A DWT Based
Perceptual Video Coding Framework
—— Concepts, Issues and Techniques

A thesis submitted in fulfillment of the requirements
for the degree of Doctor of Philosophy

Liming Mei

B. Eng.

School of Electrical and Computer Engineering
College of Science, Engineering and Technology
RMIT University
September 2008
DECLARATION

I, Liming Mei, hereby certify that except where due acknowledgement has been made, the work is that of the author alone; the work has not been submitted previously, in whole or in part, to qualify for any other academic award; the content of the thesis is the result of work which has been carried out since the official commencement date of the approved research program; any editorial work, paid or unpaid, carried out by a third party is acknowledged; and, ethics procedures and guidelines have been followed.

Liming Mei
September 2008
To my Parents,
you did everything for your child.
You taught me to be courageous and persistent.

To my parents-in-law,
you are always supportive.

To my dear wife Jingyi,
I love you with my heart and soul.
You hold my hand firmly in raining days.

To my son Ruiyang,
You bring your dad so much happiness everyday.
It is wonderful to observe and accompany your growth.

To everyone who helps and supports me,
I thank you for your company along this journey.
Table of Contents

CHAPTER 1  INTRODUCTION ........................................................................................................ 1
1.1 OVERVIEW .......................................................................................................................... 1
1.2 VIDEO COMPRESSION STANDARD ACTIVITIES ......................................................... 2
1.3 RESEARCH DIRECTION ................................................................................................. 4
1.4 CONTRIBUTIONS ............................................................................................................. 5
1.5 THESIS LAYOUT ............................................................................................................. 6

CHAPTER 2  FOUNDATION OF DWT VIDEO CODING ............................................... 9
2.1 GENERAL STRUCTURE FOR TRADITIONAL VIDEO CODING .................................. 9
   2.1.1 Predictive, Transform and Subband Coding ...................................................... 10
   2.1.2 Quantization ........................................................................................................ 11
   2.1.3 Entropy and Entropy Coding ............................................................................. 12
   2.1.4 Context Design — Root in Generic Entropy Coding ......................................... 14
   2.1.5 MQ Coder — An Adaptive Binary Arithmetic Coder ....................................... 19
2.2 DISCRETE WAVELET TRANSFORM (DWT) .............................................................. 20
   2.2.1 Traditional DWT ............................................................................................. 20
   2.2.2 Lifting Implementation of DWT ....................................................................... 25
   2.2.3 Adaptive Directional Lifting DWT .................................................................. 27
2.3 MOTION COMPENSATION IN DWT DOMAIN ......................................................... 30
   2.3.1 Low-Band-Shift Motion Estimation ................................................................. 33
2.4 BITPLANE BASED IMAGE/VIDEO CODING ............................................................. 40
CHAPTER 6  VISION MODELING FOR DWT VIDEO CODING ......................... 102

6.1  VISION, MODELING AND APPLICATION ............................................... 102
6.2  APPLICATION OF PDM TO INTRA-CODER ............................................. 110
6.3  VISIBILITY TO QUANTIZATION ERRORS IN DWT DOMAIN ................... 116
6.4  SUMMARY ................................................................................................. 125

CHAPTER 7  CONCLUSIONS ........................................................................... 126

A. CONTEXT DESIGN OF EBCOT ................................................................. 131

B. DWT VIDEO CODERS—A REVIEW .......................................................... 136

C. INTRODUCTION TO UEGK BINARIZATION ........................................ 157

BIBIOGRAPHY .............................................................................................. 160
Abstract

The efficient digital representation of image and video signals has been the subject of considerable research over the past years. Various digital video coding techniques have been developed. They are targeted for a wide range of applications, such as Video on Demand (VOD), Digital Television (DTV) / High Definition Television (HDTV) / Ultra High Definition Television (UHDTV) broadcasting and multimedia image/video database services. Some transmission media such as Cable Modem, xDSL, or UMTS require much lower data rates than those of broadcast channels. The enhanced coding efficiency can enable the transmission of more video channels or higher quality video representations within existing digital transmission capacities.

Recent years have witnessed the success of the Discrete Cosine Transform (DCT) based video compressions, where the techniques are standardized by the Moving Picture Experts Group (MPEG) and the ITU-T Video Coding Experts Group (VCEG). On the other hand, the Discrete Wavelet Transform (DWT) based techniques have been extensively investigated and the Embedded Block Coding with Optimized Truncation (EBCOT) has been adopted by JPEG2000 standard. It is therefore natural to explore the application of DWT techniques in video coders. It is also desirable to incorporate the human perception into video compression systems, regardless whether they are based on the DCT or the DWT, because the traditional quantitative quality measures used in video compression systems have proven to be inconsistent with the human perception.
The work in this thesis explores the DWT based video coding by introduction of a novel DWT (Discrete Wavelet Transform) / MC (Motion Compensation) / DPCM (Differential Pulse Code Modulation) video coding framework, which adopts the EBCOT as the coding engine for both the intra- and the inter-frame coder. The adaptive switching coding is investigated for this framework. The Low-Band-Shift (LBS) is employed for the MC in the DWT domain. The LBS based MC is proven to provide consistent improvement in terms of the Peak Signal-to-Noise Ratio (PSNR) over the simple Wavelet Tree (WT) based MC. The Adaptive Arithmetic Coding (AAC) is adopted to code the motion information. The context set of the Adaptive Binary Arithmetic Coding (ABAC) for the inter-frame data is redesigned based on the statistical analysis. To further improve the perceived picture quality, a Perceptual Distortion Measure (PDM) based on human vision model is used for the EBCOT of the intra-frame coder. A visibility assessment of the quantization error of various subbands in the DWT domain is performed through subjective tests. In summary, all these findings have addressed the issues originated from the proposed perceptual video coding framework. They include: a working DWT/MC/DPCM video coding framework with superior coding efficiency for head-shoulder sequences or those with translational motions; an adaptive switching coding based on this framework; an effective LBS based MC scheme in the DWT domain; a methodology of the context design for entropy coding of the inter-frame data; a PDM which replaces the MSE inside the EBCOT coding engine for the intra-frame coder, which provides improvement in terms of the perceived quality of intra-frames; a visibility assessment to the quantization errors in the DWT domain.
Acknowledgements

To my mother and father for bringing me to the world and doing everything for me. Thank you for your love and efforts all the time. To my wife Jingyi, thank you for your love, support and understanding along this long journey. Thank you for sacrificing yourself to the housework and the child care, so I can have time to do the research. To my dear son Ruiyang, thank you for bringing so much happiness to your dad. To my aunt Huifen, who looked after me in the first year during my candidature, thank you for the gourmet you made in that year.

My sincerest thanks go to my main supervisor, Professor Henry Wu. Thank you for your trust, patience, guidance and encouragement in the past years. You opened a new door for me, which has brought so much fun and surprise. Thank you for telling me not to give up whenever I felt tired. Special Thanks go to my co-supervisor and good friend, Dr. Damian Tan, for the numerous invaluable discussions in the past years. Thank you for sharing your knowledge and skills with me. I really appreciate your driving to my home during the weekends to discuss the research work. Thanks also go to my fellow PhD student in RMIT, Mr. Jacob Yang, for the information about H.264 parameter configurations.

Thanks go to the School of Computer Science and Software Engineering of Monash University for providing the school scholarship in 2001. Special thanks go to the Australia Research Council (ARC) for providing the APAI scholarship under a grant
number LP0219691 between 2002 and 2005. Thanks go to the School of Electrical and Computer Engineering of RMIT University for providing the school scholarship in 2005.

My thanks also go to Mr. Paul Arden, the managing director in mental images Pty Ltd. Thank you for allowing me to come late in Tuesday mornings so I can have research meetings in RMIT University. Thank you for your kind understanding of the hardship of doing part-time PhD. Special thanks go to my colleague in mental images, Mr. Sam Noonan, for his review on the text writing in part of this thesis.

Finally, I wish to thank all my friends and relatives for their helps and supports during the whole candidature period.

Liming Mei

RMIT University

September 2008
List of Publications


List of Common Abbreviations

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-D</td>
<td>One-dimensional</td>
</tr>
<tr>
<td>2-D</td>
<td>Two-dimensional</td>
</tr>
<tr>
<td>3-D</td>
<td>Three-dimensional</td>
</tr>
<tr>
<td>bpp</td>
<td>Bits-per-pixel</td>
</tr>
<tr>
<td>DCT</td>
<td>Discrete Cosine Transform</td>
</tr>
<tr>
<td>DWT</td>
<td>Discrete Wavelet Transform</td>
</tr>
<tr>
<td>ME</td>
<td>Motion Estimation</td>
</tr>
<tr>
<td>MP</td>
<td>Motion Prediction</td>
</tr>
<tr>
<td>MC</td>
<td>Motion Compensation</td>
</tr>
<tr>
<td>DPCM</td>
<td>Differential Pulse Code Modulation</td>
</tr>
<tr>
<td>EBCOT</td>
<td>Embedded Block Coding with Optimized Truncation</td>
</tr>
<tr>
<td>EZW</td>
<td>Embedded Zero-tree Wavelet</td>
</tr>
<tr>
<td>SPIHT</td>
<td>Set Partitioning in Hierarchical Trees</td>
</tr>
<tr>
<td>HVS</td>
<td>Human Visual System</td>
</tr>
<tr>
<td>MSE</td>
<td>Mean Squared Error</td>
</tr>
<tr>
<td>PSNR</td>
<td>Peak Signal-to-Noise Ratio</td>
</tr>
<tr>
<td>LBS</td>
<td>Low-Band-Shift</td>
</tr>
<tr>
<td>ADL</td>
<td>Adaptive Directional Lifting</td>
</tr>
<tr>
<td>MRMC</td>
<td>Multi-resolution Motion Compensation</td>
</tr>
<tr>
<td>WTMC</td>
<td>Wavelet-tree based MC</td>
</tr>
<tr>
<td>VLSI</td>
<td>Very Large Scale Integration</td>
</tr>
<tr>
<td>BAC</td>
<td>Binary Arithmetic Coding</td>
</tr>
<tr>
<td>ABAC</td>
<td>Adaptive Binary Arithmetic Coding</td>
</tr>
<tr>
<td>MCBABAC</td>
<td>Multi-Context Based Adaptive Binary Arithmetic Coding</td>
</tr>
<tr>
<td>Abbreviation</td>
<td>Description</td>
</tr>
<tr>
<td>--------------</td>
<td>------------------------------------</td>
</tr>
<tr>
<td>PC</td>
<td>Predictive Coding</td>
</tr>
<tr>
<td>TC</td>
<td>Transform Coding</td>
</tr>
<tr>
<td>SBC</td>
<td>Subband Coding</td>
</tr>
<tr>
<td>MPEG</td>
<td>Moving Picture Experts Group</td>
</tr>
<tr>
<td>VCEG</td>
<td>Video Coding Experts Group</td>
</tr>
<tr>
<td>QMF</td>
<td>Quadrature Mirror Filters</td>
</tr>
<tr>
<td>WTMC</td>
<td>Wavelet-Tree based MC</td>
</tr>
</tbody>
</table>
Chapter 1

Introduction

1.1 Overview

A picture is worth a thousand words. This proverb reflects how influential the visual content is to our experience to the world. Nevertheless, the widespread use of the digitized visual content nowadays is inevitably accompanied by the image/video compression due to the limit of digital storage and transmission bandwidth. One of the primary challenges for a digital video compression system is how to remove the statistical and psychovisual redundancies in the videos with minimum impact to the perceptual quality (A.B.Watson 1990; B.Girod 1993; R.J.Clarke 1995; K.R.Rao and J.J.Hwang 1996; S.Winkler 1999; H.R.Wu and K.R.Rao 2005). To answer this question, one needs to understand how the video compression works. A typical digital video compression system is composed of functional modules, such as signal transformation, Motion Prediction (MP) and Motion Compensation (MC), quantization and entropy coding, to name a few (K.R.Rao and J.J.Hwang 1996). A simplified video compression system is illustrated in Fig. 1.1 where both the preprocessing and postprocessing modules are not included. The combination of these functional modules is the evolutionary result of Predictive Coding (P.Elias 1955), Transform Coding (R.J.Clarke 1985), and/or more recent Subband Coding (J.W.Woods 1991). In a digital video compression system, the
above modules/ functional blocks are working together to achieve the rate
driven/dependent coding or the quality driven/dependent coding. The former coding
approach aims at the best video quality within the given bitrate budget, while the latter
aims at the minimum bitrate at the allowable quality. An example of such a video
compression system is the widely used DCT (Discrete Cosine Transform) / MC / DPCM
(Differential Pulse Code Modulation) hybrid coding scheme (K.R.Rao and J.J.Hwang
1996) which has been adopted by both the ITU-T and the MPEG standards (ISO/IEC
2005).

1.2 Video Compression Standard Activities

It is impossible to discuss the video compression techniques without having a glance at
the activities of video compression standards. The main organizations carrying out
internal and industry standardization activities in video compression field are the Moving
Picture Experts Group (MPEG), which is a working group of ISO/IEC, and the ITU-T
Video Coding Experts Group (VCEG). H.261 was the first practical digital video coding
standard. This design was a pioneering effort, and all subsequent international video
coding standards have been based closely on its design. MPEG-1 was the earliest
standard developed by MPEG for video and audio compression. It was later used as the
standard for Video CD (VCD). H.262 is identical in content to the video part of MPEG-2, which is the video and audio compression standard for broadcast-quality television. It has been used for over-the-air digital television, digital satellite TV services, digital cable television signals, and DVD. H.263 was developed as an evolitional improvement based on H.261, MPEG-1 and MPEG-2 standards. Its first version was published in 1995 then followed with the so-called H.263+ and H.263++, which provided enhancements of capabilities. MPEG-4 supports the coding of video/audio objects, 3D content, and low bitrate encoding. It is worth mentioning that the DWT technique has been adopted in the visual texture coding (VTC) mode of MPEG-4 (I. Moccagatta, M. Z. Coban et al. 1999). This coding mode performs the still texture coding to facilitate texture mapping on 2-D or 3-D meshes. It supports the following functionalities: 1) efficient compression; 2) coding of arbitrarily shaped objects; 3) spatial and quality scalability; and 4) error robustness in error-prone environment. In MPEG-4 VTC, EZW is used for encoding of the wavelet coefficients in the AC subbands. MPEG-4: Part 10 and ITU-T H.264 are the resulting joint work between the VCEG and the MPEG. As has been the case with past standards, its design provides the most current balance between the coding efficiency, implementation complexity and cost, based on the state of Very Large Scale Integration (VLSI) design technology. These standards represent the current main stream of the video compression techniques formally adopted by the industry. They all adopt a DCT/MC/DPCM coding structure except MPEG-4 VTC, which is quite successful in achieving high compression efficiency. Nevertheless, the shortcomings are also obvious. The so-called blocking artifact is an inherited problem especially under low bitrate (M. Yuen and H. R. Wu 1998) despite that techniques such as looping filtering are used to
improve the subjective quality. Scalability is another issue to be addressed. Applications such as the multimedia image/video browsing require scalability of the image/video bitstream in respect to resolution, quality, and temporal domain. Scalable bitstream can adaptively suit different decoding platforms. However, the current video compression standards only support limited capability of “scalability”. The third concern is the adaptation to the Human Vision System (HVS). It has been widely known that a statistically well performed video coder does not necessarily obtain the appreciation of HVS (B.Girod 1993). The human viewer is always the final judge for the quality of multimedia applications. The above issues, existing in the current main stream video compression techniques, are being addressed by researchers using different approaches.

1.3 Research Direction

The rapid development of the digital computer’s computation power has facilitated the efforts of numerous researchers for video compression in two directions. The first research direction adopts the traditional statistical approach. For example, the Multi-Context Based Adaptive Binary Arithmetic Coding (MCBABAC) in the H.264/AVC video compression standard (D.Marpe, H.Schwarz et al. 2003) uses 399 contexts based on arithmetic coding engine. H.264 also makes good use of existing DCT techniques but applies the DCT down to 4x4 pixel block. Flexible motion compensation (MC) matching blocks are suggested as well for higher compression efficiency. Recently, the DWT based image/video coding techniques have attracted widespread attention of the literature, which will be discussed in Chapter 2. The second research direction employs different aspects of the Human Visual System (HVS), or even a vision model, to predict the
responses of human viewers to the digital visual content. This finely tunes the existing
digital image/video compression techniques toward the perceptual quality criterion
instead of the traditional statistical quality criterion such as the Mean Squared Error
(MSE) or the Peak Signal-to-Noise Ratio (PSNR) by incorporating the emulated response
of the HVS into the coding process (H.R.Wu and K.R.Rao 2005). The work presented in
this thesis reflects the development of both directions. A DWT/MC/DPCM based video
coding framework is introduced in Chapter 3, which adopts the EBCOT as the coding
engine for both the intra- and inter-coder. The MC in DWT domain and the coding of
motion information are discussed in Chapter 4. The context set for the inter-frame
coding is redesigned according to the statistical analysis in Chapter 5. In Chapter 6, a
Perceptual Distortion Measure (PDM) based on a human vision model is employed by
the intra-frame coder to replace the Mean Squared Error (MSE) in the R-D optimization
inside EBCOT. A visibility assessment on the quantization error in DWT domain is
presented in Chapter 6 as well. The conclusions are drawn in Chapter 7.

1.4 Contributions

The contributions of this thesis are:

- Provision of a comprehensive and up-to-date review of the DWT video coders in
  the literature, which provides a good understanding of this school of video
  compression technologies and the research trend in the literature.

- Proposition of a novel DWT/MC/DPCM/EBCOT based video coding framework,
  which naturally inherits the coding efficiency of JPEG2000 (JTC1/SC29 2004)
  for intra-frame data and the built-in rate-distortion optimization capability.
CHAPTER 1 INTRODUCTION

- Improvement and evaluation of the Motion Compensation (MC) of the above hybrid DWT based video coder, which includes the use of the simple Wavelet Tree (WT) based MC and the Low-Band-Shift (LBS) based MC (H.W.Park and H.S.Kim 2000).
- Evaluation of different arithmetic coding algorithms for motion information encoding, which includes the Witten’s algorithm (I.H.Witten, R.M.Neal et al. 1987) and the algorithm used in H.264 (D.Marpe, H.Schwarz and T.Wiegand 2003).
- Redesign the contexts set of the CBABAC for the inter-frame data, which is based on the mutual information analysis.
- Adaptation of the vision model based perceptual distortion measure into intra-frame coding of the proposed DWT video coder.
- Visibility assessment of the quantization error in the Mallat’s decomposition scheme using Deaubechies 9/7 filters (M.Antonini, M.Barlaud et al. 1992) for video data.

1.5 Thesis Layout

The outline of this thesis is as follows, which is also illustrated in Fig. 1.2. A review of the theoretical foundations for the DWT based video coding is provided in Chapter 2. A novel DWT/MC/DPCM based video coding framework is introduced in Chapter 3, which adopts the EBCOT as the coding engine for both the intra- and inter-coder. The MC in DWT domain and the coding of motion information are discussed in Chapter 4. The
Fig. 1.2 Roadmap of This Thesis

context set for the inter-frame data is redesigned according to the statistical analysis in Chapter 5, with the PSNR coding results. In Chapter 6, a Perceptual Distortion Measure
CHAPTER 1 INTRODUCTION

(PDM) based on a human vision model is employed by the intra-coder to replace the Mean Squared Error (MSE) in the R-D optimization mechanism inside EBCOT. A visibility assessment on the quantization error in the DWT domain is also presented in Chapter 6. The conclusions are drawn in Chapter 7. In order to make the content in this thesis self-contained, the details of the EBCOT are summarized in Appendix A. An extensive review of the DWT based video coders is given in Appendix B. This arrangement helps to answer the issues originated from the original idea: how to design a DWT based perceptual video coding framework? The findings include: a working DWT/MC/DPCM video coding framework with superior coding efficiency on head-shoulder sequences or those with translational motion; an adaptive switching mechanism between frame and field coding mode; an effective LBS based MC scheme in the DWT domain; a methodology of the context design for entropy coding of the inter-frame data; a PDM which replaces the MSE inside the EBCOT coding engine for the intra-frame coder, which provides improvement on the perceived quality of intra-frames; a visibility assessment to the quantization errors in the DWT domain.
Chapter 2

Foundation of DWT Video Coding

The theoretical foundation of the DWT based video coding is provided in this chapter to make the content of this thesis self-contained. The outline is as follows. An introduction of the theories associated with the fundamental techniques used in video coding is given in Section 2.1. A brief review of the DWT and relevant theories is given in Section 2.2, which encompasses the traditional DWT, the lifting scheme, and the more recent Adaptive Directional Lifting (ADL) scheme. Section 2.3 discusses the so-called shift variant problem of the MC in the DWT domain and the LBS approach as a solution to address this issue. Section 2.4 visits the bitplane based image/video coding techniques employed in the Embedded Zerotrees of Wavelet (EZW), the Set Partitioning in Hierarchical Trees (SPIHT) and the Embedded Block Coding with Optimized Truncation (EBCOT). A summary is given in Section 2.5.

2.1 General Structure for Traditional Video Coding

This section discusses briefly the fundamental techniques which constitute the traditional video coding frameworks. They encompass the predictive/ transform/ subband coding, the quantization, and the entropy coding.
2.1.1 Predictive, Transform and Subband Coding

In predictive coding, the “previous” data elements along the spatial horizontal and/or vertical or the temporal direction are used to predict the “current” data element. Then the magnitudes of the differential signals are formed by subtracting the values of the predictions from the actual values, before being encoded and transmitted (R.J.Clarke 1985).

Signal transformation is the procedure of mapping a signal from one domain, such as the original pixel domain, onto another one (K.R.Rao and J.J.Hwang 1996), such as the frequency domain. When the signal is transformed from original domain to frequency domain, it becomes more stationary in each frequency band and easier to code (N.Ahmed, T.Natarajan et al. 1974; I.Daubechies 1990; M.Vetterli and C.Herley 1992; R.J.Clarke 1995). The role of the traditional frequency transform here is twofold: energy packing and de-correlation (R.J.Clarke 1985). After transform, the distribution of the coefficient energy is skewed across different subbands. The so-called low-frequency / DC subband, which normally could be understood as the “approximate” or “average” version of the original signal, contains significantly more average energy than that of the high-frequency / AC subbands, while the latter contain the variance or detail information. Furthermore, the transform de-correlates the interrelationship between pixels along various directions. For audio signals, the transform is applied along the temporal direction. For still image signals, transforms are performed spatially along the horizontal and vertical direction. For video signals, the transforms may be applied spatially (horizontal and vertical) and/or temporally.
The SBC was initially introduced for speech compression as mentioned in (G.Karlsson and M.Vetterli 1988). It splits the signal into a number of separate frequency components (N.S.Jayant 1984; K.N.Ngan and W.L.Chooi 1994), and then encodes each of these components separately according to certain criteria such as activity, energy or perception related parameters. It is assumed that the input signal is sampled at a frequency of $f_s = 1/T$ where $f_s \geq 2f_{\text{max}}$ and $f_{\text{max}}$ be the highest frequency of the signal. The frequency $2f_{\text{max}}$ is also known as the Nyquist rate (A.V.Oppenheim and A.S.Willsky 1997). This sampled input signal is sent to all filters in the filter bank. The filters are all band-passed with the same bandwidth but distinct frequency band. The output of the filters are subsampled by a factor $N$, where $N$ is the number of filters in the filter bank. The spectral of each individual filter-decimator output has a period of $1/NT$. The above division removes correlations among input values and provides de-correlated coefficients for later stages such as the quantizer and the entropy coder. At the synthesis stage, the signal for each subband is interpolated first, i.e., up-sampled by a factor $N$. Then the signals are filtered by the synthesis filters and summed up to form the reconstructed signal.

2.1.2 Quantization

Quantization is the process of mapping signal values, continuous or discrete, to a restricted set of values. This procedure alters the information content and therefore is an irreversible operation. The readers are referred to (A.Gersho and R.M.Gray 1991; R.M.Gray and D.L.Neuhoff 1998) for more details of quantization.
2.1.3 Entropy and Entropy Coding

The information entropy, or simply entropy, is a measure of the theoretical average information amount conveyed by the uncertainty associated with a random variable. The concept was introduced by Shannon back to 1948 (C.E.Shannon 1948). Assuming the random variable is \( X \), the entropy can be computed as

\[
H(X) = -\sum_i p(x_i) \log p(x_i)
\]  

(2.1)

where \( \{x_i\} \) is the collection of values of the possible events and \( p(x_i) = P_r(X = x_i) \) is the probability mass function of \( X \). The logarithm is normally taken to base 2 therefore the entropy value is measured in the unit of bits. Entropy is the lower bound a statistical lossless compression system can achieve in terms of coding efficiency or bit amount.

It is straightforward to see from (2.1) that events with bigger probabilities should be assigned less bits and vice versa. Simply speaking, the entropy coder achieves compression by exploiting non-uniformity in the probability under which a random variable to be coded takes on its possible values. A good universal information coding review can be found in (T.C.Bell, J.G.Cleary et al. 1990). Entropy coding is a reversible process since the information coded by the entropy coder can be fully recovered by the entropy decoder given that both the encoder and decoder are employing the same statistical assumption and updating methods (for adaptive entropy coding). The Run-length Coding (RLC), Huffman Coding and Arithmetic Coding are among the popular
entropy coders. An early RLC is the work of Golomb’s (S.W.Golomb 1966) where the run-length of the unfavorable event is coded according to the probability distribution of events. The RLC is simple to implement, although generally its coding efficiency is lower than that of the Huffman Coding and the Arithmetic Coding. The RLC is still used in today’s image processing where the compression gain is not too demanding. The Huffman Coding was named in memory of Huffman’s invention of coding the information using a binary tree (D.A.Huffman 1952). For a period of time, Huffman coding was regarded as the optimal coder in terms of coding efficiency before the introduction of the Arithmetic Coding (J.J.Rissanen 1975) into the literature. Despite its sub-optimal coding efficiency performance, Huffman Coding is easy to implement and fast to execute, therefore it has been widely used (K.R.Rao and J.J.Hwang 1996). In comparison to Huffman Coding, which assigns an integer length of code for each symbol from the source alphabet, Arithmetic Coding assigns a real number length of code for each symbol corresponding to its probability. Hence its coding efficiency is closer to the information entropy. A message is represented by an interval of real numbers between 0 and 1. During the initialization stage, the range for the message is the entire interval $[0,1)$. During the encoding of each individual symbol, the range is adjusted to the portion allocated to the new symbol. The idea of arithmetic coding originated from Shannon’s work (C.E.Shannon 1948) where symbols were sorted by their probabilities and then the cumulative probabilities were sent to the decoder. Elias code (N.Abramson 1963) is a further development of the arithmetic coding where Shannon’s scheme worked without sorting the probabilities, and the cumulative probabilities could be recursively calculated from individual symbol probability of the source string. Nonetheless, both the Shannon’s
and the Elias’ scheme suffered from increasing precision and therefore are difficult for real applications. The first Arithmetic Coding which solved the precision problem was a Last-in-first-out (LIFO) scheme, which was introduced by Rissanen in 1975 - 1976 (J.J.Rissanen 1975) (J.J.Rissanen 1976). At almost the same time, Pasco proposed a First-in-first-out (FIFO) Arithmetic Coder (R.Pasco 1976) using a similar idea to that of (J.J.Rissanen 1975). In an important work of Rissanen and Langdon (J.Rissanen and G.G.Langdon 1981), it stated that models that use any kind of alphabet extension were inferior to the best models using no alphabet extensions at all. Alphabet codes, as a result, could be used in conjunction with the best models. The work of (G.G.Langdon and J.Rissanen 1981) can be regarded as a companion with that of (J.Rissanen and G.G.Langdon 1981), while the former provided a good application case analysis, the latter offered a solid theoretical foundation.

2.1.4 Context Design — Root in Generic Entropy Coding

Assuming the information source is a discrete random process \( X = (X_1, X_2, \ldots, X_n) \), where \( X_i \) represents the discrete random variable at moment \( i \). At each moment, the value of \( X_i \) is taken from a finite alphabet. For Binary Arithmetic Coding (BAC), the alphabet is \( \{0,1\} \). In reality, the value of the message originates from the above random process can be represented by \( x = (x_1, x_2, \ldots, x_n) \), where each symbol \( x_i \) is the discrete value taken at moment \( i \). The lower bound of the average length used to code the messages out of the above random process is the entropy given by (2.2).
However, when the length of the message grows out of control, the computation of (2.2) will become difficult, and sometimes unrealistic. One possible solution of this problem is the so-called chain rule of the entropy (T.M.Cover and J.A.Thomas 1991) formulated by (2.3),

\[ H(X_1, X_2, \cdots, X_n) = \sum_{i=1}^{n} H(X_i | X^{i-1}) \]  

(2.3)

where \( X^{i-1} = X_1, X_2, \cdots, X_{i-1} \). Therefore the computation of the entropy is decomposed into the computation of conditional entropies. The corresponding minimum code length is represented by

\[ -\sum_{i=1}^{n} \log_2 p(X_i | x^{i-1}) \]  

(2.4)

where \( x^{i-1} = x_1, x_2, \cdots, x_{i-1} \). Unfortunately, for real applications especially audio, image and video coding, the computation of (2.4) is still fairly difficult. An approximation of the conditional probabilities in (2.5) can greatly simplify the computation with acceptable level of precision penalty.

\[ H(p(X_i | f(x^{i-1}))) \]  

(2.5)
where the mapping function $f(\cdot)$ is named as context set. This approximation is to assign each possible $x^{i-1}$ to a conditioning context. It can be concluded from (2.5) that the entropy coding can be partitioned into two components: the modeling and the coding. The modeling component is responsible for the estimation of the probability $p(X_i|f(x^{i-1}))$.

Assuming the value of the estimated probability by the modeler is $\hat{p}(X_i|f(x^{i-1}))$, an ideal coder will generate $-\log_2(\hat{p}(X_i = k|f(x^{i-1})))$ when the event $X_i = k$ occurs.

The average code length for $X_i$ is therefore represented by

$$
\sum_{k=1}^{N} p(X_i = k|f(x^{i-1})) \log_2 \left( \frac{1}{\hat{p}(X_i = k|f(x^{i-1}))} \right)
$$

$$
= \sum_{k=1}^{N} p(X_i = k|f(x^{i-1})) \log_2 \left( \frac{1}{p(X_i = k|f(x^{i-1})) \cdot \hat{p}(X_i = k|f(x^{i-1}))} \right)
$$

$$
= H(p(X_i|f(x^{i-1}))) + D(p(X_i|f(x^{i-1})||\hat{p}(X_i|f(x^{i-1})))
$$

(2.6)

where $D(\cdot\|\cdot)$ is the relative entropy between two distributions, and is also named as the Kullback-Leibler distance (T.M.Cover and J.A.Thomas 1991). It is defined in (2.7) by information theory (T.M.Cover and J.A.Thomas 1991),

$$
D(p(x)\|q(x)) = \sum_x p(x) \cdot \log_2 \frac{p(x)}{q(x)}
$$

(2.7)
where \( D(p(x)\|q(x)) \geq 0 \) and the equality holds if and only if \( p(x) = q(x) \). It is therefore straightforward to see from (2.6) that the use of the modeling leads to a penalty of \( D(p(X|f(x^{i-1}))\|\hat{p}(X|f(x^{i-1}))) \) with respect to the average code length. The quality of the modeling will have a direct impact on the value of this penalty. If the modeling can predict the probability ideally well, this penalty will be trivial and the average number of bits used to code the message will be close to its entropy. On the other hand, if the modeling provides a poor prediction of the probability, the relative entropy value will be fairly high. In the worst case, the message will be expanded instead of being compressed.

Information theory gives a more general relationship between two random variables in

\[
H(X|Y) = H(X) - I(X;Y)
\]

(2.8)

where \( H(X|Y) \) is a more generic form of conditional entropy and \( I(X;Y) \) the mutual information. Thus minimizing the conditional entropy means the maximization of the mutual information. In context based entropy coding, \( Y \) can be replaced with the mapping function, or the context set \( f(x^{i-1}) \).

Theoretically, it is better to have as many contexts as the distinct values of \( x^{i-1} \) because this ‘ideal’ coder will achieve the minimum conditional entropy or the maximum mutual information. In reality, however, the numbers of symbols to be coded are finite, therefore
too many contexts will cause the so-called context dilution problem (Z.Liu and L.J.Karam 2005). In the mean time, too many contexts will increase the coder’s complexity.

In order to overcome the above context dilution problem and lower the complexity of the coder, a suitable number of contexts are required which provide sufficiently accurate models for the encoding of symbols. The classification scheme proposed in (Z.Liu and L.J.Karam 2005) is adopted by the work in Chapter 5 because of the easiness for understanding and the computation efficiency. The principle can be described as: given $I$, the initial number of contexts, and $F$, the final desired number of contexts, there are $S(I,F)$ possible classification schemes. Among them, there must be an optimal classification scheme which gives the maximum MI. The classification is essentially the combination operation of contexts. The mutual information will decrease if contexts are combined unless the conditional probability distributions of the combined contexts are the same. Here the reduction of the mutual information is denoted by $MI_{\text{red}}$. The function $MI_{\text{red}}(i \leftrightarrow j)$ is used to represent for the mutual information reduction caused by the combination of context $i, \cdots, j$.

The computation of the optimal classification adopts dynamic programming in order to speed up the execution. Assuming the initial number of contexts is $I$, the algorithm can be described as follows:

1) Precompute the $MI_{\text{red}}(i \leftrightarrow j)$ for $i=1 \ldots I, j = i \ldots I$;

2) Order the $I$ contexts according to $p(y|X), y = 1$ or $0$, in the ascending order;
3) Denoting the MI reduction due to optimal classification of context \(i \ldots j\) into \(m\) groups by \(MI_{\text{red}}(i : j \Rightarrow m)\), the dynamic programming procedure can be described as in (2.9).

\[
MI_{\text{red}}(i : I \Rightarrow F) = \min_{i = 1 : \text{tod} - F + 1} \left( MI_{\text{red}}(i \leftrightarrow j) + MI_{\text{red}}(j + 1 : I \Rightarrow F - 1) \right)
\]

where \(MI_{\text{red}}(i : j \Rightarrow 1) = MI_{\text{red}}(i \leftrightarrow j)\).

The dynamic programming in step 3) will produce a mutual information reduction matrix as in (2.10).

\[
\begin{align*}
MI_{\text{red}}(1 : I \Rightarrow 2) & \quad MI_{\text{red}}(2 : I \Rightarrow 2) & \quad \ldots & \quad MI_{\text{red}}(I - 2 : I \Rightarrow 2) & \quad MI_{\text{red}}(I - 1 : I \Rightarrow 2) \\
MI_{\text{red}}(1 : I \Rightarrow 3) & \quad MI_{\text{red}}(2 : I \Rightarrow 3) & \quad \ldots & \quad MI_{\text{red}}(I - 2 : I \Rightarrow 3) \\
\vdots & \quad \vdots & \quad \quad & \quad \vdots \\
MI_{\text{red}}(1 : I \Rightarrow F) &
\end{align*}
\]

(2.10)

The computation of the above matrix elements is from top to bottom and left to right. The element in the last row, \(MI_{\text{red}}(1 : I \Rightarrow F)\), is the optimal classification. The whole classification scheme can be traced backwards from this element according to (2.10).

2.1.5 MQ Coder —— An Adaptive Binary Arithmetic Coder

The MQ coder is an adaptive binary arithmetic coder, which is implemented as a 47-states state machine (D.S.Taubman and M.W.Marcellin 2002). It shares similarity with its
predecessors the Q coder (W.B.Pennebaker and J.L.Mitchell 1988; W.B.Pennebaker, J.L.Mitchell et al. 1988) and the QM coder (W.B.Pennebaker and J.L.Mitchell 1992), and the rival M coder which emerged more recently (D.Marpe and T.Wiegand 2003). It is worth mentioning here that the MQ coder and the M coder has been adopted as the coding engine of JPEG 2000 (JTC1/SC29 2004) and H.264 (ITU-T 2005), respectively. The performance comparisons between the MQ coder and the M coder can be found in (D.Marpe and T.Wiegand 2003). All the above coders are multiplication-free arithmetic coders. They use a set of rules based on renormalization to provide probability estimations, which essentially follow those in (W.B.Pennebaker and J.L.Mitchell 1988).

2.2 Discrete Wavelet Transform (DWT)

2.2.1 Traditional DWT

Wavelets are functions generated from a primary function, $\psi$, through dilations and translations. The continuous form of wavelets can be represented by (2.11) where $\psi$ is the mother wavelet.

$$\psi^{a,b}(t) = \left| a \right|^{-\frac{1}{2}} \psi \left( \frac{t-b}{a} \right)$$

(2.11)

The function $\psi$ needs to satisfy the condition $\int \left| \Psi(\omega) \right|^2 |\omega|^{-1} d\omega < \infty$, where $\Psi$ is the Fourier transform of $\psi$. This condition can be loosened to $\int \psi(x) dx = 0$, if $\psi(t)$ decays faster than $|t|^{-1}$ for $t \to \infty$. The basic idea of the wavelet transform is to represent any
arbitrary function $f$, or signal, as a superposition of wavelets. In practice, one can define $a = a_0^m$, $b = nb_0 a_0^m$ with $a_0 > 1$, $b_0 > 0$. Both $m$ and $n$ are integers bounded by $(-\infty, \infty)$.

The wavelet decomposition in discrete form can therefore be represented by

$$f = \sum c_{m,n}(f) \psi_{m,n}$$

where $\psi_{m,n}(t) = \psi_{a_0^{m+2},nb_0 a_0^m}(t) = a_0^{-m/2} \psi(a_0^{-m} t - nb_0)$.

Mallat proposed a multiresolution decomposition scheme in (S.G.Mallat 1989). This scheme adopts two families of functions: the wavelet function $\psi$ and the scaling function $\phi$. There is also a dilated and translated version of the scaling function denoted as $\phi_{m,n}(x) = 2^{-m/2} \phi(2^{-m} x - n)$. The $\phi_{m,n}(x)$ function spans the approximation space $V_m$ of the original signal at resolution $2^m$, while the $\psi_{m,n}(x)$ spans the orthogonal complement in $V_{m-1}$ of $V_m$. That is, the transform coefficients $\langle \psi_{m,n}, f \rangle$ describe the difference of the information between resolution $2^{m-1}$ and $2^m$. The projection of signals onto the space $V_{m-1} - V_m$ and $V_m$ are given by

$$c_{m,n}(f) = \langle \psi_{m,n}, f \rangle = \sum_k g_{2^{m-k}} a_{m-1,k}(f)$$

$$a_{m,n}(f) = \langle \phi_{m,n}, f \rangle = \sum_k h_{2^{m-k}} a_{m-1,k}(f)$$

(2.13)
where \( g_t = (-1)^t h_{t+1} \) and \( h_n = 2^{1/2} \int \phi(x-n)\phi(2x)dx \). \( g \) and \( h \) are the high- and low-pass filters, respectively. \((\cdot)\) is the convolution operator. As stated by Antonini et al. (M.Antonini, M.Barlaud, P.Mathieu et al. 1992), \( a_{m,n}(f) \) is the projection of \( f \) onto the space \( V_m \). Its value at the highest resolution is denoted as \( a_{0,n} \). (2.13) depicts the analysis (forward) process while the synthesis process is given by

\[
a_{m-1,n}(f) = \sum_n \left( \tilde{h}_{2n-1} a_{m,n}(f) + \tilde{g}_{2n-1} c_{m,n}(f) \right)
\]

(2.14)

If \( \tilde{h} \) and \( \tilde{g} \) are the orthogonal/biorthogonal complement of \( h \) and \( g \), then (2.14) provides exact reconstruction.

It is desirable for the wavelet filters \( h \) and \( g \) to have finite numbers of taps from practical point of view. In this respect, orthonormal\(^1\) wavelet bases with compactly supported taps are of practical importance (M.Vetterli 1985; M.J.Smith and D.P.Barnwell 1986; I.Daubechies 1988).

A number of characteristics are desired when designing wavelet filters. Firstly, filters need to be reasonably short in length for filtering speed performance. Secondly, filters also need to have sufficient length to be smooth. Simply speaking, filters can be neither too short nor too long. Linear phase property is also highly desirable because it eliminates the need for phase compensation. Unfortunately, the two-tap Haar filter is the only known orthonormal linear phase FIR filter (M.J.Smith and D.P.Barnwell 1986; I.Daubechies 1988).

\(^1\) Orthonormal filters are normalized orthogonal filters
The biorthogonal wavelet filters were introduced to fulfill the linear phase requirement. For biorthogonal filters, the analysis and synthesis are still represented by (2.13) and (2.14). However, there are differences in derivations of synthesis filters between orthogonal and biorthogonal systems. For orthogonal filters, the synthesis filters are derived through the transpose of the analysis filters, i.e., $\tilde{h} = h'$ and $\tilde{g} = g'$. For biorthogonal filters, the synthesis filters do not equate to the transpose of the analysis filters, i.e., $\tilde{h} \neq h'$ and $\tilde{g} \neq g'$. The design conditions for biorthogonal filters are given in (M.Vetterli and J.Kovacevic 1995). Biorthogonal filters allow for exact reconstruction as formulated by (2.14). In Fig. 2.1, $\hat{X}$ is a perfect, but possibly delayed replica of the input signal $X$. For the analysis filter bank, the scaling ($\phi$) and the wavelet ($\psi$) functions are used to derive the filter kernels of the low and high pass filters $h$ and $g$, respectively.

![Filter Bank Structure](image)
The same relationship holds for $\tilde{\phi}$ and $\tilde{\psi}$ at the synthesis side. For a mathematical proof, please refer to (A.Cohen, I.Daubechies et al. 1992; M.Antonini, M.Barlaud, P.Mathieu et al. 1992). It is worth mentioning that the biorthogonal 9/7 and 5/3 filter banks (M.Antonini, M.Barlaud, P.Mathieu et al. 1992) have been formally adopted by the JPEG2000 still image compression standard (JTC1/SC29 2004). The coefficients of the biorthogonal 9/7 filter bank are illustrated in Fig. 2.2. For practical implementation, the biorthogonal 9/7 filter bank may have different normalization schemes (M.Rabbani and R.Joshi 2002). Part of the DWT process involves downsampling which decimates transform signals by a factor of 2:1 in the analysis stage (M.Vetterli and J.Kovacevic 1995). This 2:1 decimation operation can be traced back to the work of the Quadrature Mirror Filters (QMF) (D.Esteban and C.Galand 1977). A reciprocal upsampling with a factor of 1:2 is performed on transform signals prior to synthesis process. For Fig 2.1, in the absence of information loss, the relationship between the input signal, $X$, and the reconstruction, $\hat{X}$, in the Z-transform domain is given by

$$\hat{X}(z) = z^{-k}X(z), \quad k \in \mathbb{N}.$$  \hspace{1cm} (2.15)

In the Mallat’s multiresolution decomposition scheme (S.G.Mallat 1989), the transform may be applied recursively in successive low-pass subbands. Due to downsampling operation, the resolution along each dimension is half size of its parents, e.g., for 2-D signals such as images, each successive resolution is a quarter the size of its parent. For
video, the decimation operation leads to shift-variant problem (H.W.Park and H.S.Kim 2000) between successive video frames.

2.2.2 Lifting Implementation of DWT

The filtering stages of the forward and inverse DWT in Section 2.2.1 can be implemented by using the traditional convolution operations. In addition, they may be performed through the lifting scheme (W.Swel...W.Swel...I.Daub...and W.Swel...1998). The lifting scheme uses intermediate stages to perform the DWT. This approach significantly reduces the memory usage and computation complexity of the DWT in comparison with the convolution. Generally, the lifting operation consists of three major steps: lazy wavelet transform; prediction; and
updating. The lazy transform simply categorizes the input sequence, \( x_i \), into even and odd indexed subsequences denoted as \( s_i^0 \) and \( d_i^0 \), respectively. The prediction step computes each odd sample as a linear combination of the even samples and subtracts it from the odd sample to form the prediction error as

\[
d_i^n = d_i^{n-1} + \sum_k P_n(k) x_k^{n-1}, \quad n \in \{1,2,\cdots,N\}. \tag{2.16}
\]

The update step computes the even samples by linear summation of modified odd samples derived from (2.16). The updating step can be represented by

\[
s_i^n = s_i^{n-1} + \sum_k U_n(k) d_k^n, \quad n \in \{1,2,\cdots,N\}. \tag{2.17}
\]

In both (2.16) and (2.17), the value \( k \) and corresponding weights \( P_n(k) \) and \( U_n(k) \) depend on the specific DWT filter-bank the lifting scheme implements. A general block diagram of the lifting scheme for 1-D signals is shown in Fig. (2.3). Both prediction and update steps are executed iteratively for \( N \) stages, with different weights at each stage. The output of the final prediction step is scaled by a factor \( K_1 \) to produce the high-pass DWT coefficients. The output of the final update step is scaled by factor \( K_0 \) to produce the low-pass DWT coefficients. The values of \( N \), \( K_0 \) and \( K_1 \) are dependent on the filter-bank. For 2-D signal such as still images, the lifting scheme is alternatively applied in both the horizontal and vertical spatial direction as traditional convolution scheme does.
2.2.3 Adaptive Directional Lifting DWT

For 2-D signals, the existing dyadic DWT framework generally applies separable DWT analysis and synthesis in the horizontal and the vertical spatial direction. This approach is also called the rectilinear 2-D DWT (W. Ding, F. Wu, X. Wu et al. 2007). One of the drawbacks of this approach is the lack of adaptation to image features with arbitrary orientation that are neither vertical nor horizontal. As a result, the traditional rectilinear DWT produces large-magnitude high-frequency coefficients. Recently, Ding et al. (W. Ding, F. Wu, X. Wu et al. 2007) proposed the so-called Adaptive Directional Lifting (ADL) DWT scheme. The ADL scheme performs lifting-based prediction, in local windows, in the direction of high pixel correlation. The prediction and update steps of ADL can be carried out even at the sub-pixel precision level to achieve higher directional resolution, while still maintaining perfect reconstruction. The 2-D ADL can be understood as a procedure with column direction 1-D ADL followed by row direction 1-
D ADL. The 1-D ADL requires the coordinate computation in both horizontal and vertical directions. The column direction ADL is illustrated in Fig. 2.4.

First, the column-wise even and odd samples are denoted by $x_e$ and $x_o$. They are computed from

\[
x_e(m,n) = x(m,2n) \\
x_o(m,n) = x(m,2n + 1)
\]  

(a) Prediction
Fig. 2.4 Prediction and update steps of the ADL (W.Ding, F.Wu, X.Wu et al. 2007).

Note that the vertical angle is $\theta_v$. The values at the integer, half and quarter pixel locations are represented by the shaded circle, the plus and the cross, respectively.

Then the prediction step is executed as

$$h(m, n) = x_o(m, n) - \sum_{i=a}^{b} \alpha_i x_v(m + \text{sign}(i-1)\tan \theta_v, n + i), \quad (2.19)$$

where $\text{sign}(x) = 1$ for all $x \geq 0$ and -1 otherwise, $a$ and $b$ delimit the finite support of the FIR wavelet filter.

In the update step, the even samples are computed as
The global ADL scheme partitions the image recursively into blocks of variable sizes by quad-tree. All pixels in a quad-tree block will be subject to the same ADL transform. The side information used to describe the segmentation tree and the lifting directions of individual blocks must be sent to the decoder. A JPEG2000-alike coder for still images was used in conjunction with the ADL in (W. Ding, F. Wu, X. Wu et al. 2007) by simply replacing the DWT transformation with the ADL scheme. Experimental results showed that the ADL-based coder outperformed JPEG2000 in both PSNR and perceived quality, with the improvement up to 2.0 dB on images with rich orientation features.

2.3 Motion Compensation in DWT Domain

Zhang and Zafar proposed the MRMC (Multi-Resolution Motion Compensation) scheme in the wavelet domain (Y.-Q. Zhang and S. Zafar 1992) that adopted variable block-size for different resolutions. It addressed the global motion correlation of the video signal at different scales. The motion vectors of the lowest resolution are used as the bias for higher resolutions. Joo and Kikuchi (S. Joo and H. Kikuchi 2000), on the other hand, proposed a direct Wavelet-Tree based MC (WTMC). This approach has been adopted in this thesis work with modifications in the MV information encoding to simplify the implementation and reduce overhead of motion information. The formation of a wavelet tree for WTMC is illustrated in Fig. 2.5, where a DWT tree is made of nodes from

\[
l(m, n) = x_e(m, n) + \sum_{j=0}^{d} \beta_i h(m + \text{sign}(j) \tan \theta_i, n + j)
\]  

(2.20)

2.3 Motion Compensation in DWT Domain

Zhang and Zafar proposed the MRMC (Multi-Resolution Motion Compensation) scheme in the wavelet domain (Y.-Q. Zhang and S. Zafar 1992) that adopted variable block-size for different resolutions. It addressed the global motion correlation of the video signal at different scales. The motion vectors of the lowest resolution are used as the bias for higher resolutions. Joo and Kikuchi (S. Joo and H. Kikuchi 2000), on the other hand, proposed a direct Wavelet-Tree based MC (WTMC). This approach has been adopted in this thesis work with modifications in the MV information encoding to simplify the implementation and reduce overhead of motion information. The formation of a wavelet tree for WTMC is illustrated in Fig. 2.5, where a DWT tree is made of nodes from
various oriented subbands at different resolutions. These related nodes (coefficient blocks) can also be conceptually grouped into a wavelet block (WB). Both the WB and the WT are essentially descriptions of the same set of DWT coefficients from different perspectives. The nodes are indicated by the black blocks and connected by arrows in Fig. 2.5. In fact, each individual node is composed of a small group of DWT coefficients. The parent node is in the DC band while the remaining nodes are regarded as its children. If the size of the parent node in the DC band is \( m \times n \), then its sibling nodes in the \( HL, LH \), and \( HH \) subbands in the lowest resolution is \( m \times n(m \in N, n \in N) \) as well. The sizes of other children nodes are represented by \( m \cdot 2^{-i} \times n \cdot 2^{-i} \), where \( i \) is the resolution level as illustrated in light gray color in Fig. 2.5. In the work of this thesis, the values of \( m, n \) are 2 and 2. The maximum value of \( i \) is 3.

Motions in the DWT domain between two successive pictures are estimated on the basis of the distance between wavelet trees. A single motion vector is used to describe the translational motion of the whole tree. ME procedure finds the best matching wavelet tree in the reference picture (in DWT domain) relative to the current picture according to a distortion measure. Here, the Sum of Absolute Differences (SAD), defined in (2.21), is employed as the distortion measure, where \( (m, n) \) is the location within the DC band, i.e., the top-left corner area (marked by “0”) in Fig. 2.5. \( HL_{ref,i}, LH_{ref,i} \) and \( HH_{ref,i} \) are the \( i \) th-level \( HL, LH \) and \( HH \) subbands of the reference frame, respectively. \( LL_{ref} \) stands for the horizontal low- vertical low subband in the lowest resolution. \( HL_{cur,i}, LH_{cur,i} \) and \( HH_{cur,i} \) and \( LL_{cur} \) have similar definition except that they are from the current frame.
Fig. 2.5 Forming of Wavelet Tree

\( (\Delta m, \Delta n) \) is the motion vector (motion information) measured at the lowest DWT resolution.

\[
SAD(\Delta m, \Delta n) = \sum_{i=1}^{3} \sum_{j=0}^{2^{i-1}-1} \sum_{k=0}^{2^{i-1}-1} \left[ HL_{\text{ref},i}\left((m + \Delta m) \cdot 2^{i-1} + j, (n + \Delta n) \cdot 2^{i-1} + k\right) - HL_{\text{cur},i}\left((m + \Delta m) \cdot 2^{i-1} + j, (n + \Delta n) \cdot 2^{i-1} + k\right)\right] + LH_{\text{ref},i}\left((m + \Delta m) \cdot 2^{i-1} + j, (n + \Delta n) \cdot 2^{i-1} + k\right) + LH_{\text{cur},i}\left((m + \Delta m) \cdot 2^{i-1} + j, (n + \Delta n) \cdot 2^{i-1} + k\right) + HH_{\text{ref},i}\left((m + \Delta m) \cdot 2^{i-1} + j, (n + \Delta n) \cdot 2^{i-1} + k\right) - HH_{\text{cur},i}\left((m + \Delta m) \cdot 2^{i-1} + j, (n + \Delta n) \cdot 2^{i-1} + k\right) + \right.
\]

\[
+ j, (n + \Delta n) \cdot 2^{i-1} + k\right] + \sum_{j=0}^{1} \sum_{k=0}^{1} \left[ LL_{\text{ref},i}(m + \Delta m + j, n + \Delta n + k) - LL_{\text{cur},i}(m + j, n + k)\right]
\]
2.3.1 Low-Band-Shift Motion Estimation

It is important to address the shifted-variant problem in a complete-DWT system. Simply speaking, there exist very large differences between the wavelet coefficients of an image and those with odd-pixel shifts (H.W.Park and H.S.Kim 2000). This difference is quite obvious in the oriented subbands (HL, LH and HH). This problem is demonstrated in Fig. 2.6. In Fig. 2.6, (a) a 1-level DWT is applied to frame 51 of the “Football” sequence and (b) is the 1-level DWT of the right-shifted (1 pixel) version of (a). (c), (d), (e) and (f) are the zoomed-in areas of (a). They are taken from LL, HL, LH and HH band, respectively. (g), (h), (i) and (j) are the zoomed-in areas of (b) from LL, HL, LH and HH bands, respectively. Although the head of the athlete is assumed to be relatively rigid, shift variant problem can still be observed in (d)-(j).

In order to overcome the shift-variant problem, the Low-Band-Shift (LBS) method was proposed in (H.W.Park and H.S.Kim 2000). The procedure is illustrated in Fig. 2.7 and Fig. 2.8. Here, the data structures of node and tree are used to facilitate the description of LBS. A node contains the following information:

1) horizontal shift history from the root of the tree;
2) vertical shift history from the root of the tree;
3) two dimensional array containing the subband data before the next level DWT;
4) current DWT level which the node belongs to;
(a) marked ROI in Football No.51

(b) marked ROI in Football No.51 after 1 pixel Horizontal Shift
5) weight associated with the current node;

6) four children produced by this node, which are the unshifted, 1 pixel shift along horizontal direction, 1 pixel shift along vertical direction, and 1 pixel shift along both horizontal and vertical directions, respectively.

The LBS procedure generates a quadric tree with each node in the tree having four children except for leaves. The root node of the tree contains the input image data. Its four direct children contain the unshifted, horizontally shifted by 1 pixel, vertically shifted by 1 pixel, and diagonally shifted (shifted on horizontal and vertical direction by 1 pixel, respectively), version of data, respectively. For each one amongst these four children, the data is decomposed into LL, HL, LH, and HH bands. The next level of LBS will be carried on based on the coefficients of the LL bands. For a n-level LBS-DWT system, there are \( \frac{4^{n+1} - 1}{3} \) nodes including the root. The size of data in each individual
node depends on the level. (It is assumed that the level index is incremented from the root downwards and the level index starts from 0.) The width and height of the input data is designated as $width_{Input}$ and $height_{Input}$, respectively. Then the width and height of the node at level $n$ is $\frac{width_{Input}}{2^{n-1}}$ and $\frac{height_{Input}}{2^{n-1}}$, respectively.

Fig. 2.7  1 Level Low Band Shift
CHAPTER 2    FOUNDATION OF DWT VIDEO CODING

Fig. 2.8  3 Levels of Low Band Shift

(note: each LL subband derived from 1-level DWT of its parent subband is denoted as $s_{i}^{\text{inherit,cehistory}}$, where $i$ is the resolution level. $u$, $h$, $v$, and $d$ represent for unshifted, horizontal-shifted, vertical-shifted, and diagonal-shifted, respectively.)

Each LBS node “remembers” the shift history of all its ancestors starting from the root. Four constants are defined to represent the shift at each individual stage and they are UNSHIFT, SHIFT_HOR, SHIFT_VER and SHIFT_DIA, respectively. The subband data within a specific node can be located using 1) level index; 2) shift history; 3) subband
index. ME is performed between the current Wavelet Block (WB) and the reference WB. The Mean Absolute Difference (MAD) is used as the distortion measure between the two WBs. It is written as

\[
\text{MAD}_k(\Delta m, \Delta n) = \sum_{i=1}^{3} \sum_{m_k, n_k} \left\{ \left( \text{HL}_{\text{cur}}^{(i)}(m_i, n_i) - \text{HL}_{\text{ref}}^{(i)}(\Delta m 2^i, \Delta n 2^i, m_i + \left\lfloor \frac{\Delta m}{2^i} \right\rfloor, n_i + \left\lfloor \frac{\Delta n}{2^i} \right\rfloor) \right)^2 \right\}
\]

where \((m_i, n_i)\) is the initial point of the \(i\) th-level subbands in the \(k\) th Wavelet Block. \(\text{HL}_{\text{cur}}^{(i)}, \text{HL}_{\text{ref}}^{(i)}\) and \(\text{HH}_{\text{cur}}^{(i)}, \text{HH}_{\text{ref}}^{(i)}\) are \(i\) th-level HL, LH and HH subbands from the reference frame, respectively. \(\text{LL}_{\text{cur}}^{(3)}\) is for the horizontal low-vertical low subband in the lowest resolution. \(\text{HL}_{\text{cur}}^{(3)}, \text{LL}_{\text{cur}}^{(3)}, \text{HH}_{\text{cur}}^{(3)}\) and \(\text{LL}_{\text{ref}}^{(3)}\) have similar definition as above except that they are from the current WB. \((\Delta m, \Delta n)\) is the motion vector measured at the original spatial resolution of the given image.

Under the LBS scheme, the ME and MC are performed between the current wavelet transformed frame and the reference un-decimated transformed frame “as if” it is in the original spatial domain. Signals that would have been lost during to the down-sampling in
DWT operation are preserved in the LBS. These signal samples help to provide additional accuracy for motion compensation in the DWT domain. ME/MC is no longer just performed between one wavelet tree and another as the latter approach is too “rough”. The LBS approach avoids the shift-variant problem, but incurs increased computation cost and memory usage. The memory used to buffer the whole LBS tree is 10 times of the given image size.

2.4 Bitplane Based Image/Video Coding

One of the earliest work of bitplane coding was reported by (J.W.Schwartz and R.C.Barker 1966) in 1966. It used Run End Coding (REC) / Run Length Coding (RLC) for the coding at each individual bitplane. Both the lossless and the lossy coding modes were offered with an implicit principle of coding the more important information at higher bitplanes before the coding of less important information at lower bitplanes.

The application of hierarchical bit-plane coding in video has its root in that of the still image, where the hierarchical bit-plane image coding is used for scalable image compression. With an embedded bit stream concept, the encoder can stop at any point during the generation of the bitstream, and the decoder can stop decoding at any point as well, with a coarser-to-finer reconstructed quality. An illustration of the embedded bitstream can be seen in Fig. 2.9. Among the most well-known scalable image compression schemes are Shapiro’s EZW (J.M.Shapiro 1993), Said and Pearlman’s SPIHT (A.Said and W.A.Pearlman 1996), and Taubman’s EBCOT (D.Taubman 2000).
2.4.1 Bitplane Representation

A digital image consists of pixels, which can be organized into two dimensional arrays. Each pixel can be represented by a series of bitplanes. The value at each bitplane is either 0 or 1. The bitplane decomposition is illustrated in Fig. 2.10.
2.4.2 Zero-tree coding

The concept of zero-tree stems from the fact that a wavelet transform is always followed by quantization for compression purpose. As a result, there are a high proportion of zero coefficients. There is high correlation across different coefficient positions, orientations, and scales. In order to improve the compression efficiency, it has been suggested that if a coefficient at a coarser scale is insignificant, or less than, when comparing to a octavely decreasing threshold $T$, then all the coefficients at finer scale but in the same orientation and position are likely to be insignificant as well. These insignificant coefficients therefore form a zerotree.

2.4.2.1 EZW and 3D-EZW

EZW was introduced by Shapiro (J.M. Shapiro 1993). The basic idea of that scheme is to compare the values of the transform coefficients against a series of thresholds where the thresholds at different decomposition levels have a dyadic relationship. This comparison generates significance map and nonzero values. The significance map and nonzero values are then coded by an adaptive arithmetic coder. It is this half-changing threshold that makes the bit stream embedded. Before the introduction of the implementation detail of EZW, it is important to be aware of the scanning sequence of the zerotree and the related data structures. Here the scanning sequence means the order in which sequence of coefficients in the wavelet domain are visited, which always starts from the coarser level, from HL band, to HL band, to HH band, before it goes to the next finer level, and so on. The exception occurs at the lowest level where the scanning begins at the LL band. There
are two important data structures involved in the scanning procedure, which are essentially two separate wavelet coefficients lists. The first list is called the dominant list, which contains the coordinates of those coefficients that have not yet been significant in respect to the current thresholds. The second one is called the subordinate list, which contains the magnitude of those significant coefficients. At each threshold level, the dominant list is always visited first followed by the subordinate list. It can be understood that the dominant list contains the significance map information and the subordinate list contains the non-zero value information. The significance map is then zero-tree coded in conjunction with the adaptive arithmetic coder. The non-zero values are first represented by binary alphabet symbols then coded by an adaptive arithmetic coder. The arithmetic coder is the same as that in (I.H.Witten, R.M.Neal and J.G.Cleary 1987). 3D-EZW was proposed as a natural extension of EZW (Y.Chen and W.A.Pearlman 1996) into video compression. The algorithm achieved comparable results in respect to MPEG-2. A 3D orientation tree was produced through DWT decomposition as in Fig. 2.11. In that work, a separable 3-D Quadrature Mirror Filter (QMF) band was used. The sequence was broken into segments of 16 frames and each individual segment was coded separately.
2.4.2.2 SPIHT and 3D-SPIHT

Sharing the basic idea with EZW, SPIHT was first suggested in (A. Said and W. A. Pearlman 1996). The output information in the bit stream still consists of two parts as in EZW, which are the significance map and the non-zero values. But the code is more compact so that it achieves comparable or even higher compression efficiency than the original EZW even without adaptive arithmetic coding. SPIHT still utilizes the important concept of zerotree. Before the introduction of its detail, it is necessary to brief some of the definitions/symbols used in the algorithm. It is noted that there is a node data structure, which corresponds to each transform coefficient.

- $O(i, j)$: set of coordinates of all offspring of node $(i, j)$;
- $D(i, j)$: set of coordinates of all descendants of the node $(i, j)$;
• $H$: set of coordinates of all spatial orientation tree roots (nodes in the highest pyramid level);

• $L(i, j) = D(i, j) - O(i, j)$

Fig. 2.12 Illustration of parent-offspring relationship in SPIHT

(A.Said and W.A.Pearlman 1996)

The significance information is stored in three ordered lists, and they are List of Insignificant Sets (LIS), List of Insignificant Pixels (LIP), and List of Significant Pixels (LSP). SPIHT also adopted an octavely decreasing quantization step as in EZW. At each quantization level, the processing is composed of a Sorting Pass and a Refinement Pass. For implementation details, please refer to (A.Said and W.A.Pearlman 1996).

3-D SPIHT was developed in a pretty straightforward way in (B.-J.Kim, Z.Xiong et al. 2000). The authors of (B.-J.Kim, Z.Xiong and W.A.Pearlman 2000) tried both the
wavelet packet transform and the dyadic wavelet transform along the temporal direction, which were complete-transform systems, to generate the spatio-temporal orientation tree. Here the wavelet packet transform case is illustrated in Fig. 2.13 because it is more generic. An orientation tree can thus be built by establishing a parent-offspring relationship. Once this orientation or zero-tree was built, a similar sorting pass according to the magnitude and refinement pass was applied to the coefficients. In that work, the algorithm derived results comparable to H.263 both objectively and subjectively. Felts and Pesquet-Popescu proposed a context modeling scheme based on 3D SPIHT video compression framework (B.Felts and B.Pesquet-Popescu 2000). They introduced several statistical models based on Context Tree Weighting (CTW) method to account for the distinct information resources within the 3D SPIHT infrastructure. Their method proved consistent improvement of the coding efficiency, especially at low bitrates.
2.4.2.3 EBCOT and JPEG2000

EBCOT achieves higher compression efficiency than both EZW and SPIHT in general by paying the price of much higher computational complexity. It also achieves embedded coding by packing the more important information first in the bitstream. However, EBCOT does not use the self-similarity in the zerotree structure as EZW and SPIHT does. Instead it partitions the transform coefficients into code blocks and then codes each code block independently. Its success is due to the utilization of the statistical redundancy among neighboring coefficients and across different bitplanes within the same code block. The coding of each code block is based on an 18 contexts model using the MQ coder as the coding engine (D.Taubman 2000). The EBCOT also adopts a Post Compression Rate Distortion Optimization (PCRD) in conjunction with tier 2 coding (D.Taubman 2000) to finally determine the output bit stream. EBCOT has been adopted by JPEG2000 (JTC1/SC29 2004) as the kernel component with some modification. Since JPEG2000 compression scheme has been modified and extended for the research in this thesis, its working mechanism is explained in detail in Appendix A.

The order of the aforementioned coding primitives is non-trivial. It has impact on the rate vs. distortion (R-D) optimization performance of the EBCOT coding engine. The principle of the order sounds fairly simple: coding more important information before less important information. Normally the measure of ‘importance’ is the statistical measure such as MSE. An illustration of the R-D optimization curve is drawn in Fig. 2.14 where the measure of rate is bits per pixel.
In Fig. 2.14, the dashed line is the ideal R-D curve. The use of truncated optimization brings sub-optimal sections which are drawn in solid line. It is straightforward to see that the slopes of a, b, and c are higher than those of e, f, and g. Assuming $m$ is used to denote the slope, the relationship of slopes of different sections is represented by (2.23).

$$m_a \geq m_b \geq m_c \geq \cdots \geq m_e \geq m_f \geq m_g$$  \hspace{1cm} (2.23)

Therefore for the same increase amount of rate, the decreases of distortions of these sections have the relationship in (2.24).
2.5 Summary

The theoretical foundation which the work in this thesis is built on is presented in this chapter. First, the general structure of traditional video coding is introduced. It encompasses the Predictive/Transform/Subband Coding, the quantization, the entropy coding, the context design, and the MQ coder. Second, the DWT theory is discussed, which includes the traditional DWT, the lifting implementation of DWT, and the more recent ADL scheme. Third, the MC schemes in the DWT domain are presented. Among them the WTMC and the LBS are used in this thesis work. Fourth, the bitplane based image/video coding schemes are discussed. The review of the EBCOT context design is provided in Appendix A. While the most relevant techniques of the DWT based video coder are highlighted in this chapter, a more comprehensive review of the DWT video coder in the literature is given in Appendix B.
Chapter 3

A DWT/MC/DPCM Coding Framework

A novel video coder employing a Discrete Wavelet Transform (DWT) / Motion Compensation (MC) / Differential Pulse Code Modulation (DPCM) framework and the EBCOT as the coding engine is introduced in this chapter. Although the DCT/MC/DPCM framework has been adopted by the current industry and international video compression standards (ISO/IEC 1993; ITU-T 1993; ISO/IEC 2000; ITU-T 2000; ISO/IEC 2004; ISO/IEC 2005; ITU-T 2005), the DWT based video coders have been extensively investigated since almost two decades ago (G.Karlsson and M.Vetterli 1988; A.S.Lewis and G.Knowles 1990). Among various DWT video coders in the literature\(^2\), the work of the DWT/MC/DPCM coding structure can be traced back to Zhang and Zafar (Y.-Q.Zhang and S.Zafar 1992) and the investigation is still ongoing (F.G.Meyer, A.Averbuch et al. 1997; D.Blasiak and W.-Y.Chan 1998; S.Kim, S.Rhee et al. 1998; H.W.Park and H.S.Kim 2000; G.Heising, D.Marpe et al. 2001). Before proceeding with discussions of the proposed video coder, it is of importance to compare the DCT and the DWT to highlight their advantages and disadvantages in image/video compression. Table 3.1 provides a comparison of these two transforms. The DWT has multiresolution structure, therefore it allows for resolution scalability. The computation complexity of DWT could be a slight disadvantage. However, this is a relatively trivial problem with

\(^2\) A comprehensive review of the DWT based video coder is given in Appendix B
the computation power of commercially available hardware. The artifacts generated by
the DWT and the DCT are essentially different. For DCT based video coders, the so-
called blocking artifact is the most prominent artifact (M.Yuen and H.R.Wu 1998). For
DWT based video coders, blurring and ringing are the dominant artifacts. Both DCT and
DWT have fast algorithms (W.-H.Chen, C.H.Smith et al. 1977; M.J.Narasimha and
I.Daubechies and W.Sweldens 1998; Y.Jeong, I.Lee et al. 1998; Mallat 1999). They all
support scalability in generated bitstream therefore allow decoding at various
temporal/spatial resolutions or quality levels (A.Secker and D.Taubman 2003; A.Secker
DCT based video coders support layered scalability such as the SVC (Scalable Video
Coding) extension of the H.264/MPEG-4 AVC video compression standard (H.Schwarz,
D.Marpe and T.Wiegand 2007). The Fine Granular Scalability MPEG-4 coder (W.Li
2001), nonetheless, adopts a bitplane based coding paradigm and therefore offers graceful
scalability. The DWT based video coders, on the other hand, can provide fine scalability
with 3-Dimensional (3-D) DWT (J.M.Shapiro 1993; B.-J.Kim, Z.Xiong and
Visual System is also an issue needs to take into account (H.R.Wu and K.R.Rao 2005)
when comparing these two technologies.

The contributions of the proposed coding framework in this chapter are twofold. First, the
EBCOT, which employs a built-in Multi-Contexts based Adaptive Binary Arithmetic
Coding structure (MCABAC), is adopted as the coding engine for both the intra- and the
inter-frame coding. EBCOT has proven to provide good compression efficiency and embedded rate-distortion (R-D) optimization mechanism (D. Taubman 2000). The use of EBCOT here is an extension of the EBCOT application from still image to video compression domain. Second, this framework offers a good platform for the Perceptual

<table>
<thead>
<tr>
<th>Characteristics</th>
<th>DCT</th>
<th>DWT</th>
</tr>
</thead>
<tbody>
<tr>
<td>Multiresolution - Capable of capturing details in both spatial and frequency domain</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Computation Window on Each Frame</td>
<td>Normally 8×8</td>
<td>Entire image or tile</td>
</tr>
<tr>
<td>Artifacts</td>
<td>Mainly blocking artifacts along the DCT block boundaries and others (M. Yuen and H. R. Wu 1998)</td>
<td>Mainly blurring and ringing as well as temporal fluctuation (for video)</td>
</tr>
<tr>
<td>Fast algorithm</td>
<td>Available</td>
<td>Available</td>
</tr>
<tr>
<td>Support for fine scalability</td>
<td>Available since the introduction of FGS in MPEG-4</td>
<td>Widely available</td>
</tr>
<tr>
<td>Adoption by video compression standards</td>
<td>By all video compression standards</td>
<td>MPEG-4 VTC only</td>
</tr>
</tbody>
</table>

Table 3.1 Comparison between DCT and DWT

Distortion Measures (PDM) based on the Human Visual System (HVS) where the Mean Squared Error (MSE) can be easily replaced with the PDM in the R-D optimization. The performance of the proposed coding framework is compared with H.264 Baseline Profile (BP). The results demonstrate that the proposed coder outperforms the benchmarks in terms of R-D performance on the sequence Football and Susie. More advanced motion compensation scheme such as the LBS and redesigned contexts for the adaptive binary
arithmetic coding of the inter-frame data are promising approaches to further improve the rate-distortion (R-D) performance, as will be discussed in the following chapters.

The outline of this chapter is as follows: Section 3.1 introduces the proposed video coding framework. Section 3.2 presents the adaptive frame/field coding mode based on this framework. The experimental results are demonstrated in Section 3.3. A summary is drawn in Section 3.4.

3.1 The Proposed DWT/MC/DPCM Video Coder

The block diagram of the proposed video coder is shown in Fig. 3.1. The dyadic DWT is performed \textit{a priori} on the input pictures under Mallat’s decomposition framework (S.G. Mallat 1989). Then the wavelet transformed images fall into the category of either intra- or inter-frames. The biorthogonal 9/7 filter-bank (A. Cohen, I. Daubechies and J.C. Feauveau 1992; M. Antonini, M. Barlaud, P. Mathieu et al. 1992) is used as the DWT kernel because of the reasons pointed out by Watson (A.B. Watson, G.Y. Yang et al. 1997). First, the biorthogonal 9/7 filters are linear phase filters therefore there is no need for phase compensation when they are used in cascade. Second, they are widely used and have been adopted by JPEG2000 still image compression standard (JTC1/SC29 2004) and FBI fingerprint image compression standard. Third, they have been argued to have certain attractive mathematical properties for image compression. In comparison with JPEG2000 still image compression standard (JTC1/SC29 2004), where the number of Mallat decomposition level is 5, the proposed video coder employs a three-level DWT. In the experiments it is found five-level decomposition does not bring significant more
benefits than three-level decomposition for CCIR-601 NTSC and PAL video formats with respect to coding efficiency. For each level of DWT analysis, 1-D DWT filtering is followed by 2:1 decimation along each spatial direction (M.Antonini, M.Barlaud, P.Mathieu et al. 1992). For the 1st level of DWT analysis, the input is the image itself, and the outputs are four new subbands with half of the spatial resolution of the input along each direction.

(a) System Diagram
CHAPTER 3 A DWT/MC/DPCM CODING FRAMEWORK

(b) Intra-frame Coder

(c) Inter-frame Coder

Fig. 3.1  Block Diagrams of the Proposed Video Coder
These four subbands are the horizontal low - vertical low (LL) band, the horizontal high - vertical low (HL) band, the horizontal low - vertical high (LH) band and the horizontal high - vertical high (HH) band. They are denoted as $LL^i, HL^i, LH^i$ and $HH^i$ in the Mallat decomposition framework, where $i \geq 0$, which is the index of the level (0 being the index of the original image). For the following levels of DWT analysis, the input is $LL^{i-1}$, which is the LL band produced by the previous level of DWT analysis. After $n$ levels of decomposition, there are 1 LL band, $n$ HL bands, $n$ LH bands and $n$ HH bands. The size of each individual subband is described by

$$W_{\text{band}} = 2^{-i} \cdot W_{\text{original}}, \quad H_{\text{band}} = 2^{-i} \cdot H_{\text{original}}$$ (3.1)

where $W_{\text{band}}$, $W_{\text{original}}$, $H_{\text{band}}$, $H_{\text{original}}$ and $n$ represents for the width of the subband, the width of the input image, the height of the subband, the height of the input image and the pyramidal level index, respectively.

The intra-frame DWT coefficients are quantized by a scalar quantizer first. Then the quantized DWT coefficients are encoded by the EBCOT (D.S.Taubman and M.W.Marcellin 2002) coding engine as in JPEG2000. A brief description and the detail of the context design of EBCOT are in Chapter 2 and Appendix A, respectively. In order to make the content of this chapter self-contained, the principle of EBCOT is briefly reviewed here. Under EBCOT scheme, each subband is firstly partitioned into code blocks with the nominal size of either 64 by 64 or 32 by 32 before encoding. The choice of code block size is a tradeoff between the coding efficiency and the scalability. Then the
coefficients within each individual code block are reorganized into stripes. The scanning orders of the stripes determine the visit order of data at each bitplane. EBCOT shares the idea of embedded image coding with the earlier EZW scheme (J.M.Shapiro 1993) and the SPIHT scheme (A.Said and W.A.Pearlman 1996). It is generally a bitplane quantizer with embedded entropy coder. EZW and SPIHT utilize the statistical redundancy among DWT coefficients at different decomposition orientations and levels by the construction of the Zero-tree (ZT) data structure. In contrast, EBCOT only considers the statistical redundancy existing within each individual code block despite its overall superior performance over the original EZW and SPIHT work. EBCOT is a two-tier coding engine. It is composed of a bitplane based entropy coder, an R-D optimizer and a bitstream packer. The bitplane entropy coder belongs to tier 1, which carries out MCABAC at various bitplanes. In fact, for each individual code block, the encoder visits the data from the bitplane associated with the Most Significant Bit (MSB) downwards to the one associated with the Least Significant Bit (LSB). Within each individual bitplane, three sub-bitplane passes are adopted instead of just one bitplane coding pass. Each coding pass contains a portion of the total 18 coding context set. A main advantage of this approach is the near-optimal embedding as explained previously by Section 2.4.2.3, where the information that results in the largest reduction in the distortion for the smallest increase in the bitstream size is coded first. That is to say, more important information is always coded prior to less important information. The three coding passes are the Significance Propagation Pass (SPP), the Refinement Pass (RP) and the Cleanup Pass (CP). Every bit to be encoded is assigned to a coding context according to certain causal information such as the significant information of the current bit’s neighbors, the sign if
the significant bit and so on. All these coding contexts share the same BAC coder, which is the so-called MQ-coder (D.S. Taubman and M.W. Marcellin 2002) that was regarded as the state of the art of BAC. The decoder replicates the data by 1) following the same visiting sequence and context decision principles and 2) using the MQ-decoder. As can be seen from Fig. 3.1(b), the output of EBCOT in the intra-frame coder is sent to two recipients. The first one is the intra-frame bitstream and the other one is the built-in decoder in the intra-frame encoder. The data sent to the second recipient is decoded, dequantized and fed to the inter-frame coder.

It can be seen from Fig. 3.1(c) that the inputs of the inter-frame coder come from two sources: one is the intra-frame coder and the other is the DWT data from the input picture(s). The data from the first source is named as the reconstructed DWT data (RDWTD) and the one from the second source is named as the original DWT data (ODWTD). The ME and MC are performed between the DWT data from these two sources. The wavelet tree (WT) based ME/MC is employed in the work of this chapter while the Low-Band-Shift (LBS) based ME/MC will be discussed in Chapter 4. An illustration of the form of a WT can be found in Fig. 2.5 while the ME based on WT is formulated by (2.21). In the work of this thesis, the size of the node (coefficient block) at the lowest resolution level, i.e. level 1, is $2 \times 2$. Since a dyadic DWT is used, the size of the coefficient block at each subsequent higher resolution level is increased by a factor of 2. Therefore, at level 2 and 3, the size of the coefficient block is $4 \times 4$ and $8 \times 8$, respectively. These related coefficient blocks are grouped together into a wavelet block (WB) with a size 16 by 16. Both the WB and the WT are essentially descriptions of the
same set of DWT coefficients from different perspectives. The value of \( \Delta m \) or \( \Delta n \) in (2.21) is within the search window of the motion vector (MV) (or more precisely the motion information), which is \([-4, +4]\) along the horizontal and vertical direction at the lowest resolution, respectively. As a result, the alphabet to be coded consists of 9 symbols as formulated by (3.2),

\[
symbol = |mv| \cdot 2 - \text{sign}(mv) \tag{3.2}
\]

where \( \text{sign}(mv) = 1 \) when \( mv > 0 \) and \( \text{sign}(mv) = 0 \) when \( mv \leq 0 \). As in Fig. 3.1(c), the ME is carried out between RDWTD and ODWTD to lock the best matching wavelet tree in the reference picture. The WT in the reference frame/field with the smallest SAD is chosen as the matching WT for MC. Witten et al’s AAC algorithm has been adopted (I.H.Witten, R.M.Neal and J.G.Cleary 1987) to code the resulting motion vector in the ME because it provides clear separation between the coding engine and adaptive model in the software implementation. Two identical yet independent adaptive models are used to model the statistics of the horizontal and vertical component of the motion information, respectively. After MC, the residue image is derived by subtracting the motion compensated RDWTD from the ODWTD. Then it is sent to the scalar quantization and the entropy coder. The EBCOT coding engine was used for inter-frame entropy coding with the same set of contexts as in intra-frame coder. As will be seen in Chapter 5, the context set for the EBCOT inside the inter-frame coder needs to be calibrated because the statistical behavior of the inter-frame data is different with that of the intra-frame data. A redesign of the context set is needed if it is not optimal in terms of
inter-frame data encoding.

3.2 Interlaced Video Coder

Most existing DWT based video coders operate at low to very low bit rates and are intended for video conferencing and mobile telephony applications. These applications generally code digital videos in the frame mode. However, there are other applications, such as digital TV, where the scene is taken with TV camera therefore signals in each picture frame consist of two fields. If the video coder only adopts frame mode to code these TV pictures, the so-called interlacing artifacts will consume unnecessary bits for the resulting high spatial frequency information. In fact, interlaced material is still widely used in the world and will not disappear in the next few years (E.François, J.Viéron et al. 2007). Therefore even the new scalable video coding (SVC) has worked on the support of interlaced coding (H.Schwarz, D.Marpe and T.Wiegand 2007). However, most DWT based video coders in literature do not aim at the coding of TV signals which contain interlacing artifacts. Since improved R-D performance is the main goal in this thesis work, a frame/field adaptive mode is introduced based on the framework presented in Section 3.1. The DWT is performed on the input frame or individual field depending on whether it is in frame or field mode as illustrated in Fig. 3.2. The choice of coding mode is due to the following fact: although the field mode may improve the compression efficiency of the data, it may also impede the overall compression due to the header of the second picture field. The allocation of bits is a tradeoff between higher compression ratio of the data and the overhead. When the frame mode is chosen, the whole picture is coded exactly as the video coder in the last section. Otherwise, the even and odd fields
are coded separately.

When the current picture is coded in the frame mode, the WBs in the reference frame are compared with the current WB within search window. When the current picture is coded in the field mode, the ME, MP and MC are much more complex as listed below:

1) for the top field in the current picture, the reference field could be the top or bottom field in the reference picture;

2) for the bottom field in the current picture, the reference field could be the top or bottom field in the reference picture, or the reconstructed top field in the current picture.
3.3 Experimental Results

The average PSNR results of the proposed coder are listed in Table 3.2 with the following experimental configurations. The original video materials are in monochrome NTSC format at 30 frames/s. The size of the pictures is chopped to $704 \times 480$ for the convenience of DWT decimation. The sequences *Football*, *Susie*, *Mobile&Calendar*, and *Baloon&Pops* are used and compared with the benchmark, which is the H.264 Baseline Profile (BP). The H.264 BP has been chosen because this profile accepts I and P slices but not B slices and the proposed coder currently supports I- and P-frames only. The source code of the proposed video coder is written in ANSI C++ and compiled with Microsoft® Visual C++ .NET 2003 edition. The most recent H.264/AVC reference software release JM14.0 is used to generate the results of H.264 BP. The programs run on a 32 bits computer equipped with Intel® Core™ 2 CPU (4300@1.80GHz), 2GB main memory, 250GB 7200rpm hard drive and Windows XP SP2 professional edition.

For *Football* and *Susie* sequence, the proposed coder (frame mode) obviously outperforms H.264 BP, with 0.4-1dB higher in PSNR values. For *Mobile&Calendar* and *Baloon&Pops*, most time the PSNR values of the coded videos generated by the proposed coder (frame mode) are inferior to those generated by H.264 BP. In *Mobile&Calendar*, the PSNR values of the proposed coder (frame mode) are 1.8-3.8dB lower than those of H.264 BP. In *Baloon&Pops*, however, the PSNR values of the proposed coder (frame mode) are 0.3-1.1dB higher than those of H.264 BP at the bitrate of 1.5Mbits/s to 2.5Mbits/s. They are inferior to H.264 BP, in terms of average PSNR values, at the bitrate between 3Mbits/s to 5Mbits/s. From the reconstructed images illustrated in Fig. 3.3, the
blocking artifacts do not appear as other DWT based video coders. The proposed video
coders can work under field and adaptive mode as discussed in Section 3.2. Their
dynamic PSNR performances are illustrated in Fig. 3.4, from which the PSNR values of
the adaptive mode are always the higher one among the frame and field mode for each
individual frame.

<table>
<thead>
<tr>
<th>Sequence</th>
<th>Football</th>
<th>Susie</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bitrate (Mbits/s)</td>
<td>Proposed Coder (Frame)</td>
<td>Proposed Coder (Field)</td>
</tr>
<tr>
<td>1.5</td>
<td>30.37</td>
<td>30.62</td>
</tr>
<tr>
<td>2</td>
<td>31.50</td>
<td>31.68</td>
</tr>
<tr>
<td>2.5</td>
<td>32.40</td>
<td>32.51</td>
</tr>
<tr>
<td>3</td>
<td>33.14</td>
<td>33.22</td>
</tr>
<tr>
<td>3.5</td>
<td>33.81</td>
<td>33.87</td>
</tr>
<tr>
<td>4</td>
<td>34.47</td>
<td>34.53</td>
</tr>
<tr>
<td>4.5</td>
<td>35.07</td>
<td>35.09</td>
</tr>
<tr>
<td>5</td>
<td>35.63</td>
<td>35.62</td>
</tr>
</tbody>
</table>

(a)  Football and Susie

<table>
<thead>
<tr>
<th>Sequence</th>
<th>Mobile &amp; Calendar</th>
<th>Baloon &amp; Pops</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bitrate (Mbits/s)</td>
<td>Proposed Coder (Frame)</td>
<td>Proposed Coder (Field)</td>
</tr>
<tr>
<td>1.5</td>
<td>20.14</td>
<td>18.61</td>
</tr>
<tr>
<td>2</td>
<td>20.68</td>
<td>19.08</td>
</tr>
<tr>
<td>2.5</td>
<td>21.14</td>
<td>19.84</td>
</tr>
<tr>
<td>3</td>
<td>21.69</td>
<td>20.33</td>
</tr>
<tr>
<td>3.5</td>
<td>22.40</td>
<td>20.78</td>
</tr>
<tr>
<td>4</td>
<td>22.81</td>
<td>21.25</td>
</tr>
<tr>
<td>4.5</td>
<td>23.26</td>
<td>21.85</td>
</tr>
<tr>
<td>5</td>
<td>23.74</td>
<td>22.45</td>
</tr>
</tbody>
</table>

(b)  Mobile&Calendar and Baloon&Pops

Table 3.2 Average PSNR Results
(a) Football coded by the proposed coder at 1.5Mbits/s in frame mode

(b) Football coded by the proposed coder at 3Mbits/s in frame mode
(c) Football coded by the proposed coder at 4.5Mbits/s in frame mode

(d) Susie coded by the proposed coder at 1.5Mbits/s in frame mode
(e) Susie coded by the proposed coder at 3Mbits/s in frame mode

(f) Susie coded by the proposed coder at 4.5Mbits/s in frame mode

Fig 3.3 Reconstructed Images Coded by the Proposed Coder
(a) Football coded at 1.5Mbits/s

(b) Football coded at 3Mbits/s
(c) Football coded at 4.5Mbits/s

(d) Susie coded at 1.5Mbits/s
(e) Susie coded at 3Mbits/s

(f) Susie coded at 4.5Mbits/s

Fig. 3.4 Dynamic PSNR Results
(note: the results of the proposed video coder in frame, field, and adaptive mode are represented by triangles, plus, and cross signs, respectively while those of H.264 BP are denoted by dots)

3.4 Summary

A DWT/MC/DPCM video coding framework is proposed in this chapter, which adopts EBCOT as the coding engine for both intra- and inter-frames. The results show that the proposed coder outperforms H.264 BP for Football and Susie in terms of PSNR values by 0.4-1dB in frame coding mode. For Mobile&Calendar, the PSNR values of the coded videos generated by the proposed coder (frame mode) are always inferior to those generated by H.264 BP. For Baloon&Pops, however, the PSNR values of the proposed coder (frame mode) are 0.3-1.1dB higher than those of H.264 BP at the bitrate of 1.5Mbits/s to 2.5Mbits/s. They are inferior to H.264 BP, in terms of average PSNR values, at the bitrate between 3Mbits/s to 5Mbits/s. The reason why the proposed video coding framework outperforms H.264 BP for some sequences and not for the others is two-fold. First, the motions of Football and Susie are relatively simple. Football presents translational motion and Susie is a head-shoulder type sequence. The simple WT based MC can deal with these two sequences quite well. In contrast, Mobile&Calendar and Baloon&Pops contain more complicated motion and the simple WT based MC cannot handle the so-called shift-variant problem (H.W.Park and H.S.Kim 2000). Second, the context set of the inter-frame coder needs to be optimized to efficiently code the residual information of inter-frames. From the reconstructed video frames illustrated in Fig. 3.3, blocking artifacts do not appear as they normally do in DCT based video coders (M.Yuen
and H.R.Wu 1998). The dynamic PSNR performances of the proposed coder in the frame mode, field mode, and adaptive mode are illustrated in Fig. 3.4, from which the PSNR values of the adaptive mode are always the higher one among the frame and field mode for each individual frame. More complicated ME/MC and re-designed inter-frame contexts are promising approaches to further improve the R-D performance of the proposed coder, which will be discussed in the following chapters.
Chapter 4

Motion Compensation in DWT Domain

A DWT/MC/DPCM based video coding framework was proposed in Chapter 3. The potential improvement for the inter-frame coder of the proposed coding framework is generally composed of two components: the MC in the DWT domain in conjunction with the relevant motion information coding, and the inter-frame data (residue information) coding. The first component is addressed in this chapter with the following outline. An introduction is given in Section 4.1 followed by the review of the MC techniques in DWT domain in Section 4.2. Section 4.3 presents the application of the LBS scheme for the MC in the proposed DWT/MC/DPCM video coder. Section 4.4 discusses the context design for the motion information coding based on MQ coder. A summary is drawn in Section 4.5.

4.1 Introduction

Digital video is composed of consecutive pictures along the temporal direction. There is strong temporal redundancy existing between neighboring frames. A visual example of the temporal redundancy is illustrated in Fig. 4.1 where a difference picture in Fig. 4.1(c) is calculated by simply taking the difference between the previous and current frames followed by normalization as given by
\[ \hat{\delta}(m, n) = \left( \frac{\delta(m, n) - \min(\delta(m, n))}{\max(\delta(m, n)) - \min(\delta(m, n))} \right) \times 255 \] (4.1)

where \( \delta(m, n) \) represents for the difference of the pixel values at the spatial location \((m, n)\), \(\max(\cdot)\) and \(\min(\cdot)\) are the maximum and minimum operators, respectively. Here the images are monochrome and the pixel values are ranged within \([0, 255]\).

It is therefore straightforward to see that MC in the spatial domain removes spatiotemporal redundancies. In fact, MC has been combined with the DCT to form the so-called hybrid video coding framework (K.R. Rao and J.J. Hwang 1996). Within this framework, the MC is applied to the original video frame prior to the DCT operation. After the DCT, the transformed residue is quantized and entropy coded. This DCT/MC/DPCM framework proves to be highly efficient in terms of Rate-Distortion (R-D) performance (B.-J. Kim, Z. Xiong and W.A. Pearlman 2000). However, it is also well-known that the so-called blocking and ringing artifacts exist within this coding framework, especially at low and very low bitrates, which is very unpleasant to view for human viewers (M. Yuen and H.R. Wu 1998). DWT/MC/DPCM video coding framework is among the alternatives to the DCT/MC/DPCM structure, where MC is performed in the DWT domain before DPCM encoding. A DWT/MC/DPCM video coding framework based on EBCOT has been proposed in Chapter 3 and compared with H.264 Baseline Profile (BP) with respect to the PSNR results. The results demonstrate that the proposed video coder is a competitive coding structure with two additional advantages. First, the
so-called blocking artifacts do not appear in the coded video. Second, the rate-distortion optimization in the built-in EBCOT coding engine provides finer scalability so that ‘more important’ bits are positioned prior to ‘less important’ bits in the bitstream. One of the most widely used measure for the ‘importance’ is the Mean Squared Error (MSE) which is a statistical measure. The MSE is also used in the EBCOT coding engine by default. As has been proved earlier (D.M.Tan 2002), a distortion measure based on a vision model can be employed to replace the MSE as the measure of the ‘importance’.

(a) Football No.51 Frame
Fig. 4.1 Illustration of Spatiotemporal Redundancy

(b) Football No.52 Frame

(c) Difference Image between (a) and (b)
There exist obvious disadvantages in the proposed video coders. First, the direct Wavelet Tree (WT) based ME/MC does not take the so-called shift-variant problem into account, which impedes ME in the DWT domain (H.W.Park and H.S.Kim 2000). The shift-variant problem occurs in a system employing complete wavelet transform such as the Mallat decomposition (Mallat 1999), where the wavelet coefficients are decimated by a factor of 2 after each DWT filtering operation. When an object in the reference image is shifted by odd number of pixels, there may exist very large differences between the wavelet coefficients of the reference image and the odd-pixel-shifted image (P.-Y.Cheng, J.Li et al. 1995; H.W.Park and H.S.Kim 2000) in the high-frequency subbands. These large differences in the high frequency subbands thus make the prediction of the wavelet coefficients fairly difficult and imprecise. Second, the coding of the motion vectors (MVs or more precisely, the motion information) needs more investigation as well.

4.2 Motion Compensation in DWT Video Coder

4.2.1 Traditional Motion Estimation and Compensation

Among the most popular motion estimation techniques are the block-matching algorithm (BMA) and the so-called pel-recursive technique (S.Kim, S.Rhee, J.G.Jeon et al. 1998). In BMA, an image is partitioned into motion blocks, and the same motion information, or motion vector, is assigned to all pixels belonging to the same motion block. In the translational motion model, it is assumed that an image consists of rigid objects with translation motion only (K.R.Rao and J.J.Hwang 1996) although the motion of a real object in the world could be fairly complex (T.Driemeyer 2000). In BMA, for the block
in the current frame at time $t$, a block in the previous frame at time $t-1$ within search window $\Omega$ is found, which has the minimal value of the matching criterion. A few designations are defined here before further discussion: 1) the pixel location $c = (m,n)$; 2) the motion vector $v = (v_x, v_y)$ representing the object/block motion from $t-1$ to $t$, where $v_x$ and $v_y$ are the translations in the horizontal and vertical directions, respectively, in terms of pixels. The motion vector computation can be generalized as in (4.2).

$$v = \arg \min_{v \in \Omega} f(s)$$  \hspace{1cm} (4.2)

where $f(\cdot)$ represents for the matching criterion. The most widely used matching criteria are the Maximum Cross Correlation (MCC), the Minimum Mean Squared Error (MMSE), and the Minimum Absolute Difference (MAD).

### 4.2.2 Motion Estimation and Compensation in DWT Video Coding

Zhang et al. investigated the Motion Compensation (MC) in the DWT domain (Y.-Q.Zhang and S.Zafar 1992). That is among the earliest work which reported the MC in the DWT domain. In (Y.-Q.Zhang and S.Zafar 1992), the so-called variable block-size multi-resolution motion compensation (MRMC) scheme was proposed and tested. More variants of MRMC were reported afterwards (S.Zafar and Y.-Q.Zhang 1993; X.Yang and K.Ramchandran 2000). All these schemes fall into the category of DWT/MC/DPCM approach because DWT is performed before the motion compensation.
In addition, there is another approach which performs MC in the original pixel domain followed by DWT based coding. This approach can be named as MC/DWT/DWPM scheme. The work of (D.G.Sampson, Silva et al. 1995) is among the earliest work of MC/DWT/DWPM category where an Overlapped Block Matching (OBM) algorithm is used for MC in the pixel domain followed by the DWT and the vector quantization. Cheng et al. (P.-Y.Cheng, J.Li and Kuo 1995) noticed the so-called shift-variant property due to the down- and up-sampling operation in the wavelet analysis and synthesis process, respectively. They proposed a multi-scale DWT domain based Motion Compensation scheme trying to address this issue. Unfortunately, only the mathematical framework and model were provided in that work. Ito et al. introduced a novel approach where the Motion Estimation (ME) is performed in the original pixel domain followed by the DWT, then the MC and coding are performed in the DWT domain as in (H.Ito and N.Farvardin 1995). However, the authors did not reveal more details of their implementation. The authors of (D.Lazar and A.Averbuch 2001) proposed a video coder based on optimal vector bit-allocation. For videos with small motions, still image coder was used for each individual picture while MC was used if large motion is detected. The MC/DWT/DPCM framework can be also found in the work of (S.A.Martucci, I.Sodagar et al. 1997; J.Y.Tham, S.Ranganath et al. 1998; D.Marpe and H.L.Cycon 1999) where the OBMC/DWT approach was employed. In (S.A.Martucci, I.Sodagar, T.Chiang et al. 1997) an adaptive DWT is used. For higher decomposition levels, filters with longer length are used in order to avoid the artifacts. While for lower decomposition levels, shorter length filters are used. EZW and its variants were proposed for the quantization and finally, the adaptive arithmetic coding was used for lossless entropy coding. An
approach based on picture segmentation was suggested in (V.Silva and Sa 1994). A MC/DWT wavelet video coding algorithm and an optimization framework that allocated bits efficiently among consecutive frames at the pixel level were presented in (K.K.Lin and R.M.Gray 2004). Both the intra- and inter-frame coders were based on SPIHT with some modification. More importantly, the so-called dependent optimization was thoroughly investigated. The results outperformed H.263 in respect of PSNR for more than 1dB. Heising et al. (G.Heising, D.Marpe, H.L.Cycon et al. 2001) suggested a very low bit-rate video coding scheme which combined new ideas in both ME, wavelet filter design and wavelet-based coding. MC based on image warping and overlapped block motion compensation was proposed in that work to reduce the temporal redundancies. This combined motion model had the advantage of representing more complex motion than the simple block matching schemes. Spatial decorrelation of the motion compensated residual images was performed using a biorthogonal infinite impulse response (IIR) wavelet filters with the highly efficient pre-coding scheme of ‘partitioning, aggregation and conditional coding’ (PACC). The experimental results demonstrated significant improvements of 1-2.3dB PSNR in comparison with the H.263+ TEST MODEL tmn10. In addition, the author’s intra-coding method provided a performance gain of 0.5dB PSNR on the average when compared to standard JPEG2000.

Since the work of (A.S.Lewis and G.Knowles 1990), there have been more 3D based works appearing in the literature such as (H.Ito and N.Farvardin 1995; J.Y.Tham, S.Ranganath and A.A.Kassim 1998; D.Marpe and H.L.Cycon 1999). In (D.Marpe and H.L.Cycon 1999) a motion compensated 3D video codec was proposed for very low
video compression. The work in (N.B.Karayiannis and Y.Li 2001) used the same general signal decomposition scheme. Vector quantization and replenishment were used after transform. Symmetric vector code book quantization was considered under a framework of 3D wavelet video coder in (I.K.Levy and R.Wilson 2001). 3D-EZW and 3D-SPIHT was proposed in (Y.Chen and W.A.Pearlman 1996) and (B.-J.Kim and W.A.Pearlman 1997), respectively. As there was no motion estimation/compensation in (Y.Chen and W.A.Pearlman 1996; B.-J.Kim and W.A.Pearlman 1997), these work were not susceptible to temporal propagation errors. Vector quantization technique was used in (N.B.Karayiannis and Y.Li 2001) where the employment of replenishment of code-vector labels brings the reduction of the inter-frame bitrate. The proposed video compression system produced comparable or better results than the H.261. A scheme of Spatial and Temporal Tree Preserving 3-D SPIHT (3D-SPIHT) was reported in (S.Cho and W.A.Pearlman 2002) for a more robust 3-D wavelet coder against channel errors with little increase in the complexity and little loss in noiseless channel performance. The PSNR results outperformed MPEG-2. An optimal 3-D coefficient tree structure for 3-D wavelet coding was suggested in (C.He, J.Dong et al. 2003). It was found that asymmetric 3-D wavelet tree structure could produce a higher compression ratio than traditional symmetric approaches. The authors applied the new tree structure to the state-of-the-art 3-D SPIHT video coder and showed that it brought convincing PSNR improvement for all the tested video sequences without arithmetic coding involved. A vector SPIHT was proposed in (D.Mukherjee and S.K.Mitra 2003) and it offered a better PSNR result than H.263. An innovative temporal wavelet filtering scheme called Sub-Grouping Transformation (SGT) was proposed in (Z.Gao and Y.F.Zheng 2005) for 3D
video coding. It was intended to address the ghost effect caused by large temporal distortions at very low bit-rate video compression. Furthermore, by splitting high motion coefficient arrays into smoother sub-groups, the proposed algorithm could achieve higher compression ratio at the same time. More efforts have been invested for wavelet transform tool itself such as the work in (R.Kutil 2003) where an anisotropic 3-D Wavelet Packet Bases are proposed for video coding purpose. The authors reported that there existed a fixed anisotropic wavelet packet basis that was able to outperform MPEG-4 by up to 4dB especially for video sequences with low motion.

In addition to the above work, there have been a few other approaches such as the work in (R.Dianat, M.Ghanbari et al. 2005) where DWT based coder was used for Intra-frame coding and DCT based techniques were used for inter-frames. This hybrid coding framework demonstrated a better PSNR values than the standard H.263. In addition to the framework, there have been quite a few contributions for different working components of the DWT based video coders such as the work in (Auwera, A.Munteanu et al. 2002). A bottom-up motion compensated prediction scheme was proposed in (Auwera, A.Munteanu, P.SCHELKENS et al. 2002) which overcame the periodic shift-variance of the DWT and was formalized into prediction rules using filtering operations. An optimal wavelet filter design was discussed in (M.Li and T.Nguyen 2005). It was proved that the proposed filter could compensate the shortcomings of MPEG filter and Daubechies 9/7 filters for the application of scalable video coding. A model based rate allocation scheme was proposed in (Y.-H.Yu and C.-J.Tsai 2005) for generic wavelet image and video embedded coding. The simulation results demonstrated that the new scheme could
CHAPTER 4 MOTION COMPENSATION IN DWT DOMAIN

provide a more accurate rate than the traditional bisection method.

From all the above work especially those in recent years, it is obvious that there have been much more efforts contributed to 3-D DWT video coding than 2-D approaches no matter they are MC/DWT/DPCM or DWT/MC/DPCM schemes mainly due to the increasing demands of video streaming over computer networks and video database browsing. These applications require the following properties of the video coders such as low computation complexity, scalability in addition to a good R-D performance. However, the R-D performance of MC-2D based DWT video coder still outperform 3D DWT coder because the MC is more efficient than the temporal filtering to remove the spatiotemporal redundancy. On the other hand, the DWT coder applies the transform to the whole image therefore eliminates the so-called blocking artifacts, which usually occur along the boundaries between pixel blocks. Therefore the MC-2D based DWT coder is suitable for applications with high demand for R-D performance both subjectively and objectively.

4.3 Application of LBS into DWT/MC/DPCM Coder

Park and Kim (H.W.Park and H.S.Kim 2000) proposed the LBS scheme as a wavelet based MC to overcome the shift-variant problem of the complete DWT. For details of LBS the readers are referred to Section 2.3.1 or (H.W.Park and H.S.Kim 2000). The PSNR results of the proposed video coder adopting LBS are illustrated in Table 4.1, where they are compared with the results of the proposed coder adopting WTMC scheme. The overall improvement in terms of PSNR value is quite significant. Four sequences are
used in the experiment. They are Football, Susie, Baloon & Pops, and Mobile & Calendar.

<table>
<thead>
<tr>
<th>Video Sequence</th>
<th>Football</th>
<th>Susie</th>
<th>Baloon &amp; Pops</th>
<th>Mobile &amp; Calendar</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bitrate (Mbits/s)</td>
<td>No LBS</td>
<td>With LBS</td>
<td>No LBS</td>
<td>With LBS</td>
</tr>
<tr>
<td>1.5</td>
<td>30.37</td>
<td>30.29</td>
<td>36.24</td>
<td>36.34</td>
</tr>
<tr>
<td>2</td>
<td>31.50</td>
<td>31.44</td>
<td>37.21</td>
<td>37.38</td>
</tr>
<tr>
<td>2.5</td>
<td>32.40</td>
<td>32.37</td>
<td>37.86</td>
<td>38.05</td>
</tr>
<tr>
<td>3</td>
<td>33.14</td>
<td>33.12</td>
<td>38.44</td>
<td>38.64</td>
</tr>
<tr>
<td>3.5</td>
<td>33.81</td>
<td>33.81</td>
<td>38.95</td>
<td>39.15</td>
</tr>
<tr>
<td>4</td>
<td>34.47</td>
<td>34.48</td>
<td>39.48</td>
<td>39.71</td>
</tr>
<tr>
<td>4.5</td>
<td>35.07</td>
<td>35.11</td>
<td>39.92</td>
<td>40.17</td>
</tr>
<tr>
<td>5</td>
<td>35.63</td>
<td>35.66</td>
<td>40.24</td>
<td>40.49</td>
</tr>
</tbody>
</table>

Table 4.1 PSNR Results of the Proposed Video Coder with LBS MC

The original video materials are in monochrome NTSC format at 30 frames/s. The size of the pictures is chopped to $704 \times 480$ for the convenience of DWT decimation. The source code of the proposed video coder is written in ANSI C++ and compiled with Microsoft® Visual C++ .NET 2003 edition. The programs run on a 32 bits computer equipped with Intel® Core™ 2 CPU (4300@1.80GHz), 2GB main memory, 250GB 7200rpm Harddrive and Windows XP SP2 Professional edition. The sequences are coded in I P I P … I P structure, i.e. an intra-coded frame followed by an inter-coded frame, so on and so forth. It is worth mentioning that for Mobile & Calendar, the PSNR improvement is up to 1.1 dB. The only exception is Football, where a slight drop of PSNR value occurs at low to mid bitrates. The above results proved that the LBS efficiently reduce the temporal redundancy in the DWT/MC/DPCM coding framework.
4.4 Context Design for Motion Information Encoding

In addition to Witten’s arithmetic coding scheme, it is also possible to use the MCBAC to encode the motion information. The adoption of MCBAC for motion information encoding employs 7 contexts based on MQ coder. The binarization and context design for MV coding is borrowed from H.264 (D.Marpe, H.Schwarz and T.Wiegand 2003). The \( UEGk \) binarization procedure is described in Appendix C. The context design of binarized string is the same with that in H.264 (D.Marpe, H.Schwarz and T.Wiegand 2003). Three contexts are used to code the first symbol, and one context is used to code the second, third, fourth, and fifth to ninth symbol of the prefix part, respectively. The \( EGk \) suffix part symbols are coded using a uniform context model. The PSNR results are shown in Table 4.2. The configuration of the parameters for the proposed video coder is the same with that in Section 4.3. The overall PSNR results when the coder employs the MCBAC for motion information encoding are quite similar to those when Witten’s algorithm is employed. For Mobile&Calendar, the results of MCBAC for motion

<table>
<thead>
<tr>
<th>Bitrate (Mbits/s)</th>
<th>Football</th>
<th>Susie</th>
<th>Balloon &amp; Pops</th>
<th>Mobile &amp; Calendar</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>WMC</td>
<td>MMC</td>
<td>WMC</td>
<td>MMC</td>
</tr>
<tr>
<td>1.5</td>
<td>30.29</td>
<td>30.15</td>
<td>36.34</td>
<td>36.34</td>
</tr>
<tr>
<td>2</td>
<td>31.44</td>
<td>31.33</td>
<td>37.38</td>
<td>37.38</td>
</tr>
<tr>
<td>2.5</td>
<td>32.37</td>
<td>32.29</td>
<td>38.05</td>
<td>38.06</td>
</tr>
<tr>
<td>3</td>
<td>33.12</td>
<td>33.05</td>
<td>38.64</td>
<td>38.65</td>
</tr>
<tr>
<td>3.5</td>
<td>33.81</td>
<td>33.75</td>
<td>39.15</td>
<td>39.16</td>
</tr>
<tr>
<td>4</td>
<td>34.48</td>
<td>34.42</td>
<td>39.71</td>
<td>39.71</td>
</tr>
<tr>
<td>4.5</td>
<td>35.11</td>
<td>35.05</td>
<td>40.17</td>
<td>40.18</td>
</tr>
<tr>
<td>5</td>
<td>35.66</td>
<td>35.61</td>
<td>40.49</td>
<td>40.50</td>
</tr>
</tbody>
</table>

Table 4.2: PSNR Results of Using Different Motion Information Coding Schemes. Witten
for MC Coding (WMC). MCBAC for MC Coding (MMC) information encoding are slightly better than the ones from Witten’s algorithm. For Football, the results of MCBAC are slightly worse. For Susie and Balloon&Pops, the results are highly similar.

4.5 Summary

The MC in the DWT domain based on the proposed DWT/MC/DPCM coding framework and the motion information encoding are discussed in this chapter. First, a review of the MC techniques in the DWT domain is given. Second, the LBS MC scheme is used for the proposed coding framework and proven to offer consistent improvement in terms of the average PSNR results. For Mobile & Calendar, the PSNR improvement is up to 1.1 dB. The only exception is Football, where a slight drop of PSNR value occurs at low to mid bitrates. This PSNR performance drop in Football is consistent with the relatively uniform translational motion of the scene, where LBS does not necessarily outperform the wavelet tree based ME/MC in terms of spatiotemporal redundancy removal. That is, the LBS does not bring obvious benefit to simple translational motion. The above results prove that the LBS efficiently reduces the temporal redundancy in the DWT/MC/DPCM coding framework in general. Third, a MCABAC scheme using the contexts borrowed from H.264 is employed to code the motion information here. The comparisons show that the results of this scheme are comparable to those of Witten’s algorithm in terms of R-D performance.
Chapter 5

Context Design for Inter-frame Coder

This chapter presents the redesign of the contexts for the proposed inter-frame coder, which employs the MCABAC structure. The outline is introduced as follows. Section 5.1 provides a brief review of the methodology for the context design. Section 5.2 presents the residue data analysis and the derived optimized contexts based on the proposed coder discussed in previous chapters. Section 5.3 presents the experimental results of the coder adopting the above contexts for the inter-frame coding part. A conclusion is drawn in Section 5.4.

5.1 Introduction

Prior to in-depth discussions, it is necessary to brief the theoretical foundation of the context design for the adaptive binary arithmetic coding in order to make the content in this chapter self-contained. It is assumed that the random process of the information source to be encoded and that of the modeler which predicts the probabilities of the events is $X$ and $Y$, respectively. The discrete random variable of $X$ at any moment $i$ is denoted as $X_i$. Therefore, $X$ can be represented by $X_1, X_2, \cdots, X_n$. In reality, the value of the message generated by $X$ is $x = (x_1, x_2, \cdots, x_n)$, where each symbol $x_i$ is the discrete value taken at moment $i$. There is another important quantity $X^{i-1} = X_1X_2\cdots X_{i-1}$, which
essentially describes what has happened in the past $i-1$ moments in respect to the current moment $i$. The corresponding generated message of $X^{i-1}$ is denoted as $x^{i-1} = (x_1, x_2, \ldots, x_{i-1})$. According to the relationship formulated by (2.8), where the conditional entropy, denoted as $H(X|Y)$, is determined by the subtraction of the marginal entropy $H(Y)$ or $H(X)$ and the mutual information $I(X;Y)$. In context based entropy coding, $Y$ can be approximated by the context function $f(x^{i-1})$. Theoretically, it is better to have as many contexts as the possible distinct values of $x^{i-1}$. However, in practice, too many contexts will bring the so-called context dilution problem as well as the complexity of the implementation (Z.Liu and L.J.Karam 2005). Therefore it is desired to have reasonable numbers of contexts which provide sufficiently accurate modeling.

The inter-frame data, i.e. the residue image, which is produced by the proposed video coder in previous chapters, is illustrated in Fig. 5.1, where the coefficient values are normalized to the scope $[0,255]$. In order to have a better understanding of the redundancies existing among the coefficients in the residue images, it is of great importance to investigate the bit distributions at each individual bitplane because of the bitplane coder nature of EBCOT. The bit distributions at each individual bitplane are illustrated in Fig. 5.2 where the data of the first residue image of the sequence Football at 5Mbit/s is used. The value of each individual pixel of the residue image is converted to a sign-magnitude integer one, where 32 bits precision is used in the implementation. The bitplane associated with the most significant bit (MSB) is illustrated in Fig. 5.2(a), which
Fig. 5.1 Illustration of the residue image before quantization and encoding

(a) first residue image of Football at 5Mbits/s

(b) first residue image of Mobile&Calendar at 5Mbits/s
contains merely the values of the signs and all the subsequent bitplanes contain the magnitude data. It can be seen from Fig. 5.1 and 5.2 that there are strong redundancies among and across the values at each individual bitplane because the statistical redundancy is essentially determined by the skewed probability of the events. In the case of binary arithmetic coding, the events are “0” and “1”. In Fig. 5.1 and 5.2, the distributions of “0” and “1” are obviously highly skewed and structural.

The design of the contexts for the inter-frame residue information follows a general

(a) p = 31

(b) p = 28

(c) p = 27

(d) p = 26
Fig. 5.2 Illustration of residue data at each individual bitplane

(p is the bitplane index)
approach of: 1) determining the categories of information used to form the contexts; 2) computing the amount of mutual information as specified by (2.8); 3) computing the curve of mutual information reduction vs. number of contexts, where (2.9) and (2.10) will be used; 4) determining the desired number of contexts based on the curve computed in step 3). The readers can see the practice of these four steps in Section 5.2.

5.2 Data Analysis and Context Design

Three categories of information are taken into consideration in the context redesign:

1) significance information, where the significance statuses of the 8 immediate neighbors are used to form the context for the current pixel;

2) sign information, where the sign of the neighboring pixels are used to form the contexts for the current pixel;

3) magnitude information.

Currently, the run mode is entered when the following conditions hold simultaneously (D.S.Taubman and M.W.Marcellin 2002) as described in Appendix A:

1) \( \sigma[j_0] = 0 \) for \( 0 \leq r < 4 \), where \( j_0 = j \) and \( j_r \) is the \( r \)th position from \( j \) along the visiting scan. That is to say, the four consecutive samples from the current coefficient along the scan are insignificant;

2) All four samples must have insignificant neighbors at current moment, i.e.,
\[
\kappa^\beta[j_0] + \kappa^\nu[j_0] + \kappa^d[j_0] = 0, \quad 0 \leq r < 4;
\]
3) The four samples must belong to the same single stripe column.

A binary output symbol is coded to indicate whether or not all of the four samples are insignificant at this “moment”. A value of 0 means all the four samples remain insignificant while a value of 1 means that at least one of the four samples become significant. This symbol is called “run interruption” symbol and coded with a separate context. When the run interruption symbol is 1, i.e., one of the consecutive samples become significant, the run length $r$ must be coded as well, followed by the coding of the sign of the first significant sample.

A few important symbols are listed in Table 5.1 despite the fact that they are already used in Appendix A in order to make this chapter self-contained. They will be used by the subsequent parts of this chapter.

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$j$</td>
<td>the sample location which contains coordinate values in both spatial horizontal and vertical directions.</td>
</tr>
<tr>
<td>$p$</td>
<td>the index of the current bitplane.</td>
</tr>
<tr>
<td>$\chi[j]$</td>
<td>the sign of the current coefficient.</td>
</tr>
<tr>
<td>$v$</td>
<td>the magnitude of the current coefficient.</td>
</tr>
<tr>
<td>$v^{(p)} = \left\lfloor \frac{v}{2^p} \right\rfloor$</td>
<td>the quantized coefficient value if the $p$ least significant bits (LSBs) are dropped from the magnitude.</td>
</tr>
</tbody>
</table>
\[
\sigma^{(p)}(j) = \begin{cases} 
1 & v^{(p)}(j) > 0 \\
0 & v^{(p)}(j) = 0
\end{cases}
: \text{the significance.}
\]

\(\sigma[j]\): the significance state, which is initialized to 0 and switched to 1 immediately after coding the first non-zero magnitude bit.

\(\kappa[j]\): the context label.

<table>
<thead>
<tr>
<th>Table 5.1</th>
<th>Symbols for Context Design</th>
</tr>
</thead>
</table>

### 5.2.1 Significance Contexts

When \(\sigma[j]=0\), one of the significance contexts is chosen by the encoder to code \(\sigma^{(p)}[j]\).

The significance contexts are formed by the 8 immediate neighbors. Since \(\sigma[.]\) of each neighbor is either 0 or 1, there are 256 possible distinct combinations formed by the significance states of these neighbors.

### 5.2.2 Sign Contexts

When the current coefficient is found to be significant, one of the sign contexts is chosen by the encoder to code \(\chi[j]\). The sign contexts for the current pixel will be formed by the signs from its four neighbors, which are the top, bottom, left, and right neighbor, respectively. The sign of each neighbor can have three possible values, which are positive (+1), negative (-1), and insignificant (0). Therefore there are 81 possible various combinations from these four neighbors.
CHAPTER 5 CONTEXT DESIGN FOR INTER-FRAME CODER

5.2.3 Refinement Contexts

When the current coefficient is already found to be significant, one of the refinement contexts are chosen by the encoder to encode $v^{(p)}$. The contexts are formed by 1) the significance states of the 8 immediate neighbors and 2) whether the bit being coded is the first refinement bit. There are 512 possible distinct combinations.

5.3 Experimental Results

The Football, Baloon & Pops, Susie, and Mobile & Calendar are used as the training sequences for inter-frame data analysis. The marginal probability distribution and the conditional probability distribution within each context category are computed. Based on these two probability distributions, the marginal entropy and the conditional entropy are derived. Then the maximum mutual information contained in each context category is obtained from (2.8)-(2.10). It is noted that, the statistical analysis for HL, LH, and HH bands were performed separately here, which is inspired by the work in (D.Taubman 2000; D.S.Taubman and M.W.Marcellin 2002; Z.Liu and L.J.Karam 2005). The marginal probability and the conditional probability of each individual context template described in Section 5.2.1-5.2.3 is drawn in Fig. 5.3, where the subplots in the left all show the marginal probabilities while those in the right all show the conditional probabilities. The corresponding mutual information curves to reflect the optimal contexts merge are drawn in Fig. 5.4. The theoretical maximum mutual information for the significance coding, the sign coding and the MR coding is listed in Table 5.2 and drawn in Fig. 5.4 as horizontal upper-limit lines.
Fig. 5.3 Probability Distributions of Inter-frame Data
(a) MI Curve for Significance Contexts – HL Band

(b) MI Curve for Significance Contexts – LH Band
(c) MI Curve for Significance Contexts – HH Band

(d) MI Curve for Sign Coding
There are a few points that make the statistics of the inter-frame data distinct from that of the intra-frame data:
1) It can be seen from Fig. 5.3(a), (c) and (e), the marginal probabilities of the context templates for significance coding shows strong skewed toward 0. The theoretical maximum mutual information from the significant context category is 5 times of that from the other two context categories, which shows much stronger statistical redundancy. Therefore, the contexts of significance coding are predominantly more important than the other two coding categories.

2) In Fig. 5.4(e), for magnitude refinement coding, the use of 2 contexts is good enough because more contexts will not bring about obvious gain in MI increase.

The current number of significance contexts of the EBCOT is limited to 9. They are replaced with the re-designed 9 contexts. The PSNR results of the proposed coder with these 9 new significance contexts for the inter-frame coding are listed in Table 5.3. The original video materials are in monochrome NTSC format at 30 frames /s. The size of the pictures is chopped to 704×480 for the convenience of DWT decimation. The source code of the proposed video coder is written in ANSI C++ and compiled with Microsoft® Visual C++ .NET 2003 edition. The programs run on a 32 bits computer equipped with Intel® Core™ 2 CPU (4300@1.80GHz), 2GB main memory, 250GB 7200rpm Harddrive and Windows XP SP2 Professional edition. The sequences are coded in I P I P … I P structure, i.e. an intra-coded frame followed by an inter-coded frame, so on and so forth. Each of them is coded in frame mode. The results of the coder using old JPEG2000 context set for the inter-frame coder are compared against the one using new significance context set. For Susie and Mobile&Calendar, the new context set is proven to be slightly better. For Football and Baloon&Pops, the JPEG2000 significance context set still has
higher PSNR results. Looking at Fig. 5.4(a)-(c), it can be seen that when the significance context number is limited to about 9, the theoretical mutual information the new significance coding can achieve in maximum is about 0.025 bit, while the red lines, the theoretical maximum mutual information in Fig. 5.4(a)-(c), have values above 0.03 bit. It is therefore anticipated that the coding efficiency of the significance coding can be improved with an increase of the number of contexts. For example, if the significance context number is set to 20, the theoretical maximal mutual information the new significance coding can achieve is about 0.03 bit for HL and HH band, and slightly below 0.03 for LH band.

<table>
<thead>
<tr>
<th>Sequence</th>
<th>Football</th>
<th>Susie</th>
<th>Mobile&amp;Calendar</th>
<th>Balloon&amp;Pops</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bitrate (Mbits/s)</td>
<td>JC</td>
<td>NC</td>
<td>JC</td>
<td>NC</td>
</tr>
<tr>
<td>1.5</td>
<td>30.29</td>
<td>29.91</td>
<td>36.34</td>
<td>36.35</td>
</tr>
<tr>
<td>2</td>
<td>31.44</td>
<td>31.02</td>
<td>37.38</td>
<td>37.38</td>
</tr>
<tr>
<td>2.5</td>
<td>32.37</td>
<td>31.94</td>
<td>38.05</td>
<td>38.07</td>
</tr>
<tr>
<td>3</td>
<td>33.12</td>
<td>32.69</td>
<td>38.64</td>
<td>38.66</td>
</tr>
<tr>
<td>3.5</td>
<td>33.81</td>
<td>33.38</td>
<td>39.15</td>
<td>39.16</td>
</tr>
<tr>
<td>4</td>
<td>34.48</td>
<td>34.03</td>
<td>39.71</td>
<td>39.71</td>
</tr>
<tr>
<td>4.5</td>
<td>35.11</td>
<td>34.64</td>
<td>40.17</td>
<td>40.18</td>
</tr>
<tr>
<td>5</td>
<td>35.66</td>
<td>35.19</td>
<td>40.49</td>
<td>40.49</td>
</tr>
</tbody>
</table>

Table 5.3  Average PSNR Results

(Note: JC and NC standard for JPEG2000 contexts and new contexts, respectively)

5.4 Summary

The statistical analysis of the inter-frame data is performed in this chapter. This leads to the redesign of the contexts for the inter-frame data coding. The analysis shows that the statistical redundancy in the significance coding is much stronger than that in the sign or
magnitude refinement coding. The calculated mutual information curve for optimal context template merging gives the concrete information on how the optimal contexts should be formed. The current number of significance contexts of the EBCOT is limited to 9. They are replaced with the re-designed 9 optimal contexts. The results of the coder using old JPEG2000 context set for the inter-frame coder are compared against the one using new significance context set. For Susie and Mobile&Calendar, the new context set is proven to be slightly better. For Football and Baloon&Pops, the JPEG2000 significance context set still has higher PSNR results. It is therefore anticipated that the coding efficiency of the significance coding can be improved with an increase of the number of contexts.
Chapter 6

Vision Modeling for DWT Video Coding

This chapter demonstrates the application of a Perceptual Distortion Measure (PDM) Based on Vision Model to the intra-coder of the proposed DWT/MC/DPCM video coding framework. The MSE in the R-D computation stage inside the EBCOT coding engine is replaced with the PDM. The analysis shows that the PDM enhances the visual quality of the coded intra-frames. Furthermore, a visibility assessment of the quantization error in the DWT domain is carried out to quantitatively measure the effects of the quantization error in the DWT domain to human viewers. The outline of this chapter is as follows. A brief review of the relevant work on DWT based perceptual coding in the literature is given in Section 6.1. The application of the vision model based PDM to the intra-coder is presented in 6.2. A visibility assessment to the quantization error in the DWT domain is given in 6.3. A summary is drawn in Section 6.4.

6.1 Vision, Modeling and Application

It is of no doubt that the detection and discrimination measurements of the HVS to the spatiotemporal patterns are of invaluable importance to the study of the behaviors of the vision system. With a better understanding of the casual relationships between the patterns and the sensitivities/discriminations, it is easier to predict the responses of the HVS to the more complicated visual signals such as the image/video in either original
pixel or transform domain. Based on this motive, and also driven by the fact the components in the HVS such as the optical receptors, the pathways, and the cortex neuron cells all have impacts to the final response, vision models can be established which will greatly facilitate the perception related computation where the subjective response to the visual material has to be taken into account. The image and video compressions are amongst the applications of the vision models.

Before in-depth discussion of the vision model, it is necessary to review the concept of neuron images. It is believed that Robson is the fist one suggesting this term (B.A.Wandell 1995). The basic idea is that the effects of many visual components contribute to the generation of the final representation in the visual cortex. Neurons in the visual pathways and cortex are categorized into different groups, and each individual group conveys different types of information. This “final representation” adopts the form of an image therefore allows a straightforward understanding of the output of the visual system and the relationships between input/output and among output. Cortex transform (A.B.Watson 1987) was suggested to simulate the response of simple cells in the cortex at different frequency and orientation. This scheme was implemented by a set of filters. It offered a few advantages such as scale invariance, reconstruction by addition, pyramid structure and fast algorithm. It was designed as a method of rapidly computing the responses of arrays of model cells to an image. Based on this transform, an image coding framework was proposed in (A.B.Watson 1987) for perceptual lossless coding. This scheme considered the response fields of the simple cells in the cortex and achieved a bitrate at about 1 bit/pixel. Schade (O.H.Schade 1956) suggested a single resolution
neural image computation scheme. He set up a photoelectric analog of the visual system to predict the visual sensitivities. The assumption of his experiment was that the fovea pattern sensitivity could be predicted by a single neural image. It was also assumed that the neural image is a shift-invariant linear encoding. Nevertheless, it was proven that Schade’s assumption was not the case because his prediction of the pattern sensitivity did not match the experimental data. However, Shade’s suggestion of neuron image and the use of shift-invariant transform as the first stage can be incorporated into the so-called multiresolution theory. The use of multiresolution neural image is driven by a few observed phenomena of the vision system such as the pattern adaptation, masking, to name a few. There are quite a few multiresolution analysis tools available in the literature. Amongst them are the work of (P.J.Burt and E.H.Adelson 1983; S.G.Mallat 1989; W.T.Freeman and E.H.Adelson 1991).

The visual masking phenomena can be briefly explained as follows: given a visual signal, its appearance will be hidden, to some extent, by the presence of another signal nearby. The latter signal is named as masker. Three examples of visual masking are illustrated in Fig. 6.1. It can be seen the effects of visual masking depend on the masker’s frequency and orientation. It is therefore logical to incorporate the characteristics, or even the models, of the HVS into the design of the components or the process of the image and video processing schemes. As a result, these perceptual image and video processing schemes are more adapted to the sensitivity of the visual perception. For image and video compression systems, this means that the coded image or video is more subjectively pleasing under the
(a) Masking with Different Frequency and Same Orientation

Note: the three diagrams from left to right are the original sinusoidal pattern, the sinusoidal pattern with different frequency but same orientation, and the mixture of the aforementioned two signals, respectively.

(b) Masking with Similar Frequency and Same Orientation

Note: the three diagrams from left to right are the original sinusoidal pattern, the sinusoidal pattern with similar frequency and same orientation, and the mixture of the aforementioned two signals, respectively.
(c) Masking with Similar Frequency but Different Orientation

Note: the three diagrams from left to right are the original sinusoidal pattern, the sinusoidal pattern with similar frequency but different orientation, and the mixture of the aforementioned two signals, respectively.

Fig. 6.1 Examples of Visual Masking

same bitrate in comparison with the traditional coders. The efforts could be traced back to more than three decades ago (T.G.Stockham 1972). The work of Mannos and Skrison (J.L.Mannos and D.J.Sakrison 1974) incorporated the HVS into image transmission. The authors compared a class of distortion measures developed from earlier psychophysical experiments against chosen images. They modeled the HVS in the form of Modulation Transfer Function (MTF). Unfortunately, the lack of understanding to HVS in that era was the main bottleneck. For example, the masking effects were not taken into account at all and the model they considered was oversimplified. Nonetheless, the authors pointed out that a distortion measure in accordance with subjective evaluation of image quality was critical for subjectively good image transmission or coding applications. The above transfer function was also used by other researchers (H.Lohscheller 1984; K.H.Tzou, T.R.Hsing et al. 1984). Nill (N.B.Nill 1985) and Ngan et al. (K.N.Ngan, K.S.Leong et al. 1989) introduced different variants of MTF functions, respectively, for transform coding of images. In the work of (K.N.Ngan, K.S.Leong and H.Singh 1989), an adaptive quantization was employed within the DCT based image coding framework. The cosine transform coefficients were weighted by the HVS function to generate the coefficients in perceptual domain during the quantization process. An adaptive block distortion equalization was adopted to reduce the blocking artifacts. Chitprasert and Rao compared
the MTF functions of (J.L.Mannos and D.J.Sakrison 1974; N.B.Nill 1985; K.N.Ngan, K.S.Leong and H.Singh 1989) and derived their own form of MTF function and used it for a human visual weighted progressive image transmission based on DCT technique. The DCT transform coefficients themselves were not altered during the procedure therefore there was no requirement for postfiltering operation in the decoder. Watson and Ahumada proposed a Hexagonal Orthogonal-Oriented Quadrature Pyramid (HOP) transform (A.B.Watson and A.J.Ahumada 1989) to simulate the biological transform existing in the pathway between the β retinal ganglion cells, to the parvocellular layers of the lateral geniculate cells, and from there to layers 4cβ and 4a of the striate cortex, where a large population of the neuron cells were the so-called oriented simple cells of V1. The transform had the following properties: pyramid structure, invertible, local in 2-D space and frequency, self-similar, odd and even (quadrature pairing), hexagonal lattice and computation efficient. There were 7 basis functions, one low-pass and 6 bandpass. The 6 bandpass basis functions consisted of 3 with even symmetry and 3 with odd symmetry. The three even kernels were rotations of 0, 60, and 120 degrees of a common kernel. The three odd kernels were constructed in a similar way. The low-pass coefficients were the input for the next level of transformation. A perceptual tuned sub-band image coder was introduced by Safranek and Johnston (R.J.Safranek and J.D.Johnston 1989) which used an empirically derived texture perceptual masking model to set noise-level in a DPCM quantizer. The proposed coder used 16 subbands through Generalized Quadrature Mirror Filter (GQMF) decomposition (J.W.Woods and S.D.O’Neil 1986; R.V.Cox 1986). The perceptual threshold was computed from three components: the base sensitivity, the sensitivity adjustment for different brightness, and the texture masking adjustment. The
first two components were derived using trained subjects and training images. The
texture masking adjustment was a function of the texture energy at each individual
transform coefficient. It was composed of the weighted sum of the local energy in each
AC subband and the variance in DC band at the same locality. The weights of each
subband were determined from the HVS modulation transfer function (MTF). Watson
suggested a Perceptual-component Architecture (PCA) for digital video processings
(A.B.Watson 1990). The PCA did not give a specific digital video coding scheme.
Instead, it incorporated the human visual coding as the model in all aspects of the
architecture. First, it partitioned the image stream into three opponent chromatic channels,
which were achromatic(A), red-green(RG), and yellow-blue(YB). Second, the opponent
chromatic signals were applied spatial sampling and decomposition into spatial frequency
bands as the retina, the lateral geniculate nucleus, and the primate visual cortex would do.
Again, Watson did not give an implementation of this spatial sampling and
decomposition. Instead, he discussed the prerequisites of these operations to simulate the
behaviors of the HVS. Third, the relationship between space, time, and color were
extensively discussed. The HOP (A.B.Watson and A.J.Ahumada 1989) was also
discussed as an example which implements some principles of the PCA. O’Rourke and
Stevenson introduced a HVS based wavelet decomposition (T.P.O’Rourke and
R.L.Stevenson 1995) for image compression. That work, to some extent, was based on
the research in 1960’s (F.W.Campbell and J.J.Kulikowski 1966; F.W.Campbell,
J.J.Kulikowski et al. 1966; F.W.Campbell and J.G.Robson 1968; F.W.Campbell and
L.Maffei 1970). In the work of (F.W.Campbell, J.J.Kulikowski and J.Levinson 1966), it
was found that the HVS contrast sensitivity versus the spatial frequencies were different
along various orientations. In fact, the contrast sensitivity versus spatial frequencies in a 2D space was roughly diamond shaped, from which the authors of (T.P.O'Rourke and R.L.Stevenson 1995) concluded that the passband of the subband coding system should be diamond shaped to incorporate the characteristics of the HVS. In (T.P.O'Rourke and R.L.Stevenson 1995) a HVS-based wavelet design was proposed and compared against the traditional separable wavelets and non-separable wavelets. The experimental results demonstrated a superior performance of the proposed wavelet filters in terms of perceived quality. References for the perceptual image/video coding in recent years can be found in (S.Winkler 2000; D.M.Tan 2002; M.J.Nadenau, J.Reichel et al. 2002; H.R.Wu and K.R.Rao 2005).

The response of the Human Visual System (HVS) to the quantization errors in the DWT domain is not intensively understood. There have been some efforts in the literature trying to address this issue but they are mainly for still image applications. Watson et al. measured the visual thresholds for still images in (A.B.Watson, G.Y.Yang et al. 1996) using basis DWT functions. On the other hand, they proposed a model that described the relationship of the quantization noise with the final perception on the basis of statistics. Unfortunately, the so-called masking mechanism was not considered in that work. Besides the statistics school, there has been another approach that used the existing psychophysical knowledge to ‘shape’ the information in DWT domain as in the work of (M.J.Nadenau, J.Reichel et al. 2003) where the contrast sensitivity function (CSF) was adapted to the frequencies in the DWT domain. That work successfully resulted in the improvement of the distribution of the noise across different subbands in the DWT
domain. However, only CSF but other senior psychophysical mechanisms was taken into account. A vision model was proposed by Lambrecht (B.Lambrecht 1996) that consists of four stages and they are 1) transform, 2) CSF filtering, 3) Contrast Gain Control (CGC) and 4) the pooling and summation. Tan et al. proposed a vision model based on that general framework and successfully applied it in the still image compression (D.M.Tan 2002).

6.2 Application of PDM to Intra-coder

The CGC (A.B.Watson and J.A.Solomon 1997) is a reliable vision model which evaluates the visual difference two images. It consists of three-stages: linear transform, response and detection. The linear transform decomposes an image into different frequency and orientation bands. This emulates the behavior of the HVS since neural signals in the visual cortex are frequency and orientation selective. The contrast sensitivity is then applied to the signal in each subband following frequency decomposition in the form of weights. The second stage describes the responses resulting from the interactions of different neural signals from different frequencies and/or orientations termed intra-frequency and inter-orientation masking. The third stage simulates the overall detection measure based on the responses derived in the second stage. A HVS model proposed by Tan in (D.M.Tan 2002) based on the CGC is adopted in this thesis for the intra-frame coder of the proposed video coding framework. In this model, separate response functions $R_\theta$ and $R_x$ are used to calculate both inter-orientation and intra-frequency masking, respectively. The response function has the general form:
\[ R_z = k_z \cdot \frac{E_z}{I_z + \sigma_z^q} \]  \hspace{1cm} (6.1)

where \( E_z \) and \( I_z \) are the excitation and inhibition functions, \( K_z \) and \( \sigma_z \) the scaling and saturation coefficients and \( Z \in \{\Theta, X\} \), with \( \Theta \) and \( X \) specifying inter-orientation and intra-frequency masking domains, respectively. The definitions of the excitation and inhibition functions of the two domains are given as follows:

\[ E_\theta = (A_\theta)^{p_\theta} \]  \hspace{1cm} (6.2)
\[ E_x = (A_x)^{p_x} \]  \hspace{1cm} (6.3)
\[ I_\theta = \sum_{\phi} (A_{\phi})^{q_{\phi}} \]  \hspace{1cm} (6.4)
\[ I_X = \frac{8}{N} \sum_{i} (A_i)^{q_{i}} + \sigma_{var}^{q} \]  \hspace{1cm} (6.5)

where \( A_\theta \) and \( A_x \) are the transform coefficients at orientation \( \theta \) and spatial frequency location \( x \), respectively, \( \sum_{\phi} (A_{\phi})^{q_{\phi}} \) the weighted sum of the coefficients spanning all orientations and \( \sum_{i} (A_i)^{q_{i}} \) the weighted sum of the neighboring coefficients about the pixel \( A_x \) whose size is dependent upon the frequency level of \( A_x \). The response function (6.1) is applicable to all coefficients except for those in the isotropic (DC) band. As a result, the response function of the DC band is written as
where $\overline{A}$ and $A$ are the quantized and un-quantized DC coefficients, respectively. The final stage of the HVS model detects the difference between two images. A simple squared-error function is used as defined in (6.7),

\[
D_z = |R_{\alpha,z} - R_{\beta,z}|^2
\]

(6.7)

where $R_{\alpha,z}$ and $R_{\beta,z}$ are the responses of the two images, $\alpha$ and $\beta$. Pooling equation (6.7) for all coefficients spanning all frequencies and orientations will result in the overall distortion between the two images. The final distortion measure, $D_r$, encompassing both intra-frequency and inter-orientation masking, is given in (6.8),

\[
D_r = g_x \cdot D_x + g_\theta \cdot D_\theta
\]

(6.8)

where $D_x$ and $D_\theta$ are the visual distortions after spatial and orientation masking, respectively, and $g_x$ and $g_\theta$ the gains for spatial frequency and orientation masking, respectively. In (6.1) – (6.8), $p_\theta$, $p_x$, $p$ and $q$ are parameters related to the HVS model.

The block diagram of the proposed intra coder is illustrated in Fig. 6.2. The EBCOT
coding engine is composed of two tiers. The tier 1 performs the bitplane based MCABAC and generate R-D pairs at each truncation point. The tier 2 computes the contributions from various code blocks according to the given bitrate budget. The key of the work here is to replace the MSE used in the R-D computation with the proposed distortion measure described by (6.1) – (6.8). Therefore the R-D pairs are more consistent with the perception.

![Block Diagram of the Proposed Intra Coder](image)

The video coder here is based on the coding framework introduced in previous chapters with the exception that only the I-frame mode is used in this chapter. In addition, the number of DWT decomposition levels is 5, therefore the vision model parameters in (D.M.Tan 2002) can be used in the intra-frame coder and the results can be compared with those of Motion JPEG2000. The parameter values are listed in Table 6.1. For details
regarding parameter optimization, please refer to (D.M. Tan 2002). The value of $q$ is 1 in the real implementation. The coder is compared with Motion JPEG2000 and MPEG-2 intra-frame coder. The subjective analysis shows that the proposed perceptual intra-frame coder has superior perceived quality than the two benchmarks. Some of the results are illustrated in Fig. 6.3 and 6.4.

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Manual</th>
</tr>
</thead>
<tbody>
<tr>
<td>$k_X$</td>
<td>1.0880</td>
</tr>
<tr>
<td>$k_\theta$</td>
<td>0.9876</td>
</tr>
<tr>
<td>$\sigma_X$</td>
<td>5.5550</td>
</tr>
<tr>
<td>$\sigma_\theta$</td>
<td>7.6800</td>
</tr>
<tr>
<td>$p_X$</td>
<td>2.5800</td>
</tr>
<tr>
<td>$p_\theta$</td>
<td>2.3950</td>
</tr>
<tr>
<td>$g_X$</td>
<td>0.7588</td>
</tr>
<tr>
<td>$g_\theta$</td>
<td>0.4834</td>
</tr>
</tbody>
</table>

Table 6.1 Model Parameters

(a) Coded by Proposed Intra-Coder  (b) Truncated part of (a)
Fig. 6.3  Football Frame 5

(a) Coded by Proposed Intra-Coder  
(b) Truncated part of (a)

(c) Coded by Motion JPEG2000  
(d) Truncated part of (c)

(e) Coded by MPEG-2  
(f) Truncated part of (e)
6.3 Visibility to Quantization Errors in DWT Domain

A carefully designed visual experiment is needed to acquire enough supporting data for the vision model. The block diagram of the simulation software is shown as in Fig. 6.5, where \( x \) represents for the original input. \( X \), \( \hat{X} \) and \( \hat{x} \) represents for the DWT coefficient, the “polluted” DWT coefficient, and the reconstructed signal, respectively. The noise generator ideally can generate different types of noise. In this thesis work it produces the quantization noise \( \Delta X \). The ‘polluted’ signal \( \hat{X} \) is then sent to the inverse DWT module in order to get the reconstructed image \( \hat{x} \). Our purpose here is to generate
quantization noise for certain subband in the DWT domain while keep the values of the coefficients in other subbands untouched. The noise can be represented by

$$\Delta X_{l,o} = \left[ X_{l,o} / QS \right] \cdot QS - X_{l,o} \tag{6.9}$$

where $\Delta X_{l,o}$ is the noise in the subband $(l,o)$, $X_{l,o}$ the transform coefficient in the subband $(l,o)$, $QS$ the quantization step, $l$ the resolution level, and $o$ the orientation of the subband, respectively. The biorthogonal 9/7 filter bank (M.Vetterli and J.Kovacevic 1995) is used as the filtering kernel for five levels of DWT analysis and synthesis. The original value of each pixel is in the scope of $[0,255]$ and no value shifting is performed before or after DWT.
The process of the visual experiment complies with the requirement of Rec. ITU-R BT.500-11 (BT.500-11 2002) which standards for the agreement upon the subjective assessment method for the television systems. A MITSUBISHI Diamond View 19” CRT Monitor is used (model DV19NF) to display videos with a refreshing rate 75Hz and resolution 1024x768. The viewers sit in front of the monitor, keeping a distance D that is certain times of the image height. The calculation of D is calculated as

\[
D = H_{\text{effective}} \frac{N_{\text{picture}}}{N_{\text{screen}}} \cdot C 
\]

where the \(H_{\text{effective}}\) is the effective physical height of the screen, \(N_{\text{picture}}\) the pixel number for the height of the picture, \(N_{\text{screen}}\) the pixel number for the height of the effective screen, \(C\) the ratio of D to the physical height of the picture, respectively. The values of above parameters for the test are shown as in Table 6.2. The choice of 3 for \(C\) is driven by the fact that the display device, a computer monitor, is normally viewed at shorter distances.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>(H_{\text{effective}}) (mm)</td>
<td>264</td>
</tr>
<tr>
<td>(N_{\text{picture}}) (pixels)</td>
<td>768</td>
</tr>
<tr>
<td>(N_{\text{screen}}) (pixels)</td>
<td>480</td>
</tr>
<tr>
<td>(C)</td>
<td>3</td>
</tr>
</tbody>
</table>

Table 6.2 Parameter Set

A Java program is used as the player for both original and reconstructed video sequences. The area around the picture on the screen is filled with black color. In order to utilize the
speed of the DirectDraw® Technology, the java program runs on a machine with P4 3.0GHz CPU, 1024M main memory and Windows Professional Edition operating system. NTSC format is chosen therefore the frame rate is 30 frames/s. For the convenience of the forward and inverse DWTs, the frame size is chopped from 720x486 down to 704x480 pixels. There are four natural scene video sequences used in the test and they are Balloon&Pops, Mobile&Calendar, Football, and Susie. They contain different motion types and complexities of the background. Four different quantization steps are used for each individual subband therefore the number of the ‘distorted’ sequence is:

\[ N = N_s \times N_b \times N_q \] (6.11)

where \( N_s \) is the number of original sequences, \( N_b \) the number of the subband after wavelet transform, \( N_q \) the number of quantization steps used, respectively. The values for \( N_s, N_b \) and \( N_q \) are 4, 16 and 4, respectively. As a result, \( N \) equals to 256. In order to get the assessment result as accurate as possible, it’s necessary to randomize these 256 video sequences and make sure that two neighboring reconstructed sequences are not from the same original sequence. Furthermore, the ITU-R BT.500-11 recommends that the time for each subjective test should be no longer than half hour. Therefore, the 256 video sequences are separated into 8 sessions. At the beginning of each session, several training sequences are inserted which will not be included in the statistical data. For each viewer (subject), he/she has to view all the 8 sessions.
The Double-Stimulus Impairment Scale (DSIS) method (BT.500-11 2002) is used for the subjective test. The viewers are asked to rate the perceptual distortions for the processed video sequence in five grades: imperceptible, perceptible but not annoying, slightly annoying, annoying, and very annoying.

The results of the average rating are demonstrated as in Fig. 6.6. Five viewers (subjects) participated in the visual tests. They are labeled as F, K, J, L and G, respectively. The average rating is computed as

\[
M = \sum R_i
\]  

(6.12)

where \(R_i\) represents for the rating from the ith subject. For the definition of \(R_i\), please refer to Table 6.3.

<table>
<thead>
<tr>
<th>Perception to The Noise</th>
<th>Rating ((R_i))</th>
</tr>
</thead>
<tbody>
<tr>
<td>Imperceptible</td>
<td>5</td>
</tr>
<tr>
<td>Perceptible But Not Annoying</td>
<td>4</td>
</tr>
<tr>
<td>Slightly Annoying</td>
<td>3</td>
</tr>
<tr>
<td>Annoying</td>
<td>2</td>
</tr>
<tr>
<td>Very Annoying</td>
<td>1</td>
</tr>
</tbody>
</table>

Table 6.3 Definition of the Visual Quality during the Experiment

The definition of the subband index in Fig. 6.6 is shown as in Table 6.4. For the designations of the subbands, readers are referred to Fig. 2.5.
(a) Balloon&Pops

(b) Mobile&Calendar
Fig. 6.6 Average Visual Assessment to the Quantization Noise

(c) Football

(d) Susie
It can be seen that the most sensitive subbands to the quantization noises are those of the lowest three resolution levels. The curves in Fig. 6.6 are the visual responses under different quantization steps. The real implementation of the quantization is more complex than Equ. (6.9). It can be formulated as

\[
\hat{X} = \left\lfloor \frac{X_{I,o} \cdot C_q}{Q} \right\rfloor \cdot \frac{Q}{C_q}
\]  

(6.13)

where \(C_q\) is the quantization constant and \(Q\) the quantization step. In the test, \(C_q\) is chosen as \(2^{20}\) while \(Q\) can have four values as marked in Fig. 6.6.

The results are quite interesting. Referring to (M.J.Nadenau, J.Reichel and M.Kunt 2003),
the maximum frequency of the signal measured in cpd in a picture can be formulated as

$$f_{\text{max}} = \tan(0.5^\circ) \cdot v \cdot r$$  \hspace{1cm} (6.14)

where $r$ is the resolution measured in pixels per meter, $v$ the viewing distance measured in meters. From the parameters in Table 6.2, the value of $f_{\text{max}}$ is computed as 12.57 cpd. The frequency scope of the decomposed signal in each subband is summarized as in Table 6.5. The frequency scopes in Table 6.5 have certain agreement with the CSF curve (M.J.Nadenau, J.Reichel and M.Kunt 2003) in general trend. However, there is indeed some shifts occurred because of the masking effects. For example, pure CSF results demonstrate that the most sensitive subbands should be 2HL Band, 2LH Band and 2 HH Band, but it is not always the case as in Fig. 6.6. This ‘shift’ needs more investigation in the future.

<table>
<thead>
<tr>
<th>Subband</th>
<th>Horizontal Frequency</th>
<th>Vertical Frequency</th>
</tr>
</thead>
<tbody>
<tr>
<td>5LL Band</td>
<td>$(0, \frac{1}{32} f_{\text{max}})$</td>
<td>$(0, \frac{1}{32} f_{\text{max}})$</td>
</tr>
<tr>
<td>5HL Band</td>
<td>$(\frac{1}{32} f_{\text{max}}, \frac{1}{16} f_{\text{max}})$</td>
<td>$(0, \frac{1}{32} f_{\text{max}})$</td>
</tr>
<tr>
<td>5LH Band</td>
<td>$(0, \frac{1}{32} f_{\text{max}})$</td>
<td>$(\frac{1}{32} f_{\text{max}}, \frac{1}{16} f_{\text{max}})$</td>
</tr>
<tr>
<td>5HH Band</td>
<td>$(\frac{1}{32} f_{\text{max}}, \frac{1}{16} f_{\text{max}})$</td>
<td>$(\frac{1}{32} f_{\text{max}}, \frac{1}{16} f_{\text{max}})$</td>
</tr>
<tr>
<td>4 HL Band</td>
<td>$(\frac{1}{16} f_{\text{max}}, \frac{1}{8} f_{\text{max}})$</td>
<td>$(0, \frac{1}{16} f_{\text{max}})$</td>
</tr>
<tr>
<td>4 LH Band</td>
<td>$(0, \frac{1}{16} f_{\text{max}})$</td>
<td>$(\frac{1}{16} f_{\text{max}}, \frac{1}{8} f_{\text{max}})$</td>
</tr>
<tr>
<td>4 HH Band</td>
<td>$(\frac{1}{16} f_{\text{max}}, \frac{1}{8} f_{\text{max}})$</td>
<td>$(\frac{1}{16} f_{\text{max}}, \frac{1}{8} f_{\text{max}})$</td>
</tr>
<tr>
<td>3 HL Band</td>
<td>$(\frac{1}{8} f_{\text{max}}, \frac{1}{4} f_{\text{max}})$</td>
<td>$(0, \frac{1}{8} f_{\text{max}})$</td>
</tr>
<tr>
<td>3 LH Band</td>
<td>$(0, \frac{1}{8} f_{\text{max}})$</td>
<td>$(\frac{1}{8} f_{\text{max}}, \frac{1}{4} f_{\text{max}})$</td>
</tr>
</tbody>
</table>
6.4 Summary

This chapter demonstrates the application of a Perceptual Distortion Measure (PDM) based on a vision model to the intra-coder of the proposed DWT/MC/DPCM video coding framework. The MSE in the R-D computation stage inside the EBCOT coding engine is replaced with the PDM. The analysis shows that the PDM enhances the visual quality of the coded intra-frames. Furthermore, a visibility assessment of the quantization error in the DWT domain is carried out to quantitatively measure the effects of the quantization error in the DWT domain to human viewers. The results of the assessment show that the perception to DWT domain noise as in Fig. 6.6 has certain agreement with the CSF curve in (M.J.Nadenau, J.Reichel and M.Kunt 2003) in general trend. However, there are indeed some shifts occurred because of the masking effects. These shifts need more investigation in the future.

<table>
<thead>
<tr>
<th>Table 6.5 Frequency Scopes of DWT Subbands</th>
</tr>
</thead>
<tbody>
<tr>
<td>3 HH Band</td>
</tr>
<tr>
<td>$(\frac{1}{8}f_{\text{max}}, \frac{1}{4}f_{\text{max}})$</td>
</tr>
<tr>
<td>2 HL Band</td>
</tr>
<tr>
<td>$(\frac{1}{4}f_{\text{max}}, \frac{1}{2}f_{\text{max}})$</td>
</tr>
<tr>
<td>2 LH Band</td>
</tr>
<tr>
<td>$(0, \frac{1}{4}f_{\text{max}})$</td>
</tr>
<tr>
<td>2 HH Band</td>
</tr>
<tr>
<td>$(\frac{1}{4}f_{\text{max}}, \frac{1}{2}f_{\text{max}})$</td>
</tr>
<tr>
<td>1 HL Band</td>
</tr>
<tr>
<td>$(\frac{1}{2}f_{\text{max}}, f_{\text{max}})$</td>
</tr>
<tr>
<td>1 LH Band</td>
</tr>
<tr>
<td>$(0, \frac{1}{2}f_{\text{max}})$</td>
</tr>
<tr>
<td>1 HH Band</td>
</tr>
<tr>
<td>$(\frac{1}{2}f_{\text{max}}, f_{\text{max}})$</td>
</tr>
</tbody>
</table>

6.4 Summary

This chapter demonstrates the application of a Perceptual Distortion Measure (PDM) based on a vision model to the intra-coder of the proposed DWT/MC/DPCM video coding framework. The MSE in the R-D computation stage inside the EBCOT coding engine is replaced with the PDM. The analysis shows that the PDM enhances the visual quality of the coded intra-frames. Furthermore, a visibility assessment of the quantization error in the DWT domain is carried out to quantitatively measure the effects of the quantization error in the DWT domain to human viewers. The results of the assessment show that the perception to DWT domain noise as in Fig. 6.6 has certain agreement with the CSF curve in (M.J.Nadenau, J.Reichel and M.Kunt 2003) in general trend. However, there are indeed some shifts occurred because of the masking effects. These shifts need more investigation in the future.
Chapter 7

Conclusions

This dissertation presents a novel DWT/MC/DPCM perceptual video coding framework. The proposed coder adopts the EBCOT (D.Taubman 2000) as the coding engine for both the intra- and inter-frame coder. The work encompasses a perceptual video coding framework, the adaptive coding mode, the MC in the DWT domain and the coding of motion information, the redesign of the inter-frame coding contexts, the application of a vision model based PDM to the intra-frame coder and the visibility assessment to the quantization errors in different DWT subbands.

Chapter 2 provides the theoretical foundation of the DWT based video coder. First, the general structure of traditional video coding is introduced. It encompasses the Predictive/Transform/Subband Coding, the quantization, the entropy coding, the context design, and the MQ coder. Second, the DWT theory is discussed, which includes the traditional DWT, the lifting implementation of DWT, and the more recent ADL scheme. Third, the MC in the DWT domain is presented. The WTMC and the LBS are used in this thesis work. Fourth, the bitplane based image/video coding techniques are revisited. Among the most well known schemes are the EZW, the SPIHT and the EBCOT. The EBCOT is adopted in this thesis work as the coding engine with redesigned contexts for the inter-frame data. The context design of the EBCOT is reviewed in Appendix A.
An extended review of the DWT based video coders in the literature can be found in Appendix B.

Chapter 3 presents a DWT/MC/DPCM based video coding framework. Although the DCT/MC/DPCM framework has been adopted by the current industry and international video compression standards (ISO/IEC 1993; ITU-T 1993; ISO/IEC 2000; ITU-T 2000; ISO/IEC 2004; ISO/IEC 2005; ITU-T 2005), the DWT based video coders has been extensively investigated since almost two decades ago (G.Karlsson and M.Vetterli 1988; A.S.Lewis and G.Knowles 1990). Among various DWT video coders in the literature, where a review has been given in Appendix B, the work of the DWT/MC/DPCM coding structure can be traced back to 1992 (Y.-Q.Zhang and S.Zafar 1992) and the investigation is still undergoing (F.G.Meyer, A.Averbuch and R.R.Coifman 1997; D.Blasiak and W.-Y.Chan 1998; S.Kim, S.Rhee, J.G.Jeon et al. 1998; H.W.Park and H.S.Kim 2000; G.Heising, D.Marpe, H.L.Cycon et al. 2001). The contributions of the proposed coding framework are twofold. First, the EBCOT, which employs an MCABAC scheme, is adopted as the coding engine for both the intra- and the inter-frame coding. EBCOT has proven to provide good compression efficiency and embedded R-D optimization mechanism (D.Taubman 2000). The use of EBCOT in the video coding framework is an extension of the EBCOT application from still image to video compression domain. Second, this framework offers a good platform for the PDM based on the HVS where the MSE can be easily replaced with the PDM in the R-D optimization. The performance of the proposed coding framework is compared against H.264 BP. The results demonstrate
that the proposed coder outperforms the benchmarks in terms of R-D performance on 
*Football* and *Susie*.

Chapter 4 presents the improvement of the MC in the DWT domain based on the 
proposed DWT video coder. First, a review of the MC techniques in the DWT domain is 
given. Second, the LBS MC scheme is used for the proposed coding framework and 
proven to offer consistent improvement in terms of the average PSNR results. For 
*Mobile&Calendar*, the PSNR improvement is up to 1.1 dB. The only exception is 
Football, where a slight drop of PSNR value occurs at low to mid bitrates. The above 
results prove that the LBS efficiently reduces the temporal redundancy in the 
DWT/MC/DPCM coding framework. Third, two adaptive arithmetic coding schemes are 
used to code the motion information. They are the Witten’s algorithm with an adaptive 
model (I.H.Witten, R.M.Neal and J.G.Cleary 1987) and the MCABAC with the contexts 
borrowed from H.264 (D.Marpe, H.Schwarz and T.Wiegand 2003). The comparisons 
show that the results of two schemes are comparable to each other.

Chapter 5 performs the statistical analysis of the inter-frame data. This leads to the 
redesign of the contexts for the inter-frame data coding. The analysis shows that the 
statistical redundancy in the significance coding is much stronger than that in the sign or 
magnitude refinement coding. The calculated mutual information curve for optimal 
context template merging gives the concrete information on how the optimal contexts 
should be formed. The current number of significance contexts of the EBCOT is limited 
to 9. They are replaced with the re-designed 9 optimal contexts. The results of the coder
using old JPEG2000 contexts set for the inter-frame coder are compared against the one using new significance context set. For *Susie* and *Mobile&Calendar*, the new context set is proven to be slightly better. For *Football* and *Balloon&Pops*, the JPEG2000 significance context set still has higher PSNR results. It is believed that the coding efficiency of the significance coding can be improved with an increase of the number of contexts.

Chapter 6 demonstrates the application of a PDM Based on vision model to the intra-coder of the proposed DWT/MC/DPCM video coding framework. The MSE in the R-D computation stage inside the EBCOT coding engine is replaced with the PDM. The analysis shows that the PDM enhances the visual quality of the coded intra-frames. Furthermore, a visibility assessment of the quantization error in the DWT domain is performed to quantitatively measure the effects of the quantization error in the DWT domain to human viewers. The results of the assessment show that the frequency scopes in the data acquired during the experiment have certain agreement with the CSF curve in general trend. However, there are indeed some shifts occurred because of the masking effects. These shifts need more investigation in the future.

In general, the future work will be composed of the following parts. First, the context design of the inter-frame coder needs further investigation. As stated in Chapter 5, the number of contexts is a trade-off between the coding efficiency, the computation complexity and context dilution. More statistical data is needed to verify the context set. Second, the temporal aspect (S.R.Lehky 1985) of the HVS needs to be incorporated into
the proposed coder to further improve the perceived quality of coded videos. The parameters for the HVS model presented in Chapter 6 are for the intra-frame coder based on 5-level DWT decomposition. The parameters need to be calibrated for 3-level DWT. Third, additional scalability features of the proposed coder needs to be addressed in future work. Fourth, the computation complexity of the proposed perceptual video coder needs to be analyzed quantitatively.
A. Context Design of EBCOT

Within the EBCOT scheme adopted by JPEG2000, each subband is partitioned into relatively small code blocks with a size of either $64 \times 64$ or $32 \times 32$. Each code block is labeled as $B_i$ where $i$ is the code block ID. $j$ denotes the sample location which contains coordinate values in both spatial horizontal and vertical directions. $p$ denotes the bitplane. Therefore, if a transform coefficient is represented by sign-magnitude binary format, $\chi[j]$ and $v$ is used to denote the sign and the magnitude, respectively. There is an important value $v^{(p)} = \left\lfloor \frac{v}{2^p} \right\rfloor$, which represents for the quantized coefficient value if the $p$ least significant bits (LSBs) are dropped from the magnitude. It is assumed that $K$ is a number associated with the current code block that $v^{(K)}[j] = 0$ for all $j$ within the current block. Therefore, the coding of the coefficient magnitudes will begin at bitplane $(K-1)$. Before proceeding, there is an important quantity to be defined, which is called “significance” as in the following equation.

\[
\sigma^{(p)}[j] = \begin{cases} 
1 & v^{(p)}[j] > 0 \\
0 & v^{(p)}[j] = 0 
\end{cases} \quad (A.1)
\]

Associated with the definition of $\sigma^{(p)}[j]$ is the notion of a binary “significance state”, $\sigma[j]$. Its value is initialized to 0 and switched to 1 immediately after coding the first non-zero magnitude bit.
Significance coding

The contexts of significance coding are based on the assumption that the coefficient to be coded is still insignificant, i.e. \( v^{(p+1)}[j] = 0 \). The selection among these significance contexts is based on the significance of the eight direct neighbors of the current coefficient. The context label of the current coefficient, \( \kappa^{\text{sig}}[j] \), is determined by three quantities formed by the neighboring coefficients in (A.2) – (A.4). The relationship between these three quantities and the context label is listed as in Table A.2. The symbols \( h \), \( d \) and \( v \) denote the horizontal, vertical, and diagonal direction, respectively.

\[
\kappa^h[j] = \sigma[j_1,j_2-1] + \sigma[j_1,j_2+1] \quad \text{(A.2)}
\]
\[
\kappa^v[j] = \sigma[j_1-1,j_2] + \sigma[j_1+1,j_2] \quad \text{(A.3)}
\]
\[
\kappa^d[j] = \sum_{k_1=\pm1} \sum_{k_2=\pm1} \sigma[j_1+k_1,j_2+k_2] \quad \text{(A.4)}
\]

<table>
<thead>
<tr>
<th>( \kappa^{\text{sig}}[j] )</th>
<th>LL and LH blocks</th>
<th>HL blocks</th>
<th>HH blocks</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \kappa^h[j] )</td>
<td>( \kappa^v[j] )</td>
<td>( \kappa^d[j] )</td>
<td>( \kappa^d[j] )</td>
</tr>
<tr>
<td>8</td>
<td>2</td>
<td>( X )</td>
<td>( X )</td>
</tr>
<tr>
<td>7</td>
<td>1</td>
<td>( \geq 1 )</td>
<td>( X )</td>
</tr>
<tr>
<td>6</td>
<td>1</td>
<td>0</td>
<td>( \geq 1 )</td>
</tr>
<tr>
<td>5</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>0</td>
<td>2</td>
<td>( X )</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>1</td>
<td>( X )</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>0</td>
<td>( \geq 2 )</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Table A.1 Significance Coding Context Determination
In fact, the coefficients in code block are highly skewed to be insignificant. In order to reduce the cost of computation, a so-called run mode significance coding was introduced. The following prerequisites must be fulfilled in order to enter this mode:

1) \( \sigma[j, r] = 0 \) for \( 0 \leq r < 4 \), where \( j_0 = j \) and \( j_r \) is the \( r \)th position from \( j \) along the visiting scan. That is to say, the four consecutive samples from the current coefficient along the scan are insignificant;

2) all four samples must have insignificant neighbors at current moment, i.e.
\[
\kappa^h[j, r] + \kappa^v[j, r] + \kappa^d[j, r] = 0, \quad 0 \leq r < 4;
\]

3) the four samples must belong to the same single stripe column.

A binary output symbol is coded to indicate whether or not all of the four samples are insignificant at this “moment”. A value of 0 means all the four samples remain insignificant while a value of 1 means that at least one of the four samples become significant. This symbol is called “run interruption” symbol and coded with context 9. When run interruption symbol is 1, i.e., one of the consecutive samples become significant, the run length \( r \) must be coded as well, followed by the sign of the first significant sample.

**Sign Coding**

The sign coding primitive is invoked immediately after the current coefficient is found to be significant. It is therefore straightforward to see that the sign coding can be used at most one for each individual coefficient. There are 5 contexts for sign coding in EBCOT,
which employ the redundancy between the signs of neighboring coefficients. There are two quantities defined in (A.5) and (A.6) as an indication of the “net sign bias”.

\[
\chi^h[j] = \chi[j_1, j_2 - 1] \sigma[j_1, j_2 - 1] + \chi[j_1, j_2 + 1] \sigma[j_1, j_2 + 1] \quad (A.5)
\]

\[
\chi''[j] = \chi[j_1 - 1, j_2] \sigma[j_1 - 1, j_2] + \chi[j_1 + 1, j_2] \sigma[j_1 + 1, j_2] \quad (A.6)
\]

The values of above two quantities range between [-2, 2], therefore there are 25 possible values for \(\chi^h[j]\) and \(\chi''[j]\). In order to further reduce the number of contexts to only 5, truncated bias terms are defined in (A.7)-(A.8).

\[
\bar{\chi}^h[j] = \text{sign}(\chi^h[j]) \min\{1, |\chi^h[j]|\} \quad (A.7)
\]

\[
\bar{\chi}''[j] = \text{sign}(\chi''[j]) \min\{1, |\chi''[j]|\} \quad (A.8)
\]

The mapping between the truncated bias terms, the context labels and the so-called sign-flipping factor, \(\chi^{flip}\), are listed in Table A.2. The symbol to be coded is 0 when \(\chi[j] \cdot \chi^{flip} = 1\) and 1 if \(\chi[j] \cdot \chi^{flip} = -1\), respectively.

<table>
<thead>
<tr>
<th>(\bar{\chi}^h[j])</th>
<th>(\bar{\chi}''[j])</th>
<th>(\kappa^{sign})</th>
<th>(\chi^{flip})</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>14</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>13</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>-1</td>
<td>12</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>11</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>10</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>-1</td>
<td>11</td>
<td>-1</td>
</tr>
<tr>
<td>-1</td>
<td>1</td>
<td>12</td>
<td>-1</td>
</tr>
<tr>
<td>-1</td>
<td>0</td>
<td>13</td>
<td>-1</td>
</tr>
</tbody>
</table>
APPENDIX A. CONTEXT DESIGN OF EBCOT

| -1 |  1 | 14 | -1 |

Table A.2 Mapping of the truncated biases, context labels, and sign-flipping factor

Magnitude Refinement Coding

The final type of coding primitives is the magnitude refinement coding. It is used to code the magnitude value of those samples which have already been found to be significant, i.e. \( \sigma^{(p+1)}[j] = 1 \). Here comes another important definition, \( \tilde{\sigma}[j] \), which is employed to denote the value of the significance state variable, delayed by one bitplane. That is to say, if \( \sigma[j] \) is toggled to 1, \( \tilde{\sigma}[j] \) still remains 0 until after the first magnitude bit of the corresponding coefficient has been coded. The mapping of the context labels for magnitude refinement coding and \( \tilde{\sigma}[j] \) is listed as in Table A.3.

<table>
<thead>
<tr>
<th>( \delta[j] )</th>
<th>( \kappa^{ag}[j] )</th>
<th>( \kappa^{mag} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>15</td>
</tr>
<tr>
<td>0</td>
<td>&gt;0</td>
<td>16</td>
</tr>
<tr>
<td>1</td>
<td>( x^a )</td>
<td>17</td>
</tr>
</tbody>
</table>

Table A.3 Magnitude Refinement Coding Context Labels

The above coding primitives are used to code the data of each individual code blocks. This is called tier 1 coding. After tier 1 coding, a Rate- Distortion (R-D) optimization is adopted which determines the required bit rates of each code block under certain bitrate limit for the whole bit stream. In the implementation of JPEG2000 standard, Mean Squared Error (MSE) is used as the distortion measure during the R-D calculation. In the final bit stream, the contributions from different code block also need to be coded. This is implemented via the so-called tag-tree coding (D.S.Taubman and M.W.Marcellin 2002).
B. DWT Video Coders—A Review

Although most of the current international and industry standards of video compression adopt the DCT based techniques, DWT keeps attracting widespread attention from the literature due to the following reasons. Firstly, the block based DCT coders have inevitable blocking artifacts at low and very low bitrates, which is very unpleasant for the subjective scores (M.Yuen and H.R.Wu 1998; BT.500-11 2002). By intuition this problem can be addressed by simply increasing the size of the DCT block. However, this approach means the impractical increase of the computation complexity of DCT, which is a big obstacle for both the hardware and software implementation. Secondly, DWT based image/video embedded coders offer genuine graceful scalability in quality, spatial, and temporal domains without sacrificing the coding efficiency through careful ordering of the data bits. On the other hand, most DCT based coders can only offer layered “scalability” and the introduction of the enhanced layer brings loss of the compression efficiency. The scalability property has become more demanding in recent years especially for web based applications under different or varying bandwidth conditions. Thirdly, the DWT can effectively retain the property in both the original and the frequency domain, as discussed in Chapter 2. As the result it is more adapted to the characteristics of the HVS than the DCT does.

The DWT based image/video coding has its root in the Subband Video Coding (SBC). The SBC was initially developed for speech compression as mentioned by Karlsson and Vetterli (G.Karlsson and M.Vetterli 1988). The theories and the associated techniques
were extended to multi-dimensional signals (M.Vetterli 1984) and tested on image compressions (J.W.Woods and S.D.O'Neil 1986). It was then applied to the Three-dimensional (3D) video coding by Karlsson and Vetterli (G.Karlsson and M.Vetterli 1988). The block diagram of the 3D video analysis framework in that work is illustrated in Fig. B.1, where FIR filters with 2, 5, and 3 taps length were used for temporal, spatially low-pass, and spatially high-pass filtering, respectively. It is noteworthy that one and two levels of Mallat’s decomposition (S.G.Mallat 1989) were separately applied to the temporally high-passed and low-passed signals. Similar scalar quantizations were applied to the output of all 11 subbands followed by the PCM encoding except for band 1, which adopted DPCM encoding.

Lewis and Knowles introduced a video compression scheme using 3D wavelet transforms (A.S.Lewis and G.Knowles 1990). To the best of our knowledge, this is the earliest work which applied DWT for 3D SBC. The authors adopted the subband analysis approach as
in Fig. B.2 where each individual picture was applied with the 3-level Mallat’s Decomposition using Daubechies 4/4 orthonormal wavelet filters (I. Daubechies 1988) for spatial decomposition, then Haar filter bank was applied to each of two such consecutive spatially-decomposed pictures for temporal decomposition. After the 1\textsuperscript{st} level temporal decomposition, the subbands containing the high temporal frequencies were applied with the 2\textsuperscript{nd} level temporal decomposition. The 3\textsuperscript{rd} level temporal decomposition came afterwards with a balanced tree structure. The down-sampling factor of 2:1 was used during all subband splitting procedures. The characteristics of the Human Visual System (HVS) were incorporated into the design of the quantizer by using an extension of Marr’s technique in the time dimension. It was assumed that the motion or change between frames produced “edge” in time, while other information such as small motions, noise,
lighting changes and texture motion contributed to minor information. Therefore, for any coefficient in Fig. B.2 c to h, its value would be quantized to zero if the corresponding coefficient in Fig. B.2 a and/or b were zero. The authors further claimed that this scheme was computationally cheaper than DCT/MC/DPCM.

Ohm (J.-R.Ohm 1992) proposed a temporal domain sub-band video coding with motion compensation scheme (MC-SBC) for inter-frame coding. The basic idea was to apply MC before subband coding for inter-frames. QMF filter bank was used for subband decomposition. The experimental results were presented in terms of PSNR values between the variants of the proposed inter-frame coders and the benchmarks. For CIF video format, the PSNR values of the reconstructed videos coded by the MC-SBC inter-frame coder were significantly higher than the MC-DPCM coder, and much higher than those using SBC schemes without MC. This investigation gave a good clue for the use of MC in subband video coding.

Woods and Naveen suggested a theoretical framework to allocate bits in subband video coding and compared different filter sets (J.W.Woods and T.Naveen 1992) among QMF (J.D.Johnston 1980), perfect reconstruction linear phase filters, a subjectively optimized filter (Kronander 1989), and two wavelet filters (A.N.Akansu, A.Haddad et al. 1992). Their bit allocation scheme considered the variance after quantization within and across the subbands and the basis functions of the filters. The authors concluded that QMFs and wavelet filter sets produced better results over others. And they observed that although the wavelet filter has shorter taps, they were rated subjectively equivalent to the QMFs at
all the bit rates they studied.

Zhang and Zafar introduced a motion compensation scheme in the DWT domain for the inter-frame coding, which was the so-called multi-resolution motion compensation (MRMC) (Y.-Q.Zhang and S.Zafar 1992). This work was based on an important assumption that the motion activities for a particular subband at various resolutions were highly correlated since they actually specified the same global motion structure at various resolutions and frequency ranges within the wavelet decomposition framework. Daubechies orthonormal wavelet bases (I.Daubechies 1988) were used for DWT analysis and synthesis purpose. In the MRMC scheme, the motion vectors were first calculated for the lowest resolution subbands. In fact, motion information is a more precise term than motion vector here because the computation happens in the wavelet but pixel domain. Then, motion vectors at the subbands of the higher resolutions were refined using the motion information obtained at the lowest resolution. The sizes of the motion blocks could be represented by \( p \cdot 2^{M-m} \) where \( p \) is the size of the block at the lowest resolution, \( M \) the total number of resolutions, and \( m \) the index of the resolution where the block belongs to, respectively. It was thus straightforward to see that the motion block size was proportional to the resolution level within the decomposition structure as illustrated in Fig. B.3. This variable block size made the MRMC scheme distinct from the work in (K.M.Uz, M.Vetterli et al. 1991) where the block size was same across different resolutions. Within Fig. B.3 the blocks belonging to the same wavelet tree were denoted with the gray color, despite that they were distributed at different resolutions and orientations. The motion vectors of the lowest resolution had the same value among the
DC band, HL, LH and HH band. The motion vectors of all the blocks within other subbands were calculated separately using Formula B.1 where $V_{DC}$ was the motion vector of the block in the DC band. $\Delta_o(\delta x, \delta y)$ and $o$ represented for the incremental motion vector found by a full search in the search window $\Omega$ and the orientation of the subband, respectively. The calculation of $\Delta_o(\delta x, \delta y)$ is demonstrated as in Formula B.2,

$$V_o(x, y)^{(m)} = V_{DC}(x, y) \cdot 2^{M-m} + \Delta_o(\delta x, \delta y)$$  \hspace{1cm} (B.1)

where $(x_1, y_1)$ is the most left-upper coefficient of each block with gray color in Fig. B.3, $X$ and $Y$ the width and height of the search window at different resolution, respectively. The authors further classified the variants of the MRMC scheme into four subcategories:

1) wavelet decomposition $\rightarrow$ multi-resolution motion compensation $\rightarrow$ multiscale quantization $\rightarrow$ entropy encoder;
Fig. B.3 Variable Block-Size Multiresolution Motion Estimation

2) wavelet decomposition $\rightarrow$ multiresolution motion compensation $\rightarrow$ DCT $\rightarrow$ uniform quantization $\rightarrow$ entropy encoder;

3) motion compensation $\rightarrow$ wavelet decomposition $\rightarrow$ multiscale quantization $\rightarrow$ entropy encoder;

4) motion compensation $\rightarrow$ wavelet decomposition $\rightarrow$ DCT $\rightarrow$ uniform quantization $\rightarrow$ entropy encoder.

The authors compared the PSNR results among the above four video coders and concluded that the 1$^{st}$ variant significantly outperformed others.

Yuan and Chan suggested an Overlapped MRMC for DWT domain motion compensation (Y.Yuan and C.W.Chan 2000). It was aimed to address the effect of block boundary discontinuities due to the nature of BMA in MRMC. It assumed that the predictions of one block were determined by a set of motion vectors assigned to neighboring blocks. As
the result, each pixel in one block of motion compensated frame was a weighted sum of a set of prediction values from the reference reconstructed frame. The authors reported the PSNR results of prediction frames with about 0.6dB and 1dB higher than those of MRMC when MC refinement was disabled and enabled respectively.

Cafforio et al. (C.Cafforio, C.Guaragnella et al. 1994) quantitatively studied the motion compensation within the subband video coding framework. They noticed the so-called shift-variant phenomena and compared different motion compensation schemes within a subband video coder. They mathematically proved that: 1) motion estimation and compensation could be carried out on the full resolution images; and 2) it was possible to correct motion effects directly within the subbands. The second point was of practical meaning because it indicated the feasibility of the layered transmission required by broadcasters. Unfortunately, they didn’t offer a solution how to address the second issue.

Taubman and Zakhor introduced a multirate 3-D SBC scheme with camera pan compensation in order to generate a single embedded bitstream supporting multiple decoder resolutions and a wide variety of bitrates. The camera pan compensation was applied before 3D SBC in order to pre-distort the video sequence as to eliminate the effects of camera pan motions. Four spatial decomposition levels were applied to the images followed by three-level temporal decomposition. The next level temporal decomposition was always applied to the low-temporal frequency subband derived from the last level temporal decomposition.
Ngan and Chooi suggested a 3D subband approach for very low bitrate video coding (K.N.Ngan and W.L.Chooi 1994). It first split the input video into four temporal subbands using Johnston 10-tap QMF filters (J.D.Johnston 1980) or 4-tap QMF filters (A.N.Akansu, A.Haddad and H.Caglar 1992). Motion detection was then performed on these temporally-analyzed subbands on the macroblock basis. If a macroblock was found to contain temporal activity (TA) or motion, the four associated macroblocks in the temporal subbands were further decomposed into 16 spatiotemporal subbands. Otherwise, the macroblock was not coded at all. The macroblocks derived from the lowest temporal-frequency subband were coded using the SIM3-like coder (ITU-T SGXV 1993) while the others were coded using VQ. It was worth mentioning that the leaf-subbands in the decomposition tree had the same depth from the root, which was the original image. This symmetry made this scheme distinct from most of the 3D-SBC scheme in the literature.

Podilchuk et al. introduced a 3D subband video coding scheme (C.I.Podilchuk, N.S.Jayant et al. 1995) based on the same 3D decomposition structure of (G.Karlsson and M.Vetterli 1988) as illustrated previously in Fig. B.1. Two-tap Haar and 10-tap QMF filter bank (J.D.Johnston 1980) was used for temporal and spatial analysis, respectively. The authors noticed that subband 8, which corresponded to the high temporal / low spatial frequency, acted like a motion detector. In the mean time, many of the spatial details were contained by the low spatiotemporal subbands (subband 1-4). Therefore, accurately encoding subband 1-4 and 8 was of importance. Different encoding methods were adopted to deal with high, mid, and low bitrates. For high bitrate, an adaptive
differential pulse code modulation (ADPCM) was used to code the lowest frequency band (subband 1), where the prediction gain was significant and subband 1 was also used to determine the areas to be encoded in the higher frequency bands. PCM was used to encode the higher frequency bands where the prediction gains were low. For mid bit rate, ADPCM was used for the lowest subband where the prediction gain was significant and a geometric vector quantization (GVQ) scheme was used for the higher subbands. At the lower bitrates, since the bitrate could not meet the requirements needed for an effective DPCM coder, an unbalanced tree-structure VQ (UTSVQ) scheme was used to encode subband 1 and GVQ was employed to encode the high-frequency subbands. The authors further concluded that the advantages of a 3-D video coder were two fold: firstly, any type of apparent motion was systematically captured in the appropriate subband while in contrast the traditional MC approach based on block matching algorithm (BMA) was tailored to translational motion of rigid objects; secondly, the subband framework could adapt to the amount of motion. When the corresponding subband contained no significant motion data, it would simply not be encoded therefore the precious bits were saved.

Ito and Farvardin (H.Ito and N.Farvardin 1995) proposed a novel wavelet video coder using entropy-constrained trellis coded quantization (ECTCQ), where the original design of ECTCQ could be traced back in (T.R.Fischer and M.Wang 1992). The video sequence was first spatially decomposed into several subbands and motion compensated in the frequency domain. The motion estimation was performed on the basis of macroblock between the current frame and the reconstructed previous frame. The reconstructed previous frame was passed through analysis filters corresponding to the each individual
subband in the Mallat’s decomposition structure without any decimation, shifted by the amount of motion and subsampled. For each subband the motion compensated samples were subtracted from the current samples to obtain the inter-frame prediction error for that band. The prediction errors were quantized using the ECTCQ in each individual subband.

Chen and Pearlman introduced a 3D subband video coder using the zero-tree method (3D-ZT). Their subband decomposition structure is illustrated in Fig. B.4. In fact, it was very close to that used in (G.Karlsson and M.Vetterli 1988) with the exception that a second layer of temporal decomposition was employed for the Low temporal – Low Horizontal – Low Vertical filtered-decimated output. Zero-tree coding belongs to the family of progressive transmission techniques, where the initial segment of a bitstream contains necessary information to reconstruct an image at coarse resolution or quality. When more bits arrive from the same bitstream source, the decoder is able to refine the image gradually. EZW (J.M.Shapiro 1993), SPIHT (A.Said and W.A.Pearlman 1996), and EBCOT (D.Taubman 2000) are the typical progressive coders for the still image compression. 3D-ZT could be regarded as an extension of SPIHT from 2D to 3D application. The same 9-tap QMF filter was used for both the temporal and spatial filtering. 4 levels of temporal decomposition were adopted. As the result, the input sequence was segmented into groups of 16 frames (GOF) and they were decomposed and encoded one another. The authors reported the results were comparable with MPEG-2. However, there would be discontinuities existing around the boundaries of GOFs. And these discontinuities contributed to the so-called temporal fluctuation.
Meyer et al. proposed a novel MC scheme of wavelet coefficients for very low bit rate video coding (F.G.Meyer, A.Averbuch and R.R.Coifman 1997). It was based on the fact that wavelet expansions were highly dependent on the alignment of the signal and the discrete grid chosen for the analysis. In order to perform the MC in the wavelet domain, one needed to correctly predict the coefficients of a signal that had been translated. The authors drew two conclusions from analysis: first, the lowpass filtered signal was smooth and it was possible to accurately interpolate the lowpass coefficients of the shifted signal from the lowpass-decimated coefficients of the unshifted signal; second, it was impossible to accurately interpolate the highpass coefficients of the shifted signal from the highpass-decimated coefficients of the unshifted signal. In order to overcome the above prediction difficulty for DWT coefficients in highpass subbands, the authors suggested a specially-designed wavelet coefficients pyramid within which all the high
pass coefficients were derived without decimation as illustrated in Fig. B.5.

![Diagram of DWT Coefficients Pyramid for MC](image)

**Fig. B.5** Computation of DWT Coefficients Pyramid for MC (by Meyer et al.)

The original signal was lowpass filtered twice followed by 2:1 decimation, which was different from the normal lowpass-decimation operation; and the signal was highpass filtered without decimation. As could be seen in Fig. B.5, one would get the decimated twice low pass filtered subband at the lowest resolution and a group of undecimated highpass filtered subbands. The motion estimation and compensation of an image \(I(t+1)\) based on the reference image \(I(t)\) began with the building of the aforementioned pyramid of \(I(t)\). Then the pyramid of \(I(t+1)\) was predicted from that of \(I(t)\) by MC at each level of pyramid in DWT domain. The authors further compared the proposed MC scheme with the traditional BMA and reported that the suggested one outperformed the other in terms of both the subjective and the objective quality.
Kim et al. proposed a two-stage MRME algorithm (S.Kim, S.Rhee, J.G.Jeon et al. 1998) for DWT based inter-frame coding. In the first stage, the authors adopted a MRME scheme similar with that in (Y.-Q.Zhang and S.Zafar 1992) with the following distinctions: firstly, the information of all subbands in the lowest resolution was utilized to determine the motion vectors for the low-pass subband, which were also used as the base motion vectors for other subbands, whereas only the information of the low-pass subband was used in the original MRME scheme (Y.-Q.Zhang and S.Zafar 1992); secondly, the absolute values of all wavelet coefficients in the motion block (MB) were tested against the predetermined threshold in the subband. If the test was positive, the MB was filled with all zeros with no motion estimation at all. In the second stage of this two-stage MRME scheme, a bit allocation between the motion information and the residual error was accomplished by reducing the number of motion vectors in the uniform area. This was based on a bottom-up construction of the quadtree, which was also named by some researchers in the literature as wavelet tree. Four nodes in the quadtree would be merged into a larger node if this resulted in the saving of the total bitrate.

Ohm reported a video coding scheme using layered Vector Quantization (VQ) and Subband Coding (SBC) (J.-R.Ohm 1993). Along with a companion paper published one year later (J.-R.Ohm 1994), the author highlighted the concept of Temporal Filtering along The Motion Trajectory or Direction, which was often mentioned by other researchers. Besides the VQ, 3-D SBC scheme with MC prediction (MC-SBC) was compared against the 3-D SBC without MC, the traditional MC prediction, and the
traditional MC Interpolation as inter-frame coders. The MC-SBC performed MC before the subband decomposition along the spatial and temporal directions. The analysis demonstrated that MC-3D SBC outperformed the others in terms of PSNR values. In addition, the main advantages of MC-3D SBC were the high energy compaction capability and the nonrecursive decoder structure. However, there existed a problem in the MC-SBC scheme, which was the so-called non-connection problem, as analyzed by Ohm in his work in detail (J.-R.Ohm 1993). Assuming 2-tap filter was used along the temporal direction (although more complicated case was possible), and $A$ and $B$ representing two original-image frames. In the case of a spatially variant motion-vector field, which was the generic scenario, under MC-SBC scheme, it was possible that some pixels of frame $A$ not be filtered at all, while others might be filtered twice or even more times. The above not filtered pixels are called not-connected pixels. Ohm provided the solution to address the above problem: shifting the coordinates to change the “not connected” areas into “connected” ones. When no-connection or double-connection was detected, substituting the coefficient values in the low-temporal bands by the values from frame $B$.

Joo and Kikuchi proposed a DWT domain MC scheme based on the concept of wavelet tree. This algorithm sacrificed the prediction accuracy for simplicity and fast execution speed (S.Joo and H.Kikuchi 2000).

Qiu et al. (T.Qiu, X.Wu et al. 2000) reported a 3D DWT video coder driven by a non-dyadic 3D wavelet decomposition and high-order Markov context modeling for adaptive
arithmetic coding. MC was not performed in this structure because the authors observed that conventional MC prior to 3D DWT tended to spread rather than pact error energy along object edges, thus hindered the subsequent context-based entropy coding.

Heising et al. introduced a MC/DWT/DPCM scheme (G. Heising, D. Marpe, H. L. Cycon et al. 2001) which adopted new components in motion estimation and compensation, wavelet filter design, and entropy coding, respectively, for low bitrate video coding. A new adaptive MC technique using image warping and overlapped block motion compensation (OBMC) was proposed for MC in the spatial domain. This adaptive MC scheme had the advantage of representing more complex motion than the normal BMA. A one parametric family of biorthogonal infinite impulse response (IIR) wavelet filters was adopted for both the spatial decorrelation of the intra- and inter-frame residue data. A pre-coding scheme of ‘partitioning, aggregation and conditional coding’ (PACC) was used before the final arithmetic coding. The experimental results demonstrated a 1.0-2.3 dB PSNR improvement in comparison to those of the H.263+ test model TMN10 using advanced coding options. It was also mentioned by other people (D. Lazar and A. Averbuch 2001) that the performance of PACC outperformed version 5.1 (VM) of MPEG-4 and it was proposed for the ITU-T H.26L. The intra coding mode also outperformed the JPEG2000 in terms of the coding efficiency.

Lazar and Averbuch reported a wavelet-based video coder via bit allocation (BA). The proposed BA method utilized a Lagrangian multiplier technique to determine the optimal bit rate for each frame of a given group of frames (GOF). The coder consisted of two
layers: the first one was merely a combination of wavelet still compression and optimal BA (WCBA). It did not utilize MC interpolation or 3-D wavelet decomposition. When heavy motion was detected, the second layer was invoked which utilized wavelet compression, optimal BA, the MC with BMA and error correction (WCBAM). It is worth mentioning that the MC was performed in the spatial domain.

Seigneurbieux and Xiong reported a 3-D wavelet video coder with rate-distortion optimization (P.Seigneurbieux and Z.Xiong 2001). They extended the concept of space-frequency quantization (SFQ) (Z.Xiong, K.Ramchandran et al. 1997) from 2-D to 3-D. A lifting based 3-D wavelet transform was employed to process one part of the sequence at a time continuously, therefore eliminating the boundary effects over GOPs. Within 3-D SFQ a spatiotemporal tree-pruning process was adopted based on R-D optimization. Afterwards a uniform quantization and progressive 3-D entropy coder were used. The authors concluded that this coder, even without MC, outperformed MPEG-4 and other 3-D DWT video coders for most sequences at the same bit rate.

Bottreau et al. introduced a fully scalable 3D Subband Video Codec (V.Bottreau, M.Bénetière et al. 2001). Groups of frames were first temporally filtered using Motion-Compensated Temporal Filtering (MCTF) followed by spatial wavelet decomposition. Haar filters and BMA were used for temporal filtering and MC respectively. A SPIHT-alike coder named Fully Scalable Zerotree Coder (FSZC) in conjunction with arithmetic coder provided the scalability in terms of both the temporal, spatial and SNR, or quality.
Moyano et al. proposed a novel 3D wavelet transform decomposition for video compression (E.Moyano, F.J.Quiles et al. 2001) given a name by 3D-V. The core idea was a new temporal decomposition approach as illustrated in the following Fig. B.6 where frames along the temporal direction were partitioned into groups of 4 frames. The temporal decomposition was started by apply wavelet filter bank to the first 4 frames in the sequence and derived 2 low and 2 high frequency bands, respectively. Then the resulting 2 low frequency bands and 2 new frames from the original sequence were used to perform the next temporal decomposition. In this way, the memory requirement was limited to 4 frames at any moment and the latency was a constant which was irrelevant to the number of frames within Group of Frame (GOF). Different efforts were carried out in the literature trying to address the bottleneck of coding efficiency of 3D DWT video
Recently, Secker and Taubman introduced a Lifting-based Invertible Motion Adaptive Transform (LIMAT) framework (A.Secker and D.Taubman 2003) for highly scalable 3D video compression. The authors used motion compensated lifting steps to implement the temporal wavelet transform, which preserves the invertibility because of the nature of lifting. This scheme could be adapted to different motion models. The work was based on a few observations in the 3-D DWT coding literature. Firstly, without MC, temporal filtering produced visually disturbing ghosting artifacts in the low-pass temporal subbands. Secondly, for the “frame-warping” approach where frames were invertibly warped prior to the 3D DWT (D.Taubman and A.Zakhor 1994), the localized expansion and contraction were not successful to be represented. Thirdly, the so-called “block displacement” methods such as the work of (J.-R.Ohm 1994) was unable to deal with the localized expansion and contraction very well. Fourthly, under the approaches of “frame-warping” and “block displacement”, temporal filters with tap lengths longer than that of Haar wavelet filter did not bring significant performance improvement. The LIMAT framework was essentially a construction of invertible motion adaptive temporal transform, within which the temporal DWT was represented by the lifting with motion-compensated lifting steps. The authors demonstrated the performance of the LIMAT framework in terms of the adaptation to a wide variety of motion models and temporal filters. Among them are the deformable mesh motion model and the Daubechies 5/3 filter bank. They showed superior coding gain in terms of PSNR values for the reconstructed videos. A companion paper was published one year later (A.Secker and D.Taubman
2004). In this work, the idea of scalable motion coding was combined with the aforementioned LIMAT framework. Motion parameters were usually coded losslessly as side-information within most video coders. However, the tradeoff between the volume of motion information and the coding efficiency was indeed an issue for video coders. In particular, at low bitrates the motion information could consume a significant amount of the available bits where high-precision motion information was of little benefit (A. Secker and D. Taubman 2004). The authors observed that the video distortion introduced by scaling the motion information after compression was approximately linearly related to the motion parameter MSE. The authors proposed two rate-allocation strategies to achieve this scalable-motion-information target. At low bitrates, significant gains were observed in comparison to lossless motion information coding.

Generally speaking, the DWT based video coders can be classified into three general categories. They are 3D DWT video coder, MC/DWT/DPCM video coder, and DWT/MC/DPCM video coder. The 3D DWT video coders do not employ the inter-frame coding. Instead they apply the temporal decomposition to utilize the spatiotemporal redundancy. The application of motion compensation before the temporal filtering stage proved to be a promising to achieve high compression efficiency, and this approach was named as motion compensated temporal filtering (MCTF) based 3D DWT video coding. MC/DWT/DPCM is somehow similar with the widely used DCT/MC/DPCM video coding structure with DCT replaced with DWT. It performs ME and MC in the pixel domain first. Then the residue image is applied with the wavelet transform followed by the DPCM encoding. The DWT/MC/DPCM approach, on the other hand, applies wavelet
APPENDIX B. DWT VIDEO CODERS —— A REVIEW

transform to the input images before all the other procedures. Then the MC is performed in the transform or frequency domain. The DPCM encoding is applied to the residue image after MC. This approach is adopted by this thesis.
C. Introduction to \textit{UEGk} Binarization

The \textit{UEGk} binarization is constructed for motion vectors (MVs). The horizontal and vertical component of each individual MV is binarized according to the same rule, respectively. Each UEGK binary string consists of a TU prefix and an EGk suffix. Before going further, it is helpful to define TU code and EGk code first using C-style pseudo code.

1) \textbf{TU code}:

\begin{verbatim}
unsigned int TU(unsigned int x, unsigned int S)
{
    assert( x >=0 && x<=S);
    if(x <S)
        return x “1” bits plus a terminating “0” bit
    else
        return x “1” bits
}
\end{verbatim}

2) \textbf{EGk code}:

\begin{verbatim}
void EGk ( unsigned int x, unsigned k)
{
    while(1) {
        //unary prefix part of EGk
\end{verbatim}
If \( x \geq (1 \ll k) \) \{

\begin{align*}
& \text{put (1)} \\
& x = x - (1 \ll k) \\
& k++
\end{align*}
\}

} else {

\begin{align*}
& \text{put (0) // terminating “0” of prefix part} \\
& \text{while (k--) // binary suffix part of EGk} \\
& \text{Put ((x\gg k) \& 0x01)} \\
& \text{break}
\end{align*}

})

}

In the implementation, \( S \) and \( k \) in the above pseudo code is 9 and 3, respectively. The whole \( UEGk \) code procedure is described by the following pseudo code:

if \( (MV == 0) \)

output only “0”

else

{

//Construct prefix part using TU binarization with \( S = 9 \)

\( TU(\min(\|MV\|,9),9) \)

//Construct suffix part using EG3 binarization
if( |MV| ≥ 9 )

    EGk(|MV| − 9,3)

    // append sign

    append “1” for negative or “0” for positive

}
'Bibliography


