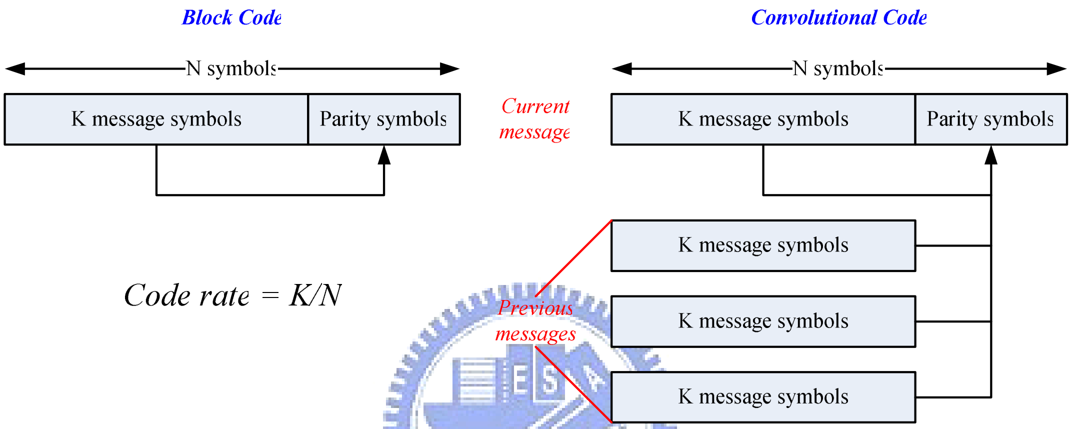


Figure 4-7 Error Correction Codes

There exist two categories of the error correction codes (ECC), the block (memory-less) codes and the convolutional (memory) codes [30]. As shown in Figure 4-7, a block code contains K message symbols and (N-K) parity symbols which are produced by the message symbols for data correction, and a convolutional code contains K message symbols and (N-K) parity symbols which are produced by the current and the previous message symbols for data correction. Using ECC to improve the data capacity of an image is inappropriate. It is because that the basic concept of the channel coding theorem is sacrificing the code rate to reduce the error probability. Since the data capacity of an image is also the maximum rate for data carrying, there is little room for carrying extra bits introduced by the channel coding.

The only use of the error correction codes (ECC) is to correct a small amount of random errors and the cost is sacrificing some data capacity.

H =

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	C_1	C_2	C_3	C_4	C_5	C_6	C_7
1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	1	0	0	0	0	1	0	0	1	0	1	0	0	0	0	0	1	1	1	0	0	0	0	0	0
2	0	0	0	0	1	0	0	1	1	1	1	1	1	1	1	1	0	0	1	0	0	1	0	0	1	0	0	0	0	1	0	0	0	1	0	0	0	0	0
3	0	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	1	1	1	1	1	1	1	1	0	0	1	1	0	1	1	0	0	0	1	0	0	0	0
4	0	0	1	0	0	0	1	0	0	0	1	0	0	1	0	1	1	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	0	0	0	1	0	0	0
5	0	1	1	0	0	1	0	1	0	1	0	0	1	0	0	1	0	0	0	0	1	1	1	1	0	1	1	0	1	0	0	0	0	0	0	0	1	0	0
6	1	0	0	0	0	1	1	0	1	0	0	0	1	1	1	0	1	1	1	1	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0
7	1	1	0	1	1	0	0	0	1	1	1	1	0	0	0	0	0	1	0	0	0	0	0	1	0	1	0	1	0	0	0	1	0	0	0	0	0	0	1
	1	1	1	1	1	1	1	1	2	2	2	2	2	2	1	2	3	3	2	1	3	2	1	3	1	4	3	3	4	2	1	1							
	6	5	4	3	2	5	4	2	6	5	4	3	5	4	2	4	4	6	3	3	5	3	3	5	2	5	4	4	5	3	3	4							
	7	7	5	7	7	6	6	5	7	7	7	7	6	6	6	5	6	7	6	6	6	5	5	7	4	7	5	7	6	4	4	7							

Figure 4-8 Parity-check matrix of the (39, 32) SEC-DED code

Figure 4-8 shows a parity-check matrix of a shortened Hamming code used to produce the parity symbols of the (39, 32) single-error-correction, double-error-detection (SEC-DED) code [26], which can correct one error bit and detect two error bits of a 39-bits data that carries 32-bit information. Since the shortened Hamming code is adopted, the data that should be embedded in the base layer becomes a 117-bits data and the first enhancement layer should contain a 156-bits encoded key.

4.2.2.3 The Perceptually Removable Watermarking (PRW)

Since the aforementioned method all failed to apparently improve the data capacity of an image, we try a different approach. Recalling that the purpose of watermarking in our applications is to transmit the encrypted data together with the decryption key, thus, the quality of the watermarked image is not important, as long as the image can be recovered at the end of the entire process. If we can perceptually remove the embedded watermark after the correct watermark is extracted, we achieve the goal of producing good quality images at the end.

Based on the idea that the quality of the watermarked image is not important, a perceptually removable watermark is proposed. In this scheme, the watermark is strengthened to increase the data capacity. Thus, the quality of the watermarked image is lower. As shown in Figure 4-9, the data to be embedded is firstly sent to the ECC Encoder to produce the shortened Hamming code for reducing the decoding errors. The seed search scheme is still used to improve the embedding capacity. Still the watermark intensity is proportional to the JND values of the host image so that the recovered image visual quality reaches its best.

The key assumption of the proposed perceptually removable watermarking method is that the JND values calculated before and after the watermark embedding are very similar. That is, if we can derive the same watermark intensity, the JND values, from either the host image or the watermarked image, we can easily remove the watermark from the watermarked image by simply subtracting the watermark of the same intensity from the watermarked image. Since the watermarked image is different from the original one, this assumption is not totally valid. A recursive method is thus proposed. As shown in Figure 4-9, the JND values are firstly calculated and used to remove the watermark. Then, the reconstructed image after the first iteration is fed back to re-calculate the JND values for the second iteration watermark removing. This process repeats until the JND values are converged to fixed values, that is, the JND values calculated in the previous and current iterations are the same. In our experience, the JND values will converge to a certain range in only two iterations. Thus, for reducing the computational complexity, only two iterations are applied to derive the desired JND values.

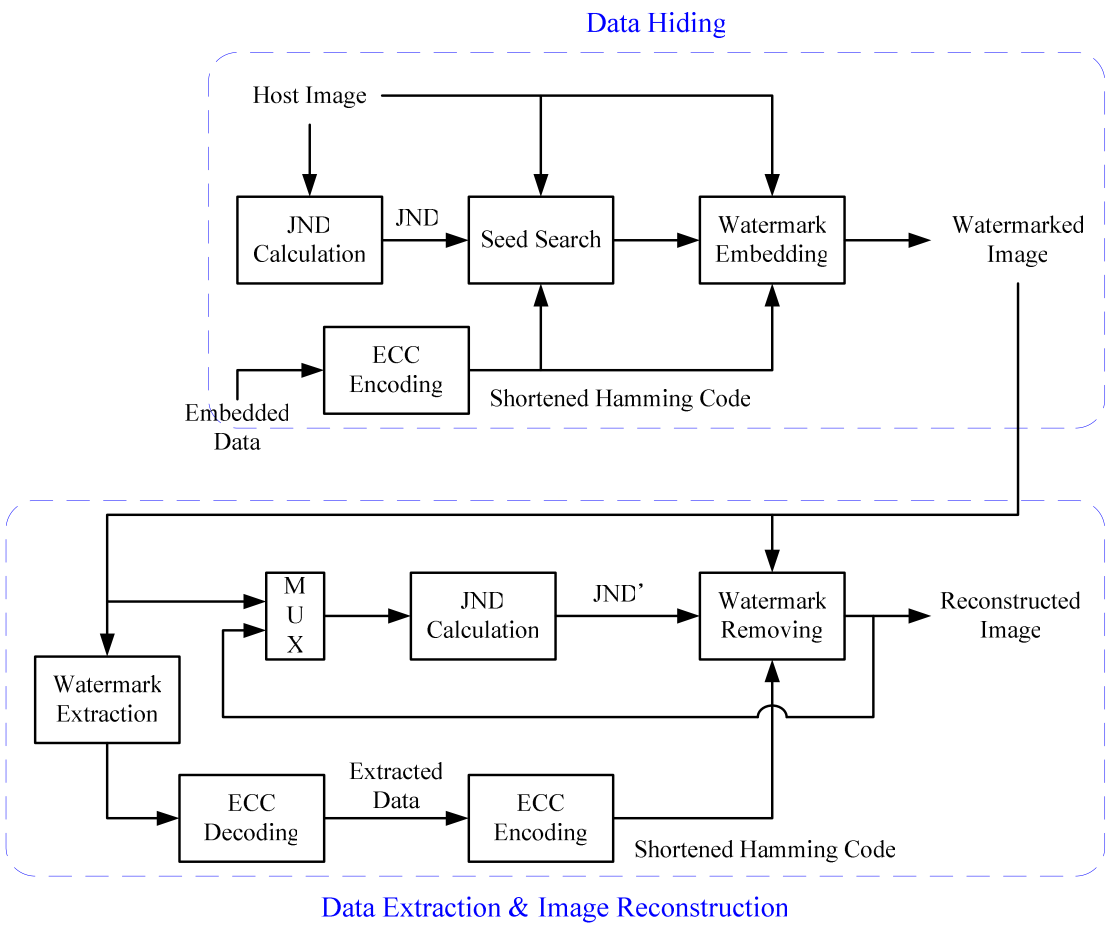


Figure 4-9 Architecture of the Perceptually Removable Watermarking: Data Hiding, Data Extraction and Image Reconstruction

Because the watermark removing process can not be perfect, the recovered image after watermark removing still contains a small amount of residual watermark. Now if the intensity of the residual watermark is proportional to the JND values, the perceptual quality affected by the residual watermark will be very small. Thus, to reduce the visibility of the residual watermark in the image, the JND values are needed for deciding the intensity of the watermark. Someone may question that why not using a fixed value to be the intensity of the watermark and then removing the watermark using the correct intensity rather than the estimated JND values. It is because the image is passed through the JPEG2000 encoder and coefficients will be quantized to achieve the objective compression rate. For this reason, the watermarked image is somewhat changed and the fixed watermark intensity is no longer meaningful. Thus, the reconstructed image still contains some residual watermark and in this case, the perceptual quality is no longer protected by the HVS model. Table 4-2 is a comparison between the fixed-value watermark intensity and the watermark intensity proportional to the JND values.

	baboon	fruit	Lena	peppers
Fixed intensity				
BER_1	0.00	0.00	0.00	0.00
BER_2	0.00	0.00	0.00	0.00
PSNR	24.97	35.45	37.85	37.18
Proportioned intensity				
BER_1	0.00	0.00	0.00	0.00
BER_2	0.00	0.00	0.00	0.00
PSNR	25.42	35.99	38.34	37.49

Table 4-2 Watermarking with fixed and proportioned intensities

One drawback of the watermark with fixed intensity is the terrible perceptual quality of the base layer, which contains the watermark and is served as the preview layer without watermark removing. It is also the main reason that we do not use the watermarking scheme with fixed watermark intensity.

4.2.3 The PSNR-based watermarking technique

Protecting the scalable media with PSNR scalability is much more difficult than protecting the scalable media with resolution scalability. It is because the coefficients of one layer in resolution-based scalable media are independent to coefficients of other layers. Thus, any change to the coefficients of one layer has no effects on the coefficients of any other layer.

For the PSNR-based scalable media, as shown in Figure 4-10, the coefficients of each layer is a collection of bits derived from the coefficients of the whole image. That is, one or several bits of a coefficient are collected to form a portion of a layer in the PSNR-based scalable media. For this reason, how to embed and extract the watermark in each layer independently is a difficult problem.

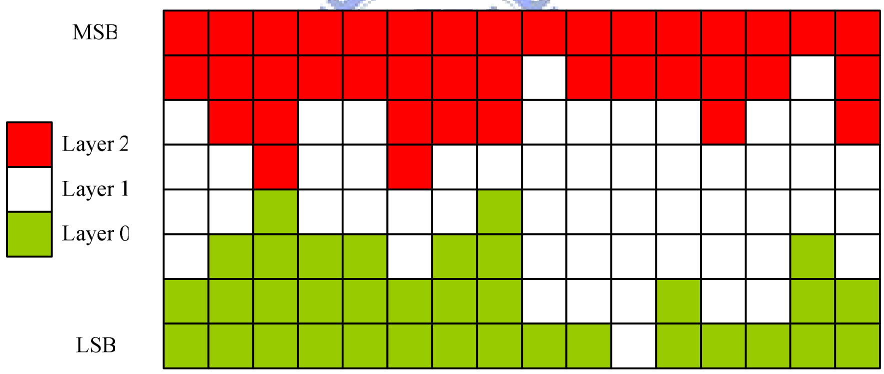


Figure 4-10 Layers of PSNR-based scalable media

Ideally, the watermarking technique for PSNR-based scalable media embeds watermarks independently into different layers and makes sure that coefficients of each layer will not overflow. But considering the JPEG2000 standard, it is impossible to know in advance which bit belongs to which layer before the entropy coding and the rate-distortion optimization are completed. Thus, to ideally embed the watermarks into different layers must pay an extra computational overload. Furthermore, since the entropy coding and the rate-distortion optimization should be redone after the watermark is embedded, a bit originally belongs to one layer may belong to another layer and the embedded data may then spread to another irrelevant layer. This will also make the watermark can not be removed effectively and possibly affect the result of watermark extraction.

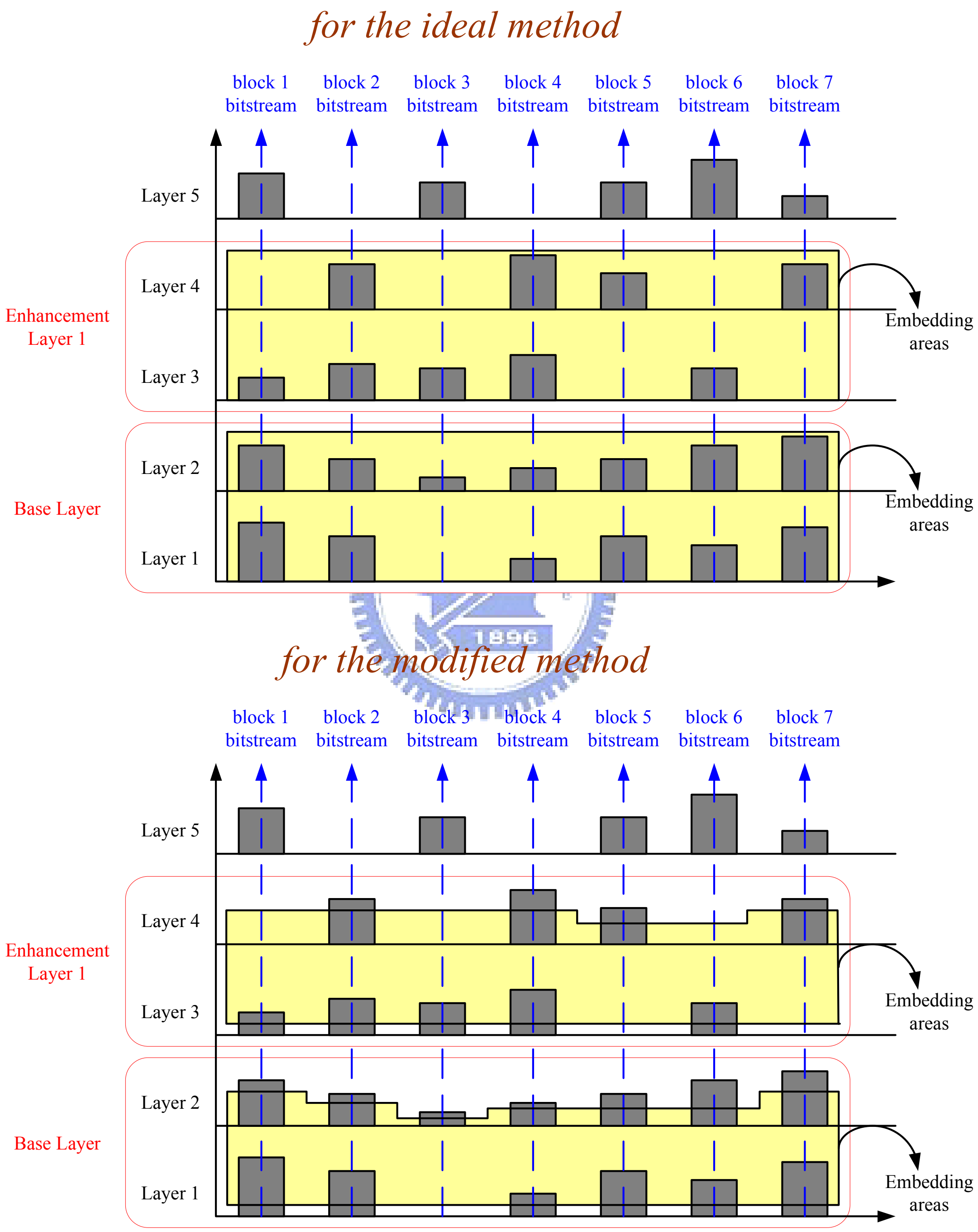


Figure 4-11 Different embedding ranges of the ideal watermarking technique for PSNR-based scalable media and the modified one

Since the ideal watermarking technique for the PSNR-based scalable media is impracticable, a modified version is proposed. In this method, the area of watermark embedding is limited in a certain range to ensure that the redoing of entropy coding and rate-distortion optimization does not spread the watermark of one layer to another irrelevant layer. Figures 4-11 shows the different embedding ranges of the ideal method and the modified one. It is a layered data with different bits in different layers, and the embedding range for the ideal method is equal to the embedding layer itself while the embedding range for the modified one is smaller than the embedding layer.

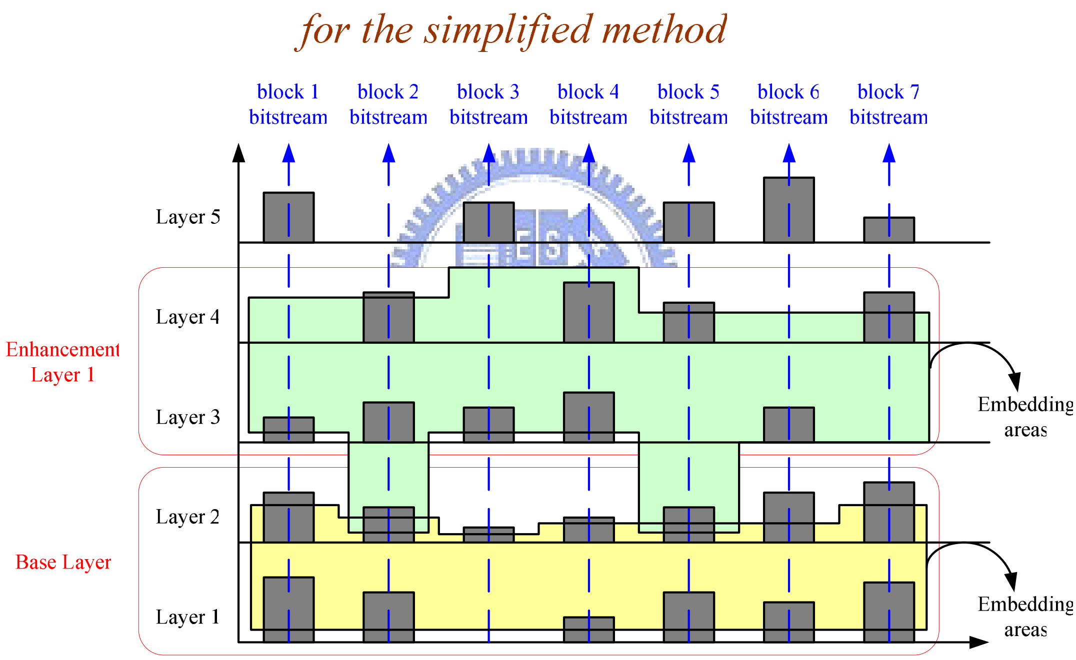


Figure 4-12 The embedding areas of the simplified watermarking technique for PSNR-based scalable media

Considering the rate-distortion optimization and the three coding passes of the JPEG2000 standard, the aforementioned method is too complicated to implement because the embedding ranges are hard to decide to ensure that the redoing of entropy coding and rate-distortion optimization will not spread the watermark of one layer to another irrelevant layer. Thus, a simplified method is proposed. In this method, we do not care about the spread of watermarks, and the only thing is to ensure that each layer contains most energy of the watermark which belongs to it. That is, some embedding bits belongs to one embedding layer are embedded into another embedding layer but most bits are embedded into the correct layer. Figure 4-12 shows an example of the embedding areas using the simplified method.

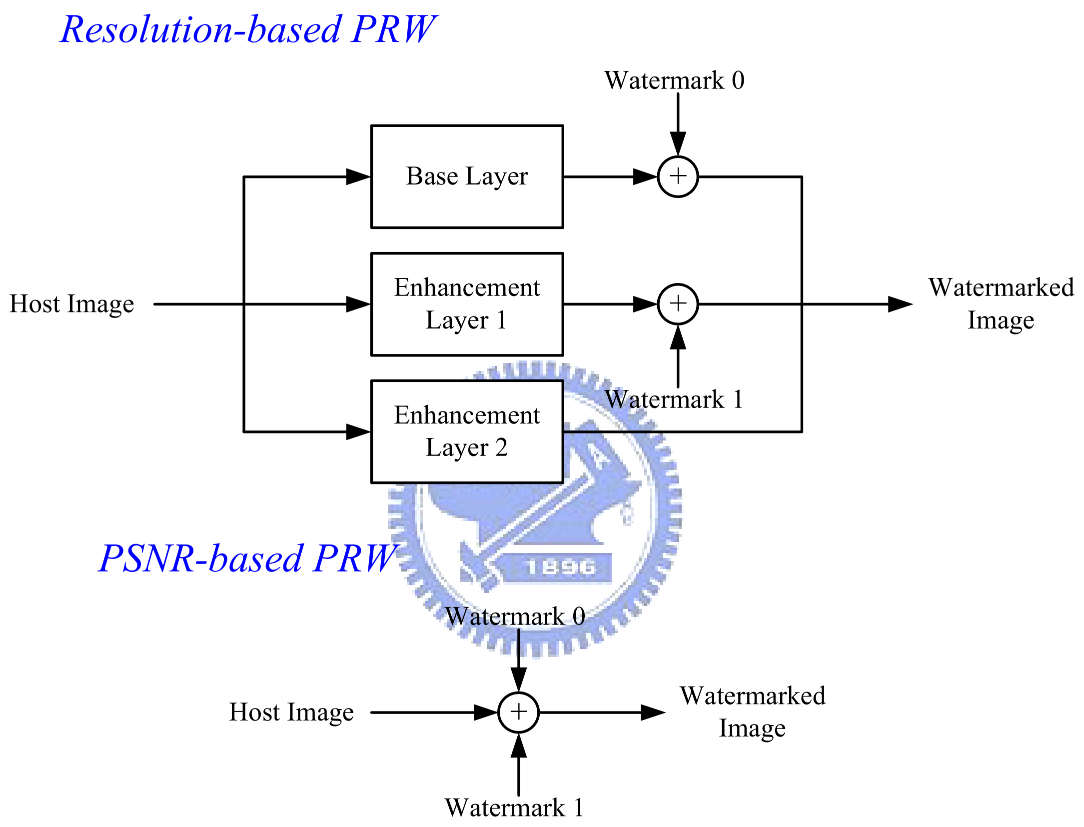


Figure 4-13 Watermark embedding of resolution-based PRW and PSNR-based PRW

Since the spread of watermark is no longer a problem, we could embed watermark into each layer using different intensities and the perceptually removable watermarking technique can be employed again. As shown in Figure 4-13, the only difference between the perceptually removable watermarking (PRW) techniques for the resolution-based and the PSNR-based scalable media is that the PRW for the PSNR-based scalable media repeats to embed different watermarks with different intensities into the same coefficients. And the HVS used for the PSNR-based watermarking is slightly changed for simplicity, that is, the $\Xi(l,i,j)^{0.2}$ in (3.1.1-6) is simplified to (4.2.3-1).

$$\Xi(l,i,j) = Var\left\{ I_3^3(1+y+\frac{i}{2^{3-l}}, 1+x+\frac{j}{2^{3-l}} \right\}_{\substack{x=0,1\\y=0,1}} \qquad (4.2.3\text{-}1)$$

Since the watermark of one layer will spread to other layers, the watermark removing scheme of the PSNR-base PRW is not the same with the resolution-based PRW as shown in Figure 4-14. We should note that in the PSNR-based PRW, the removing action is redone when a new layer is received for quality improving. It is because the watermark belonging to the previous layer cannot be removed completely at the watermark removing stage of the previous layer due to the spread of watermark. Thus, we should try to remove the watermark of the previous layer again when more data is received to improve the efficiency of watermark removing.

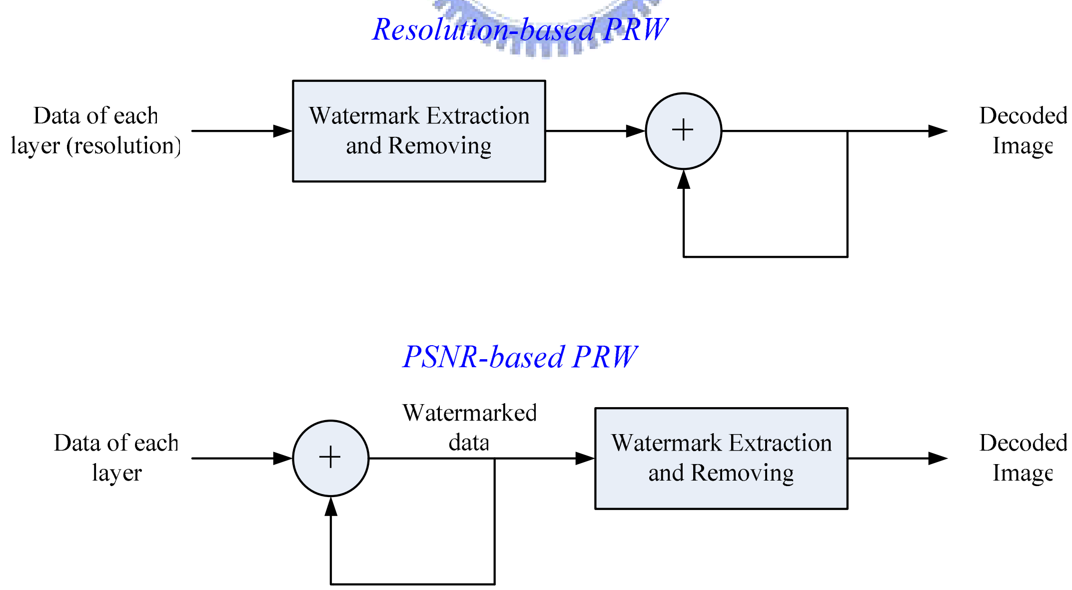


Figure 4-14 Watermark Extraction and Removal of Resolution-based and PSNR-based PRW

4.3 Combination of the Layered Protection Scheme with the JPEG2000 Standard

JPEG2000 standard provides an optional file format that applications may choose to use to hold the JPEG2000 compressed image data as shown in Figure 4-15 [1]. We could use this file format to implement our layered protection scheme of scalable media.

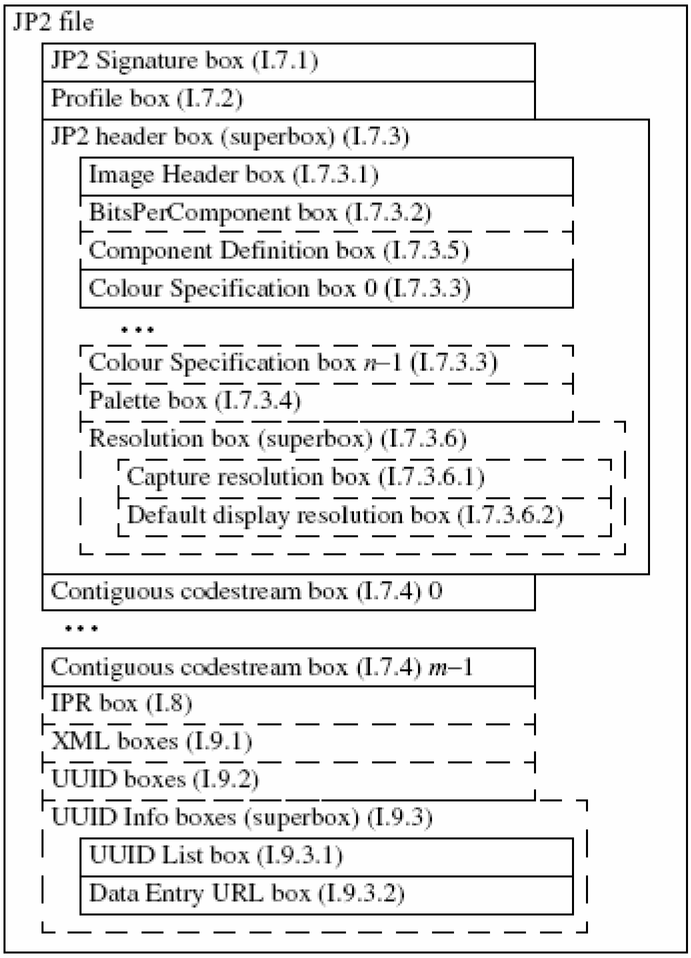


Figure 4-15 Conceptual structure of a JP2 file

Conceptually, the JP2 file format encapsulates the JPEG2000 codestream along with other core pieces of information about that codestream. The building block of the JP2 format is called a box. All data are encapsulated in boxes, and the definition of each specific box type defines the kinds of data that may be found within a box of that type. Some boxes will be defined to contain other boxes.

In Figure 4-15, boxes with dashed borders are optional in conforming the JP2 format files. It is important to note that for the contiguous codestream boxes, when displaying an image, a conforming reader should ignore all codestreams after the first codestream is found in the file. Thus, we could use this feature to encapsulate the un-encrypted base layer into the first contiguous codestream box to serve as the preview layer and the remained two encrypted enhancement layers are put into the following contiguous codestream boxes to be decrypted later.

Another important feature of the JP2 file format is that a box which carries the application specific data (metadata) is ignored because it is an unknown box type to an unauthorized reader, and we could use this feature to tell an authorized reader that this JP2 file is a layered protected image and could be decoded as images with various qualities based on different levels of authority while an unauthorized reader only gets the preview image.

Chapter 5
Simulation Results

In this chapter, all simulation results will be shown and discussed. All the simulations are done under the following conditions. The layers of the resolution-based watermarking are divided as shown in Figure 4-4 with a compression ratio of 20 and the layers of the PSNR-based watermarking are divided as (1) the base, (2) the 1^{st} enhancement and (3) the 2^{nd} enhancement layers and every layer has the same amount of data, and the overall compression ratio is of 20. As discussed in section 4.2.2.2, there are 117 bits embedded in the base layer and 156 bits embedded in the 1^{st} enhancement layer. For the resolution-based watermark and the PSNR-based watermark, the global parameter α, which is shown in (4.1-4), for the base layer is about 3 to 4 times greater than the global parameter α for the 1^{st} enhancement layer.

All colored images used in simulation are shown in Figure 5-1, and from left to right, top to bottom, they are Airplane, baboon, earth, flower, fruit, lena, m22401, nuclei, peppers, peppers2, Sailboat, stone, sun, Tiffany and BMW. All images' sizes are of 512x512 except that the BMW image is of 768x512.

In Tables 5-1 to 5-4, the "ideal PSNR" represents the PSNR value between the originally uncompressed image and the compressed image using the JPEG2000 encoder with a ratio of 20, and the "PSNR" represents the PSNR value between the originally uncompressed image and the fully reconstructed images with watermark embedding and extraction. The BER_ecd_base and BER_ecd_enh1 represent the bit error rate (BER) between the originally embedded and the extracted shortened Hamming codes in the base and the 1^{st} enhancement layers, while BER_base and

BER_enh1 are the BER between the extracted message, which is corrected by the shortened Hamming code, and the originally embedded message in the base and the 1st enhancement layers. Looking at tables 5-1 to 5-4, we could find that some random errors occurs and are corrected by the shortened Hamming code, and this is the purpose of using the shortened Hamming code.

Figure 5-1 Images used for simulation

	Airplane	*baboon*	*BMW*	*earth*
BER_ecd_enh1	0.000000	0.009487	0.000000	0.000000
BER_ecd_base	0.000000	0.000085	0.000000	0.000000
BER_enh1	0.000000	0.000000	0.000000	0.000000
BER_base	0.000000	0.000000	0.000000	0.000000
PSNR	36.12	25.39	34.95	29.21
Ideal PSNR	36.43	25.44	35.50	29.35
	flower	*fruit*	*lena*	*m22401*
BER_ecd_enh1	0.000000	0.000000	0.000000	0.000000
BER_ecd_base	0.000000	0.000000	0.001282	0.000000
BER_enh1	0.000000	0.000000	0.000000	0.000000
BER_base	0.000000	0.000000	0.000000	0.000000
PSNR	36.13	35.83	37.93	29.97
Ideal PSNR	36.38	36.17	38.41	30.06
	nuclei	*peppers2*	*peppers*	*Sailboat*
BER_ecd_enh1	0.000000	0.000000	0.000000	0.000000
BER_ecd_base	0.000000	0.000000	0.000000	0.000000
BER_enh1	0.000000	0.000000	0.000000	0.000000
BER_base	0.000000	0.000000	0.000000	0.000000
PSNR	29.77	40.14	37.29	29.61
Ideal PSNR	30.05	40.82	37.86	29.82
	stone	*sun*	*Tiffany*	
BER_ecd_enh1	0.000000	0.000000	0.000000	
BER_ecd_base	0.000427	0.000000	0.000000	
BER_enh1	0.000000	0.000000	0.000000	
BER_base	0.000000	0.000000	0.000000	
PSNR	28.49	36.53	34.10	
Ideal PSNR	28.66	36.62	34.35	

Table 5-1 Results of resolution-based PRW using 5_3 filter in JPEG2000

	Airplane	*baboon*	*BMW*	*earth*
BER_ecd_enh1	0.000000	0.000256	0.000000	0.000321
BER_ecd_base	0.000000	0.000000	0.000000	0.000000
BER_enh1	0.000000	0.000000	0.000000	0.000000
BER_base	0.000000	0.000000	0.000000	0.000000
PSNR	36.38	25.42	34.92	29.11
Ideal PSNR	37.13	25.57	36.10	29.63
	flower	*fruit*	*lena*	*m22401*
BER_ecd_enh1	0.000000	0.000000	0.000000	0.000000
BER_ecd_base	0.000000	0.000000	0.000000	0.000000
BER_enh1	0.000000	0.000000	0.000000	0.000000
BER_base	0.000000	0.000000	0.000000	0.000000
PSNR	36.54	35.99	38.34	30.08
Ideal PSNR	37.34	36.75	39.67	30.34
	nuclei	*peppers2*	*peppers*	*Sailboat*
BER_ecd_enh1	0.000000	0.000000	0.000000	0.000000
BER_ecd_base	0.000000	0.000000	0.000000	0.000000
BER_enh1	0.000000	0.000000	0.000000	0.000000
BER_base	0.000000	0.000000	0.000000	0.000000
PSNR	30.13	39.89	37.499	29.65
Ideal PSNR	30.78	42.01	38.69	30.10
	stone	*sun*	*Tiffany*	
BER_ecd_enh1	0.000000	0.000000	0.000000	
BER_ecd_base	0.000000	0.000000	0.000000	
BER_enh1	0.000000	0.000000	0.000000	
BER_base	0.000000	0.000000	0.000000	
PSNR	28.64	37.08	34.32	
Ideal PSNR	29.02	37.64	34.98	

Table 5-2 Results of resolution-based PRW using 9_7filter in JPEG2000

	Airplane	*baboon*	*BMW*	*earth*
BER_ecd_enh1	0.000000	0.000000	0.000000	0.000833
BER_ecd_base	0.000000	0.004103	0.000000	0.000000
BER_enh1	0.000000	0.000000	0.000000	0.000000
BER_base	0.000000	0.000000	0.000000	0.000000
PSNR	35.39	25.25	34.34	28.91
Ideal PSNR	36.43	25.44	35.50	29.35
	flower	*fruit*	*lena*	*m22401*
BER_ecd_enh1	0.000000	0.000000	0.000000	0.000000
BER_ecd_base	0.000000	0.000000	0.000000	0.000000
BER_enh1	0.000000	0.000000	0.000000	0.000000
BER_base	0.000000	0.000000	0.000000	0.000000
PSNR	35.69	35.23	37.20	29.81
Ideal PSNR	36.38	36.17	38.41	30.06
	nuclei	*peppers2*	*peppers*	*Sailboat*
BER_ecd_enh1	0.000000	0.000000	0.000000	0.000000
BER_ecd_base	0.009316	0.000000	0.000000	0.000000
BER_enh1	0.000000	0.000000	0.000000	0.000000
BER_base	0.000000	0.000000	0.000000	0.000000
PSNR	29.71	38.93	36.69	29.38
Ideal PSNR	30.05	40.82	37.86	29.82
	stone	*sun*	*Tiffany*	
BER_ecd_enh1	0.001923	0.000000	0.000000	
BER_ecd_base	0.004359	0.000000	0.000000	
BER_enh1	0.000000	0.000000	0.000000	
BER_base	0.000000	0.000000	0.000000	
PSNR	28.39	35.63	33.45	
Ideal PSNR	28.66	36.62	34.35	

Table 5-3 Results of PSNR-based PRW using 5_3 filter in JPEG2000

	Airplane	*baboon*	*BMW*	*earth*
BER_ecd_enh1	0.000000	0.003397	0.000000	0.000000
BER_ecd_base	0.000000	0.001368	0.000000	0.000000
BER_enh1	0.000000	0.000000	0.000000	0.000000
BER_base	0.000000	0.000000	0.000000	0.000000
PSNR	35.93	25.40	34.66	29.00
Ideal PSNR	37.13	25.57	36.10	29.63
	flower	*fruit*	*lena*	*m22401*
BER_ecd_enh1	0.000000	0.000000	0.000000	0.000000
BER_ecd_base	0.000000	0.000000	0.000000	0.000000
BER_enh1	0.000000	0.000000	0.000000	0.000000
BER_base	0.000000	0.000000	0.000000	0.000000
PSNR	36.45	35.70	38.09	30.01
Ideal PSNR	37.34	36.75	39.67	30.34
	nuclei	*peppers2*	*peppers*	*Sailboat*
BER_ecd_enh1	0.000000	0.000000	0.000000	0.000000
BER_ecd_base	0.004188	0.000000	0.000000	0.000000
BER_enh1	0.000000	0.000000	0.000000	0.000000
BER_base	0.000000	0.000000	0.000000	0.000000
PSNR	30.06	40.02	37.31	29.52
Ideal PSNR	30.78	42.01	38.69	30.10
	stone	*sun*	*Tiffany*	
BER_ecd_enh1	0.000000	0.000000	0.000000	
BER_ecd_base	0.003248	0.000000	0.000000	
BER_enh1	0.000000	0.000000	0.000000	
BER_base	0.000000	0.000000	0.000000	
PSNR	28.66	36.53	33.91	
Ideal PSNR	29.02	37.64	34.98	

Table 5-4 Results of PSNR-based PRW using 9_7filter in JPEG2000

In Tables 5-1 to 5-4, we can find that some of the extracted data is not correct but can be corrected by the shortened Hamming Code. And this is the original purpose that we combine the ECC with the perceptually removable watermark.

	Airplane	*baboon*	*BMW*	*earth*
res. based 5_3	41.13	32.54	38.18	33.46
res. based 9_7	39.96	32.22	37.33	32.81
PSNR-based 5_3	38.33	31.04	36.41	32.09
PSNR-based 9_7	37.77	30.86	35.99	31.70
	flower	*fruit*	*lena*	*m22401*
res. based 5_3	41.42	40.59	41.62	36.71
res. based 9_7	41.06	39.12	40.78	35.54
PSNR-based 5_3	39.23	38.16	39.47	35.35
PSNR-based 9_7	40.04	37.48	39.29	33.94
	nuclei	*peppers2*	*peppers*	*Sailboat*
res. based 5_3	34.48	43.40	41.01	36.16
res. based 9_7	33.53	42.21	40.50	34.93
PSNR-based 5_3	34.32	40.44	39.16	34.77
PSNR-based 9_7	33.40	41.81	39.03	33.68
	stone	*sun*	*Tiffany*	
res. based 5_3	32.38	43.71	40.12	
res. based 9_7	31.89	42.08	39.22	
PSNR-based 5_3	31.67	39.34	36.99	
PSNR-based 9_7	31.09	38.89	36.62	

Table 5-5 PSNR shown for the compressed and the reconstructed images

In Tables 5-1 to 5-4, the quality of the reconstructed image with 9_7 filters seems to have a better performance, but the 9_7 filter also produces a better PSNR when used in the JPEG2000 standard alone. Thus, Table 5-5 provides a comparison between the compressed and the reconstructed images. We could find that in either the resolution-based watermarking or the PSNR-based watermarking, the 5_3 filter leads to a better reconstructing performance. This is because in JPEG2000 standard, compression scheme using the 5_3 filter will bypass the quantization step as introduced in section 2.1.4, and the quantization step may reduce the accuracy of the recalculated JND values in the image reconstruction stage. Thus, bypassing the quantization step will make a better performance is what we expect. And only comparing Table 5-1 and Table 5-2 with Table 5-3 and Table 5-4, we can find that the PSNR-based watermarking has a worse performance; this is because the watermark is incompletely removed due to the characteristics of PSNR-based watermarking and the stronger watermark intensity which is necessary to protect the extracted information in the base layer.

	Airplane	*baboon*	*BMW*	*earth*
res. based 5_3	58.053802	59.987682	57.544605	58.980923
PSNR-based 5_3	∞	∞	∞	∞
	flower	*fruit*	*lena*	*M22401*
res. based 5_3	56.687248	57.961796	57.069214	59.131187
PSNR-based 5_3	∞	∞	∞	∞
	nuclei	*peppers2*	*peppers*	*Sailboat*
res. based 5_3	57.189980	56.536079	56.601677	56.913784
PSNR-based 5_3	∞	∞	∞	∞
	stone	*sun*	*Tiffany*	
res. based 5_3	59.337577	57.913830	57.939541	
PSNR-based 5_3	∞	∞	∞	

Table 5-6 PSNR shown for the uncompressed and the uncompressed reconstructed images

Table 5-6 is a comparison between the original the uncompressed image and the uncompressed reconstructed images, which eliminates the effects of compression and purely displays the performance of the perceptually removable watermarking (PRW). We should note that the PSNR between the host image and the reconstructed image using the PSNR-based watermarking is infinite; this is due to the simplified HVS model, that is, the coefficients used to recalculate the JND values are not changed by the embedded watermark and then the recalculated JND values are the same as the original ones, which makes the image be perfectly reconstructed.

Figures 5-2 to 5-5 shows the reconstructed images at different reconstruction levels. The first image is the base layer image without watermark removing. It is the preview layer. The next two images are images with the 1st enhancement layer and with all enhancement layers and the watermarks are removed. We can see that the picture quality of the preview layer is quite different between the resolution-based and the PSNR-based scalable media. This is due to the effect of watermarks in the PSNR-based watermarked image cannot be masked by the low resolution of the image. Since the difference between the host image and the fully reconstructed image is not visible, Figures 5-6 to 5-9 are the 10-times amplified absolute differences, and from left to right, top to bottom, are Airplane, baboon, earth, flower, fruit, lena, m22401, nuclei, peppers, peppers2, Sailboat, stone, sun, Tiffany and BMW. We can find that the differences concentrate on the edge of the images, where the human eyes are less sensitive to the noise. This implies that the residual watermarks are masked by the HVS model. Also, a more textured image has a less reconstructed image quality. It is because a more textured image will lose more information through the compression process, and this will also make the recalculated JND values be more different from the original ones. Since the quality of the watermark removal process depends on whether the recalculated JND values are the same as the original ones, a more textured image will have a less reconstructed image quality.

Finally, we conclude that the resolution-based watermark has a better reconstructed image quality. Using the 5_3 filter also makes the watermark removal process perform better, but using the 9_7 filter derives a better reconstructed image.

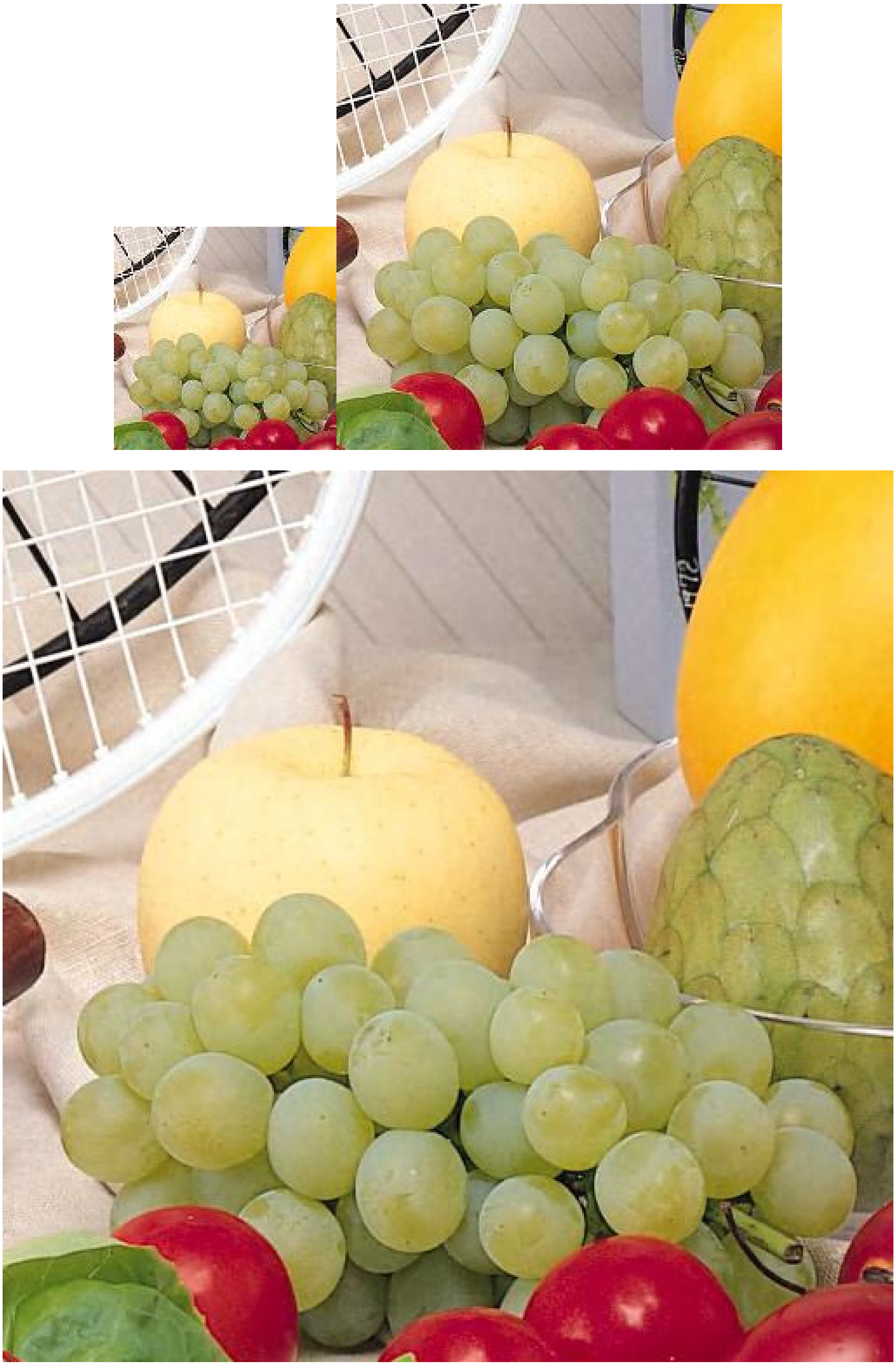


Figure 5-2 The decoded images of the resolution-based scalable media at different reconstruction levels with 5_3 filter

Figure 5-3 The decoded images of the resolution-based scalable media at different reconstruction levels with 9_7 filter

Figure 5-4 The decoded images of the PSNR-based scalable media at different reconstruction levels with 5_3 filter

Figure 5-5 The decoded images of the PSNR-based scalable media at different reconstruction levels with 9_7 filter

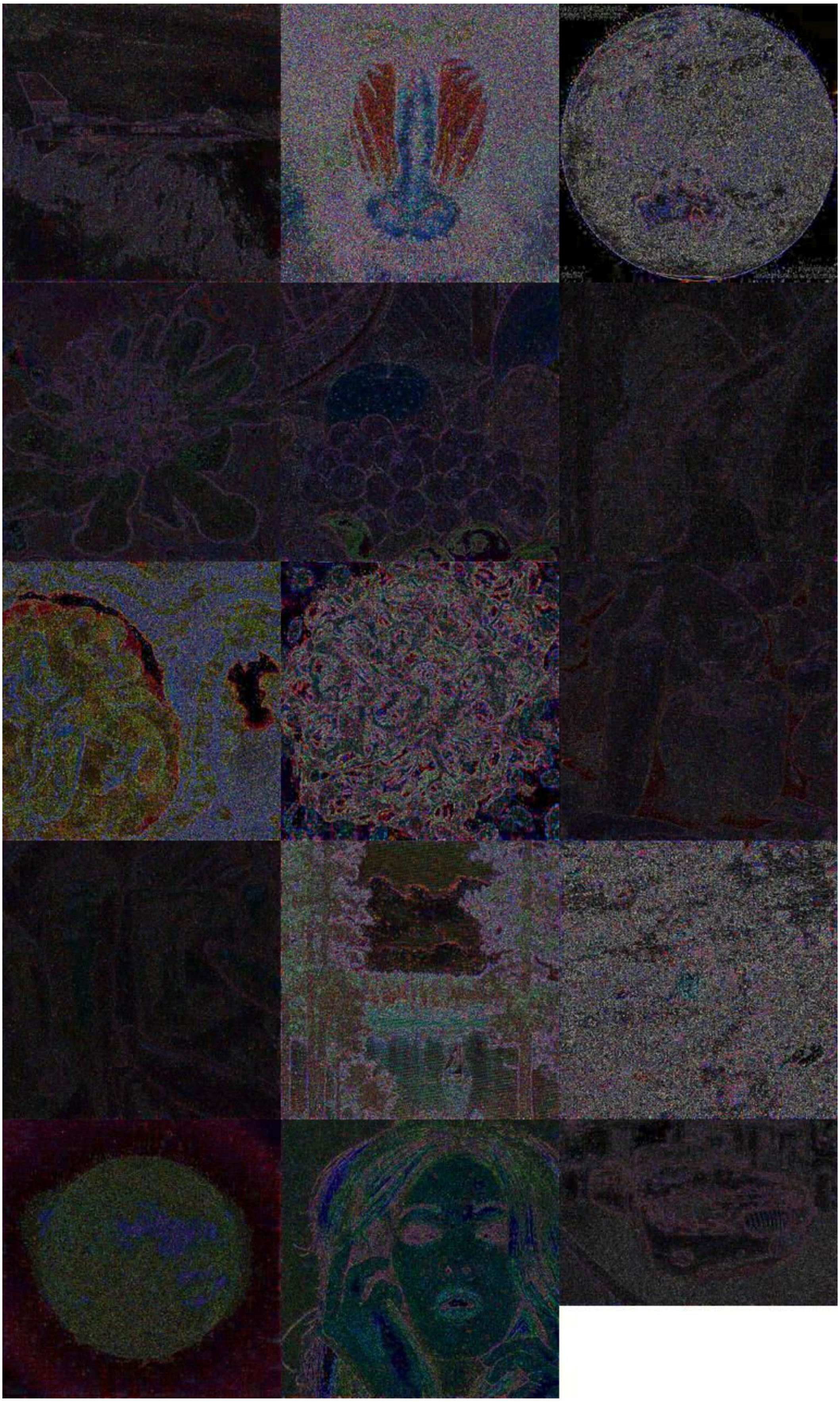


Figure 5-6 Amplified absolute difference of res.-based watermarking with 5_3 filter

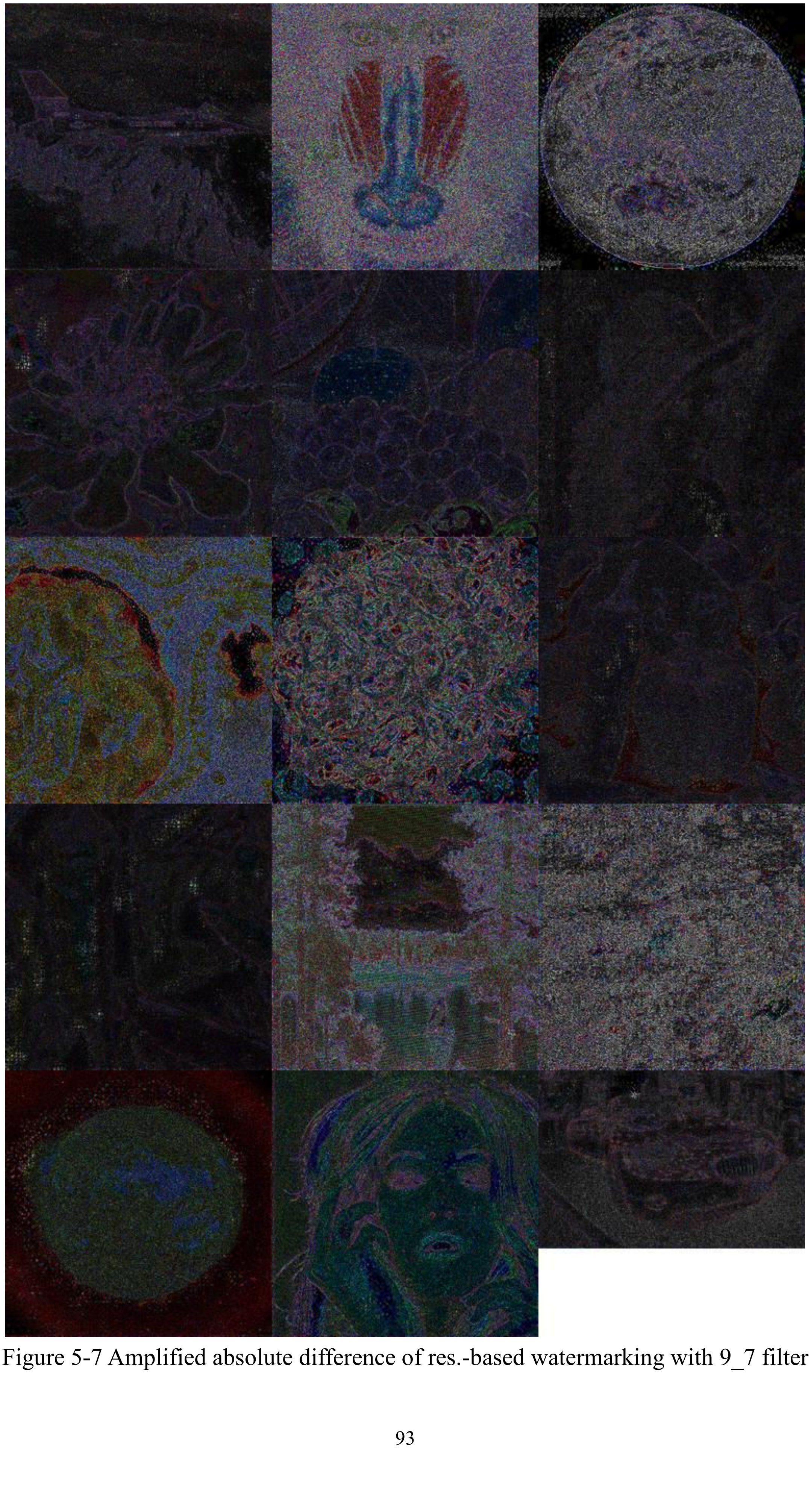


Figure 5-7 Amplified absolute difference of res.-based watermarking with 9_7 filter

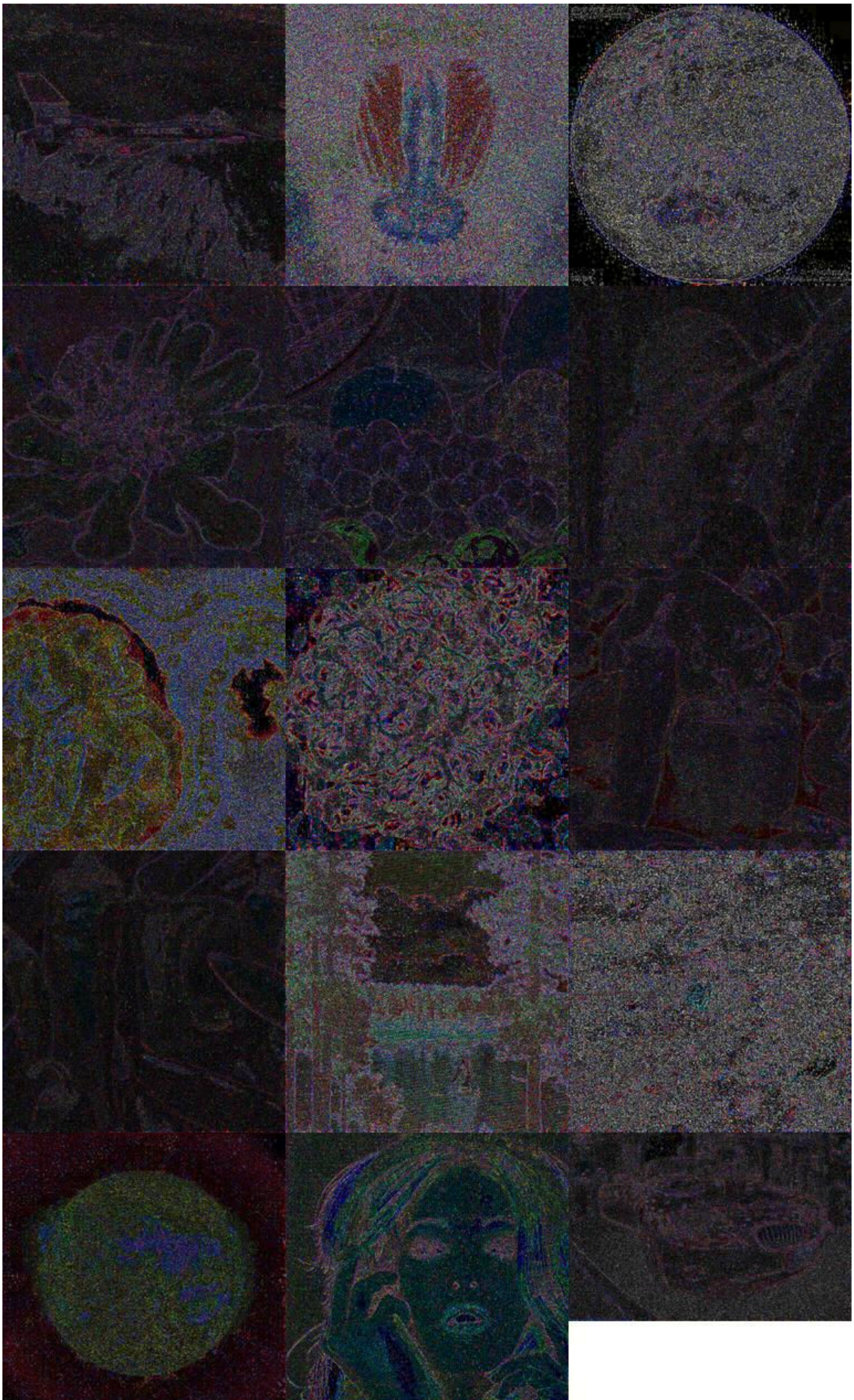


Figure 5-8 Amplified absolute difference of PSNR-based watermarking with 5_3 filter

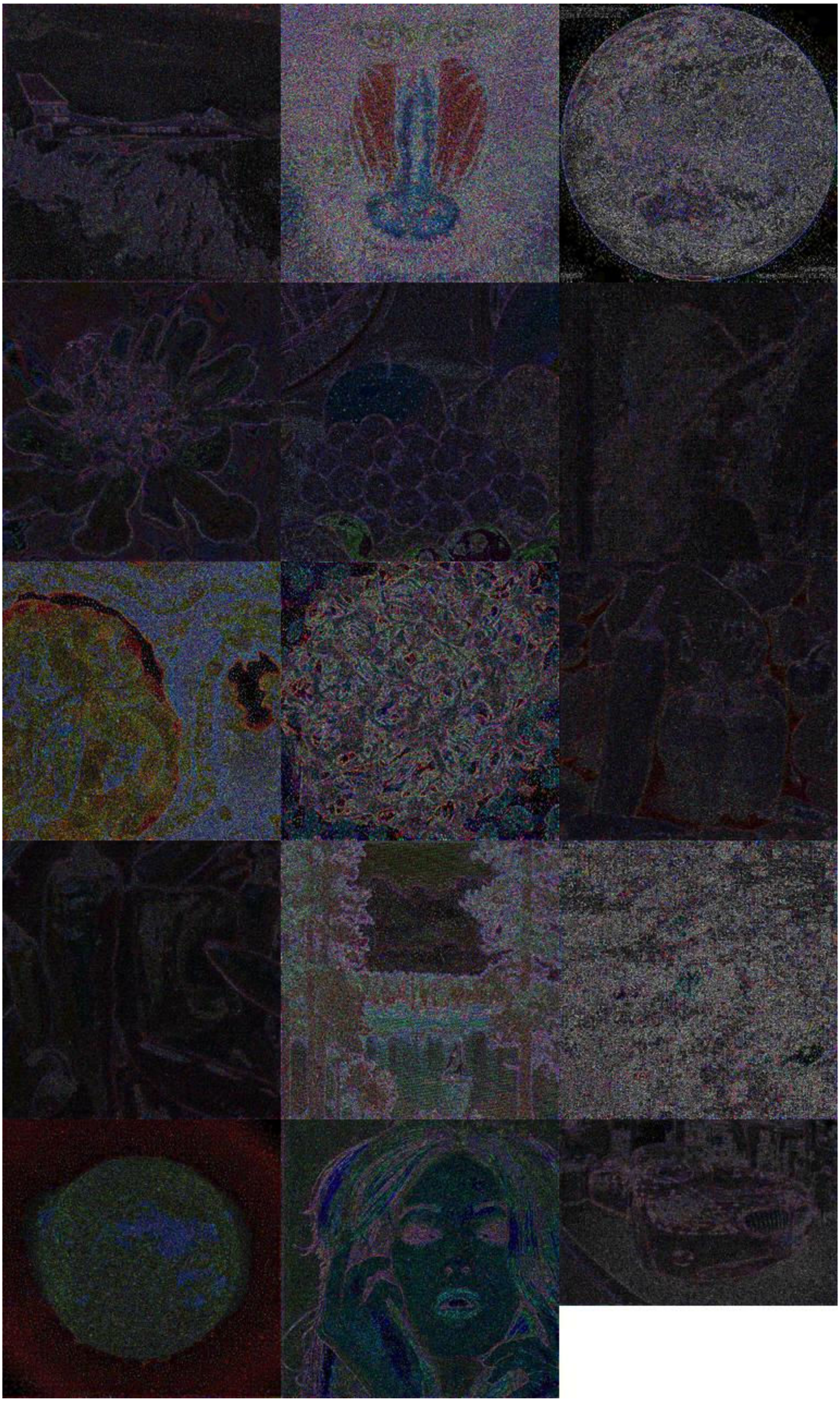


Figure 5-9 Amplified absolute difference of PSNR-based watermarking with 9_7 filter

Chapter 6 Conclusions and Future Work

Traditionally, the digital watermarking technique is used to protect the intellectual property of digital contents. It hides the information into the host data without perceptual distortion and the watermark should be robust: that is, the watermark should be very difficult to destroy.

In this thesis, the watermark is used to transmit data, which is different from the traditional use of watermarks. Thus, we propose a watermarking technique, the perceptually removable watermarking (PRW), to implement the layered protection scheme on scalable media. It is a special watermarking technique that is very different from the traditional watermarking techniques because the watermarked image is very noisy. But with large capacity, it fits well to our layered protection scheme since good quality pictures can be reconstructed at the end after the watermark is removed. Simulation results verify that the layered protection scheme is well implemented with an acceptable preview image and almost unaffected reconstructed images.

Another contribution of this work is that the layered protection scheme is successfully combined with the JPEG2000 standard. An unauthorized user treats the layered protected data as an ordinary JPEG2000 file and can only decode a low quality preview image. But the authorized user based on his/her given level of authority can retrieve images with better quality.

Since the video contents such as movies often have high commercial values, how to efficiently protect them can be more important than protecting the still images. Thus, a possible future research topic is to protect the video contents using an extension of the techniques presented in this paper, the layered protection scheme with perceptually removable watermarking.

References

[1] JPEG2000 Part I Final Committee Draft Version 1.0 (ISO/IEC FCD15444-1) ISO/IEC JTC1/SC29/WG1 March 2000

[2] C. Christopoulos, A. Skodras, and T. Ebrahimi, "The JPEG2000 Still Image Coding System: An Overview," IEEE Transactions on Consumer Electronics, Vol. 46, No. 4, pp.1103-1127, Nov 2000

[3] M.J. Nadenau and J. Reichel, "Opponent Color, Human Vision and Wavelets for Image Compression," in Proceeding of 7th Color Image Conference, Scottsdale, AZ, pp.237-242, 16-19 Nov 1999

[4] D. Taubman, E.O.M.W.G. Seroussi and I.U.F. Ono, "Embedded Block Coding in JPEG2000," in Proceeding of IEEE International Conference on Image Processing, Vol. 2, pp.33-36, Vancouver, Canada, Sept 2000

[5] J.L. Mitchell and W.B. Pennebaker, "Software Implementation of the Q-coder," IBM J. of Res. Develop, Vol. 32, No. 6, pp.753-774, Nov 1988

[6] C.D. Vleeschouwer, J.F. Delaigle, and B. Macq, "Invisibility and Application Functionalities in Perceptual Watermarking – An Overview," in Proceeding of IEEE, Vol.90, No. 1, Jan 2002

[7] W.A. Wan Adnan, S. Hitam, S. Abdul-Karim, and M.R. Tamjis, "A Review of Image Watermarking," in Proceeding of Student Conference on Research and Development, pp.381-384, Aug 2003

[8] S.H. Kwok, C.C. Yang, and K.Y. Tam, "Watermark Design Pattern for International Property Protection in Electronic Commerce Applications," in Proceeding of the 33rd Annual Hawaii International Conference on System Science, Vol. 2, Jan 2000

[9] C.I. Podilchuk and E.J. Delp, "Digital Watermarking: Algorithms and Applications," IEEE Signal Processing Magazine, Vol. 18, Issue 4, pp.33-46, July 2001

[10] G.C. Langelaar, I. Setyawan, and R.L. Lagendijk, "Watermarking Digital Image and Video Data. A state-of-the-Art Overview," IEEE Signal Processing Magazine, Vol. 17, Issue 5, pp.20-46, Sept 2000

[11] S. Burgett, E. Koch, and J. Zhao, "Copyright Labeling of Digitized Image Data," IEEE Communication Magazine, pp.94-100, Mar 1998

[12] D.N. Vizireanu and R.O. Preda, "A New Digital Watermarking Scheme for Image Copyright Protection using Wavelet Packets," in Telecommunications in Modern Satellite, Cable, and Broadcasting Services, 2005 7th International Conference, Vol. 2, pp.518-521, Sept 2005

[13] Z. Guannan, W. Shuxun, and W. Quan, "An Adaptive Block-based Blind Watermarking Algorithm," in Proceeding of ICSP'04 7th International Conference, Vol. 3, 2004

[14] M. Saito, M. Kino, and S. Wada, "HVS Based Wavelet Watermarking with Bit Data Embedded in Color Images," in Proceeding of ICCS 8th International Conference, Vol. 1, pp.485-488, Nov 2002

[15] T.S. Chen, J. Chen, and J.G. Chen, "A Simple and Efficient Watermarking Technique Base on JPEG2000 Codec," in Proceeding of the IEEE 5th International Symposium on Multimedia Software Engineering, 2003

[16] B.A. Wandell, *Foundations of Vision Sunderland*, MA: Sinuaer, 1995

[17] L.A. Olzak and J.P. Thomas, *Handbook of Perception and Human Performance: Seeing Spatial Patterns*, New York: Wiley, Vol.1, Ch. 7, pp.1-56, 1986

[18] S. Newhall, "Preliminary report of the O.S.A. subcommittee on the spacing of the Munsell colors," J. Opt. Soc. Amer., Vol. 30, 70, 617, 1940

[19] T.N. Cornsweet, *Visual Perception*. New York: Academic, 1970

[20] J.L. Mannos and D.J. Sakrism, "The effects of a visual fidelity criterion on the

ending of images," IEEE Trans. Inform. Theory, Vol. IT-20, pp.525-536, July 1974

[21] S.J. Wee and J.G. Apostolopoulos, "Secure scalable streaming enabling transcoding without decryption," IEEE Int. Conf. Image. Proc., pp437-440, 2001

[22] F.C. Chang, H.C. Huang, and H.M. Hang, "Layered Access Control Schemes on Watermarked Scalable Media," IEEE International Symposium on Circuits and Systems, Vol. 5, pp.4983-4986, May 2005

[23] G. O'Driscoll, *The Essential Guide to Digital Set-top Boxes and Interactive TV*, Prentice Hall, 2000

[24] M. Barni, F. Bartolini, and A. Piva, "Improved Wavelet-Based Watermarking Through Pixel-Wise Masking," IEEE Transaction Image Processing, Vol. 10, No. 5, May 2001

[25] A.S. Lewis and G. Knowles, "Image compression using the 2-D wavelet transform," IEEE Trans. Image Processing, Vol. 1, pp.244-250, Apr 1992

[26] M.Y. Hsiao, "A class of Optimal Minimum Odd-weight-column SEC-DED Codes," July 1970

[27] M. Srinivas and L.M. Patnaik, "Genetic Algorithms: A Survey," Digital Object Identifier, Vol. 27, Issue, pp.17-26, June 1994

[28] J. Cox, M.L. Miller, and J.A. Bloom, *Digital Watermarking*, Academic 2002

[29] C.E. Shannon, "A Mathematical Theory of Communication," The Bell System Technical Jornal, Vol.27, pp.379-423, 623-656, July, October 1948

[30] S.B. Wicker, *Error Control Systems for Digital Communication and Storage*, PrenticHall, 1995