8-2-2003

Motion Estimation and Compensation in the Redundant Wavelet Domain

Suxia Cui

MOTION ESTIMATION AND COMPENSATION IN THE REDUNDANT WAVELET DOMAIN

By

Suxia Cui

A Dissertation
Submitted to the Faculty of
Mississippi State University
in Partial Fulfillment of the Requirements
for the Degree of Doctor of Philosophy
in Computer Engineering
in the Department of Electrical & Computer Engineering

Mississippi State, Mississippi

August 2003

MOTION ESTIMATION AND COMPENSATION IN

THE REDUNDANT WAVELET DOMAIN

By

Suxia Cui

Approved:

James E. Fowler
Associate Professor of Electrical &
Computer Engineering
(Major Professor and Dissertation
Director)

Robert J. Moorhead, II
Professor of Electrical &
Computer Engineering
(Committee Member)

Nicholas H. Younan
Professor of Electrical &
Computer Engineering
Graduate Coordinator of Electrical &
Computer Engineering
(Committee Member)

Thomas Philip
Professor of
Computer Science
(Committee Member)

A. Wayne Bennett
Dean of the College of Engineering

Name: Suxia Cui

Date of Degree: Aug. 2, 2003

Institution: Mississippi State University

Major Field: Computer Engineering

Major Professor: Dr. James E. Fowler

Title of Study: MOTION ESTIMATION AND COMPENSATION IN THE REDUNDANT WAVELET DOMAIN

Pages in Study: 101

Candidate for Degree of Doctor of Philosophy

Despite being the prefered approach for still-image compression for nearly a decade, wavelet-based coding for video has been slow to emerge, due primarily to the fact that the shift variance of the discrete wavelet transform hinders motion estimation and compensation crucial to modern video coders. Recently it has been recognized that a redundant, or overcomplete, wavelet transform is shift invariant and thus permits motion prediction in the wavelet domain.

In this dissertation, other uses for the redundancy of overcomplete wavelet transforms in video coding are explored. First, it is demonstrated that the redundant-wavelet domain facilitates the placement of an irregular triangular mesh to video images, thereby exploiting transform redundancy to implement geometries for motion estimation and compensation more general than the traditional block structure widely employed. As the second contribution of this dissertation, a new form of multihypothesis prediction, redundant wavelet multihypothesis, is presented. This new approach to motion estimation and compensation produces motion predictions that are diverse in transform phase to increase prediction accuracy. Finally, it is demonstrated

that the proposed redundant-wavelet strategies complement existing advanced video-coding techniques and produce significant performance improvements in a battery of experimental results.

DEDICATION

To my parents, Jin & Lijuan, my husband, Yonghui, and my son, Steven.

ACKNOWLEDGMENTS

I would like to thank my advisor, Dr. James E. Fowler, for his insight of choosing wavelet-based video coding as a research topic, for his patient in helping me solving problems, for his enthusiasm toward academic excellence, and for his encouragement which guided the completion of this dissertation. All my committee members—Dr. Robert J. Moorhead, Dr. Nicolas H. Younan, and Dr. Thomas Philip—have given me precious advice throughout the period during which this research was being done. I greatly appreciate their time and effort to serve on my committee.

I am grateful for financial support from the National Science Foundation (NSF) and the U.S. Naval Research Laboratory–Stennis Space Center (NRL-SSC), as well as the research facilities provided by the Engineering Research Center of Mississippi State University.

I thank my parents, for their love and care throughout all these years of my study. They set for me an excellent model for doing research and encouraged me to fulfill my potential in my academic career. I am gratefully indebted to my husband, Yonghui. To him I offer my deepest appreciation for his invaluable support, for his constant companion, and for his sharing with me the happy times as well as the difficult times during the course of my Ph.D. Finally, I thank my son, Steven, for the joy and inspiration he brought to me.

TABLE OF CONTENTS

CHAPTER Page

LIST OF TABLES

LIST OF FIGURES

FIGURE Page

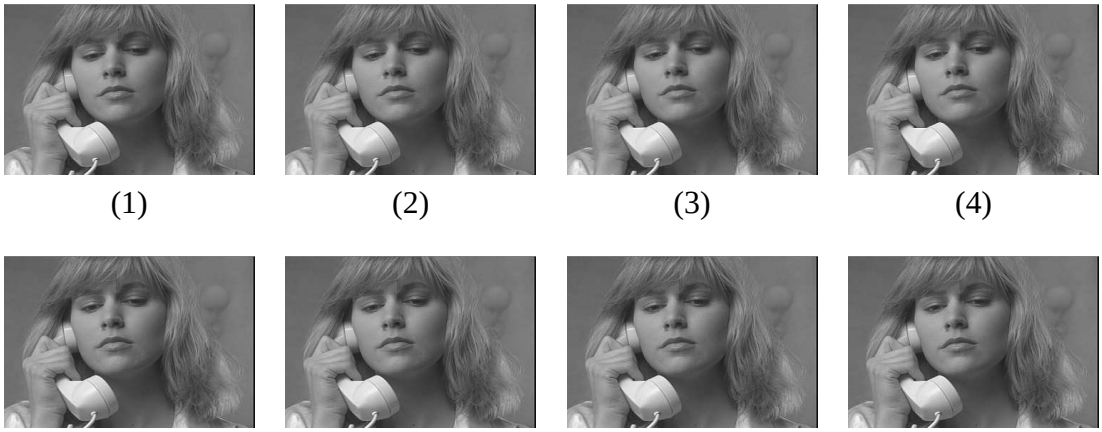
FIGURE Page

CHAPTER I

INTRODUCTION

Over the last several decades, researchers have searched for efficient ways to compress, or code, video sequences. The key aspect of this search centers on decorrelation. A sequence of images is highly correlated both temporally as well as spatially. That is, temporal correlation is evident in the fact that subsequent frames in a video sequence usually appear almost identical. In most cases, only small portions of the scene change from frame to frame. For example, the sequence "Susie" (Fig. 1.1) contains a person talking on the phone with relatively little movement. Even in the high-motion sequence "Football" (Fig. 1.2), the players are running and diving, but the background does not change. In the sequence "Coastguard" (Fig. 1.3), although the background is moving, the main object, a yacht, remains in the center of the scene. The sequence "Mother & Daughter" (Fig. 1.4) is a video-conference sequence with only minor movement since both the background and the position of the two persons are unchanged throughout much of the time.

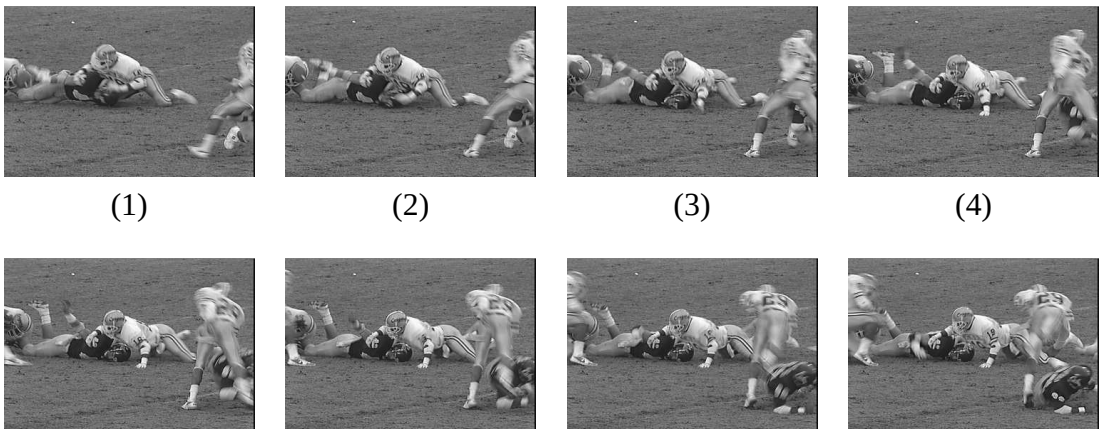
To decorrelate a video sequence temporally, modern video coders employ motion estimation and motion compensation (ME/MC). ME/MC forms a prediction of the current frame using the frames which have been already encoded. Consequently, one needs to transmit the corresponding residual image instead of the original frame, as well as a set of motion vectors which describe the scene motion as observed at the encoder. Since the residual frame typically contains much less signal energy than the original frame, and since the motion vectors are relatively few, the total bit rate to code the motion-estimated frame is usually much less than to code each frame as a still image.



(1) (2) (3) (4)

(5) (6) (7) (8)

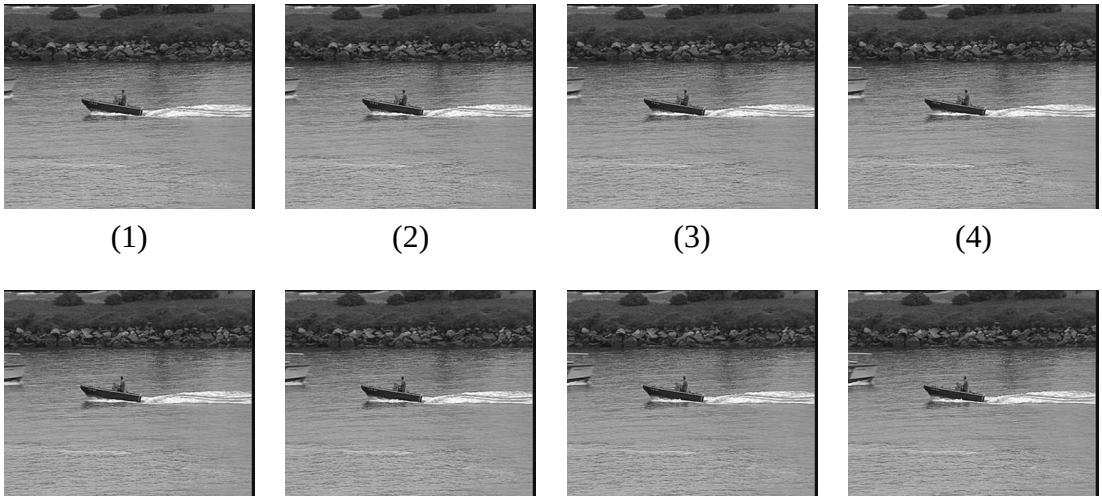
Figure 1.1: First 8 frames of the “Susie” sequence.



(1) (2) (3) (4)

(5) (6) (7) (8)

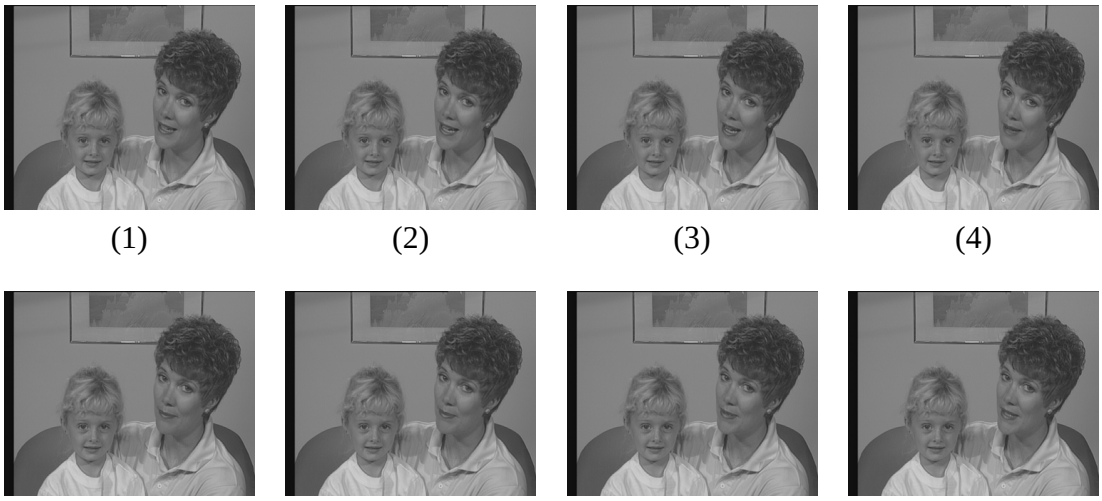
Figure 1.2: First 8 frames of the “Football” sequence.



(1) (2) (3) (4)

(5) (6) (7) (8)

Figure 1.3: First 8 frames of the “Coastguard” sequence.



(1) (2) (3) (4)

(5) (6) (7) (8)

Figure 1.4: First 8 frames of the “Mother & Daughter” sequence.

A number of motion-estimation (ME) algorithms have been developed in order to provide efficient prediction of scene motion between frames. ME schemes can generally be categorized as either feature matching or region matching. Feature-matching ME is based on tracking specific image features (e.g., edges); however, the region-tracking methods are used almost exclusively in modern coders. The most widely used region-matching technique is block matching, in which the current image is divided into small blocks. The previous frame, called the reference frame, is searched for the best-matching block for a given block in the current frame, and the resulting motion vector, (Δ_x, Δ_y), indicates the position of the best-matching block. To limit the computational complexity of the ME process, the search is usually limited to some window surrounding the block position in the reference frame. The procedure of block matching is illustrated in Fig. 1.5 and the calculation of the residual image is

$$\mathrm{Diff}(x, y, t, \Delta t) = f(x, y, t) - f(x + \Delta_x, y + \Delta_y, t - \Delta t), \tag{1.1}$$

where $\mathrm{Diff}(x, y, t, \Delta t)$ denotes the calculated residual image at position (x, y) in a time period Δt from time t, while $f(x, y, t)$ denotes the image value at position (x, y) and time t. This block-based ME/MC approach to video coding was first introduced in [1].

After a video sequence has been decorrelated temporally, there usually exists a great deal of correlation between pixels of the same frame. To reduce this spatial correlation, modern video coders perform a reversible transformation in each residual image such that, in the transform domain, the energy of the image is relocated to an easily coded form. There are several methods to spatially transform an image, such as the Discrete Fourier Transform (DFT), the Discrete Cosine Transform (DCT), and the Discrete Wavelet Transform (DWT). Among them, the DCT is the most widely used transform because of its fast implementation, its early development, and its extensive use in still-

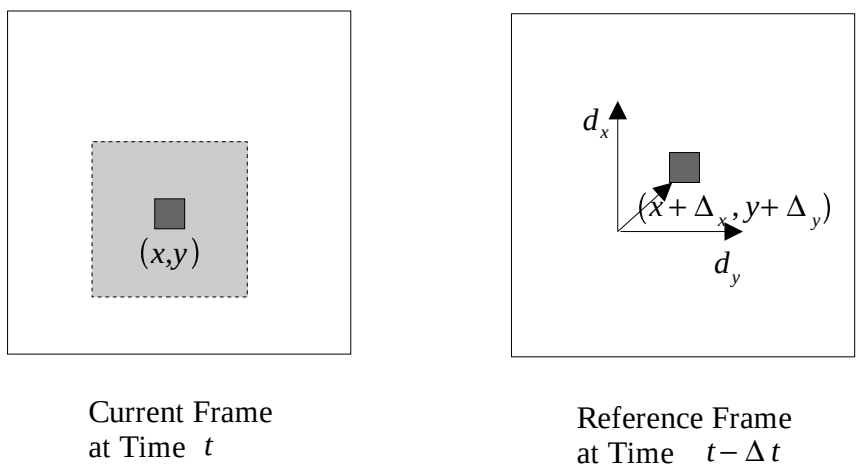


Figure 1.5: The block-matching algorithm. The dashed block shows the search window.

image compression. The traditional hybrid-coding architecture, which features ME/MC followed by a DCT, is widely employed in modern video-compression systems and an integral part of standards such as H.261 [2], MPEG-1 [3], MPEG-2 [4], H.263 Version 1 [5], H.263 Version 2 (H.263+) [6], and MPEG-4 [7]. A diagram of this traditional architecture is shown in Fig. 1.6.

However, given the promising performance of recent wavelet-based still-image compression algorithms, such as set partitioning in hierarchical trees (SPIHT) [8], there has recently been interest in deploying ME/MC within such algorithms to produce wavelet-based video coders. It is hoped that wavelet-based video coding can not only increase coding efficiency, but also introduce a high degree of scalability into the coding scheme such that one compressed representation can be decoded at a variety of rates and fidelities.

Briefly, a DWT is a multiresolution transform that uses the successive application of filters to produce low-resolution and high-resolution components, or subbands, of the original signal. For 2D images, a DWT produces a baseband (a low-resolution approximation to the image) and a variety of horizontal, vertical, and diagonal subbands of increasing resolution, as illustrated in Fig. 1.7. We can see that most of the the energy in DWT-domain coefficients is packed into the lower-resolution bands. Based on this property, a number of effective still-image coders have been devised, of which one of the most popular is the SPIHT coder [8]. In SPIHT, all coefficients are processed in a parent-offspring structure of hierarchical trees as illustrated in Fig. 1.8. SPIHT uses the fact that regions of low energy in a given subband can predict even larger regions of low energy in higher-resolution subbands for efficient coding.

The most straightforward way to replace the DCT with a DWT in a typical video coder is to perform ME/MC in the spatial domain and to calculate a DWT on the resulting residual image, resulting in a system as shown in Fig. 1.9. It has long been

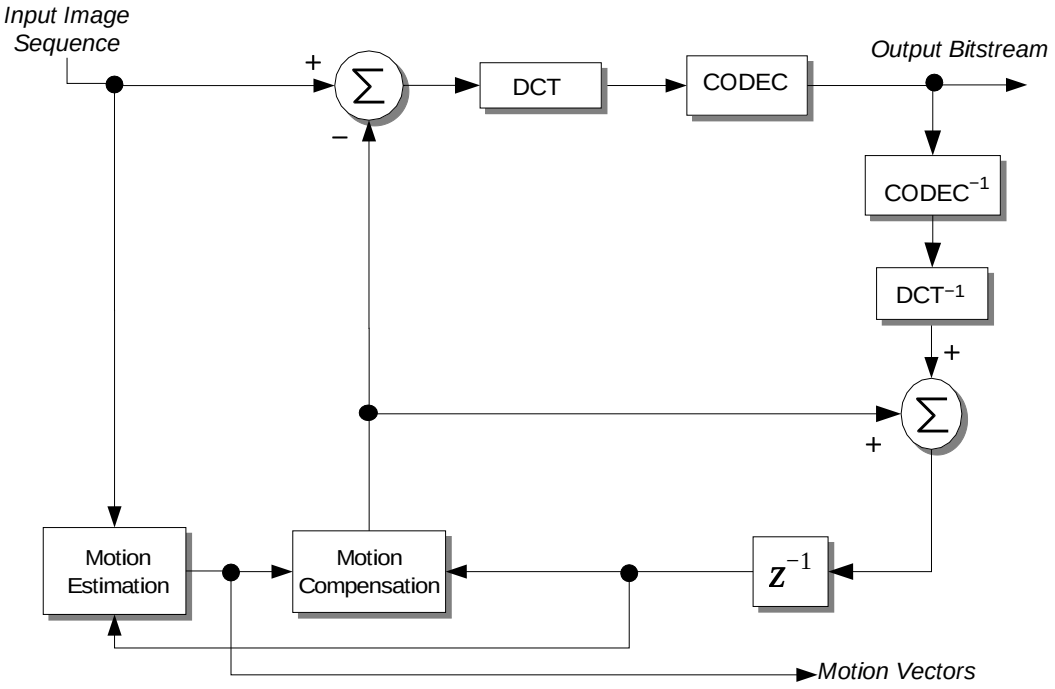
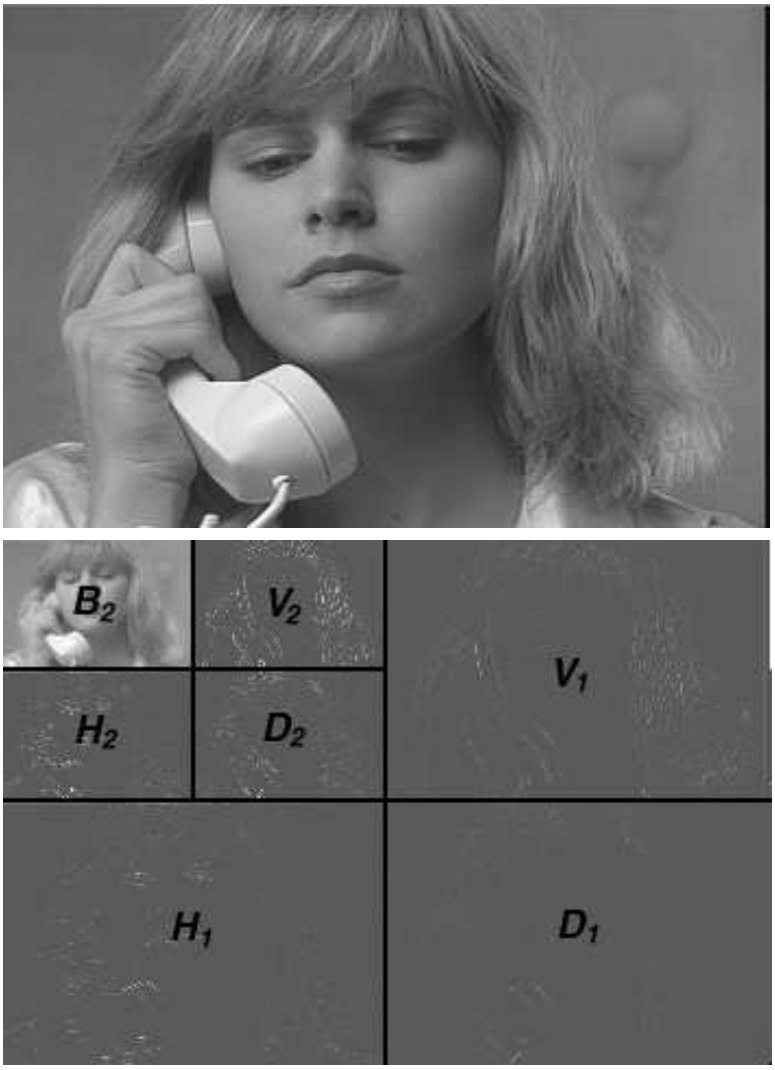


Figure 1.6: The traditional hybrid coder with motion estimation and motion compensation (ME/MC) followed by a discrete cosine transform (DCT). z^{-1} = frame delay, *CODEC* is any still-image coder.



(a)

(b)

Figure 1.7: Original and two-scale DWT decomposition of the first frame of the "Susie" sequence. (a) Original, (b) Two-scale DWT. B_j, H_j, V_j, and D_j denote the baseband, horizontal, vertical, and diagonal subbands, respectively, at scale j.

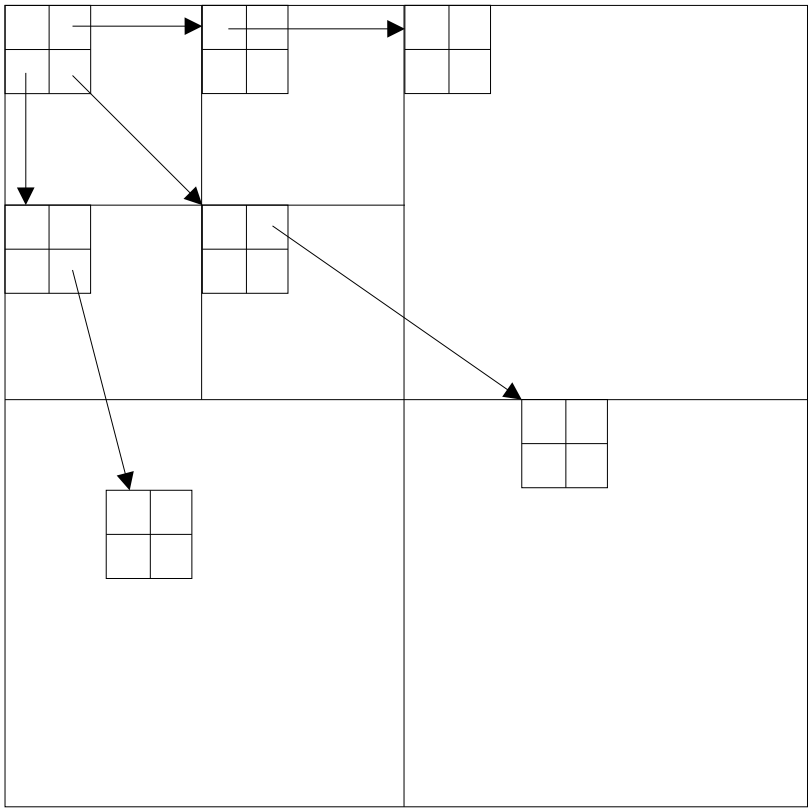


Figure 1.8: Structure of hierarchical trees with the DWT subbands as employed by SPIHT.

known (e.g., [9, 10]) that this simple approach has certain drawbacks due to blocking artifacts which are exacerbated when the DWT is deployed as is usual as a full-frame transform. To reduce these blocking artifacts, it has been proposed [11] to use overlapped block motion compensation (OBMC) in the spatial domain before the DWT.

An alternative paradigm, shown in Fig. 1.10, would be to have ME/MC take place in the wavelet domain. Wavelet-domain ME/MC eliminates the inefficiency due to high-frequency blocking artifacts; more important, perhaps, is that resolution-scalable coding without drift becomes possible. Both direct [9] and hierarchical [12, 13] block-matching of DWT coefficients have been proposed. However, the fact that the usual critically sampled DWT used ubiquitously in image-compression efforts is shift variant greatly hinders the ME/MC process when deployed in wavelet domain.

To demonstrate the difficulty that the shift variance of the DWT poses in the task of tracking motion, consider the example illustrated in Figs. 1.11 and 1.12. Shown in Fig. 1.11 is a signal $s(n)$ and a shifted version of the signal, $s(n-1)$. We perform a 1-scale DWT on both $s(n)$ and $s(n-1)$ and display the resulting coefficients in Fig. 1.12. Here the Cohen-Daubechies-Feauveau 9-7 filter [14] is used. In original signal domain, the effect of the shift is readily apparent, and the "motion" of the signal waveform is easily determined by comparing $s(n-1)$ to $s(n)$. However, in the wavelet domain, the low-band and high-band signals suffer from the shift-variant characteristic of the DWT. We can see that, although there is still some correlation between low-band outputs, the high-band signals are completely dissimilar. In any event, the obtaining of accurate motion vectors for ME will not be possible using either the low-band or high-band signals in the DWT domain.

In order to overcome the shift variance of DWT, a number of proposals [15–27] have been made to use an overcomplete, or redundant, wavelet transform for ME/MC since such a redundant discrete wavelet transform (RDWT) lacks subsampling and is thus shift

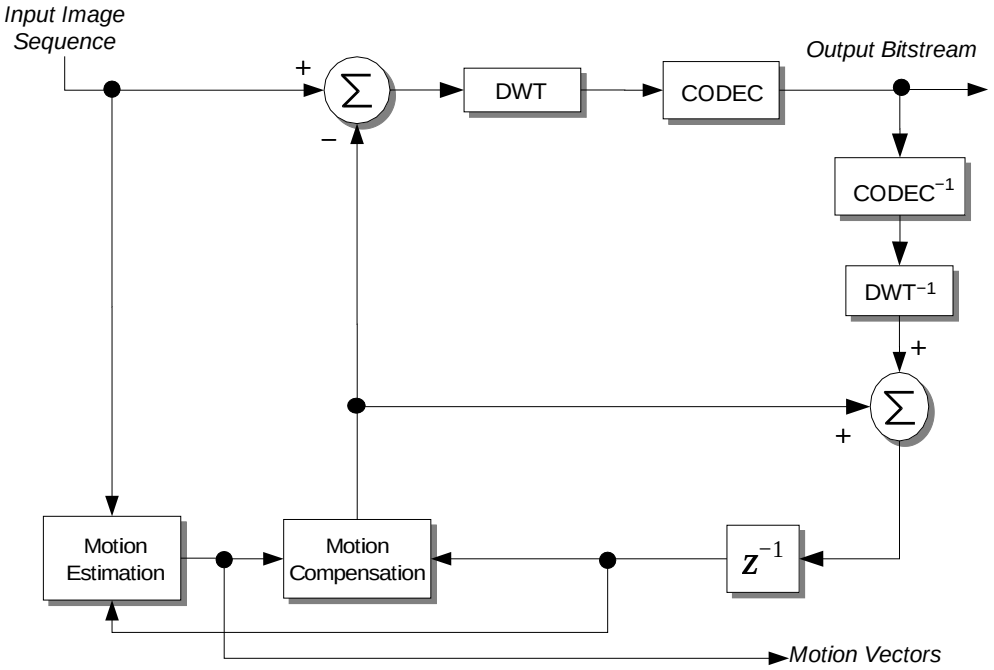


Figure 1.9: The traditional hybrid coder with a DWT replacing the usual DCT. z^{-1} = frame delay, *CODEC* is any still-image coder operating in the critically-sampled-DWT domain.

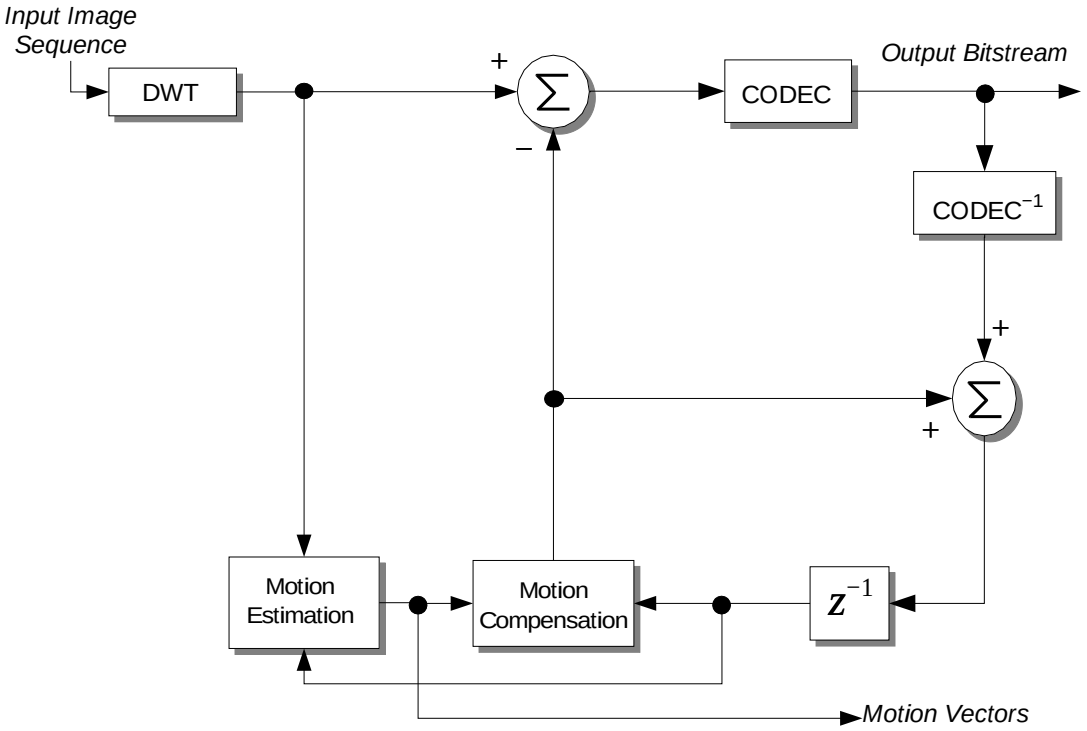


Figure 1.10: Hybrid coder with ME/MC taking place in the DWT domain. z^{-1} = frame delay, *CODEC* is any still-image coder operating in the critically-sampled-DWT domain.

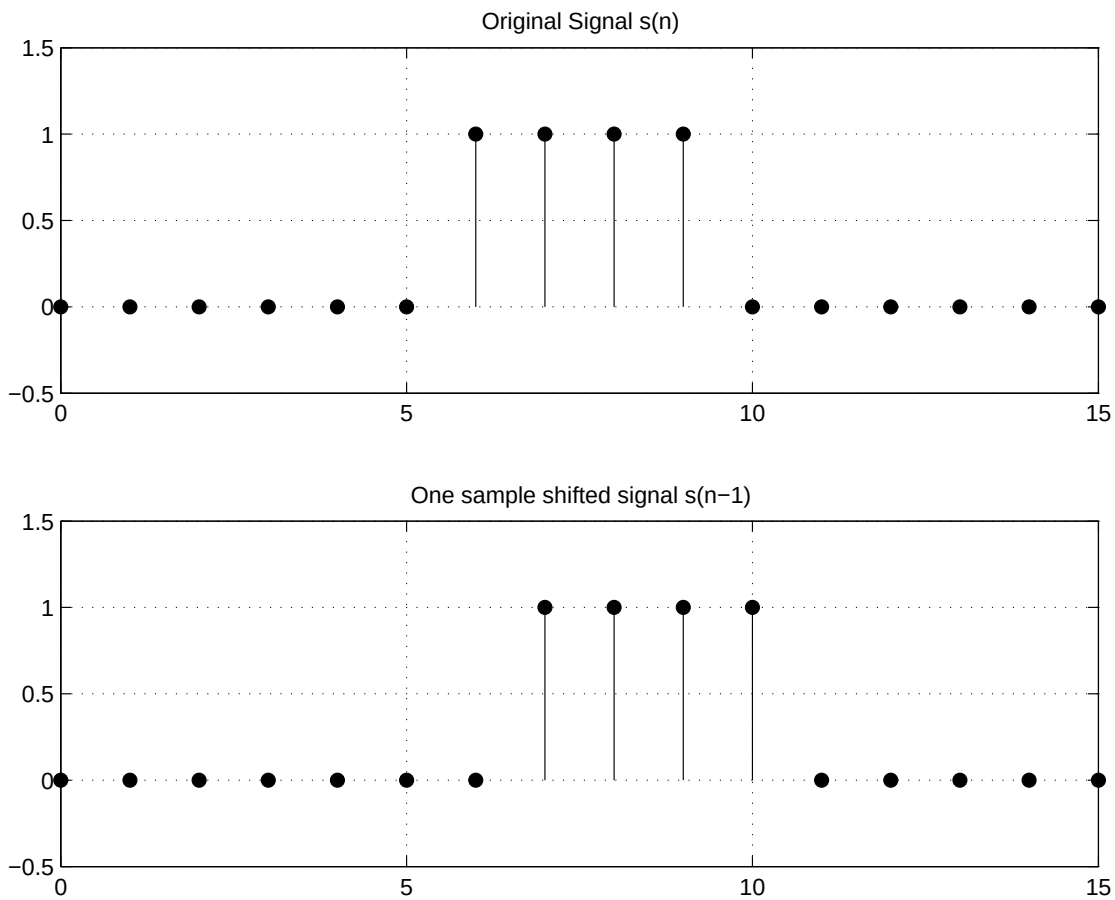


Figure 1.11: Signal $s(n)$ and its shifted version $s(n-1)$.

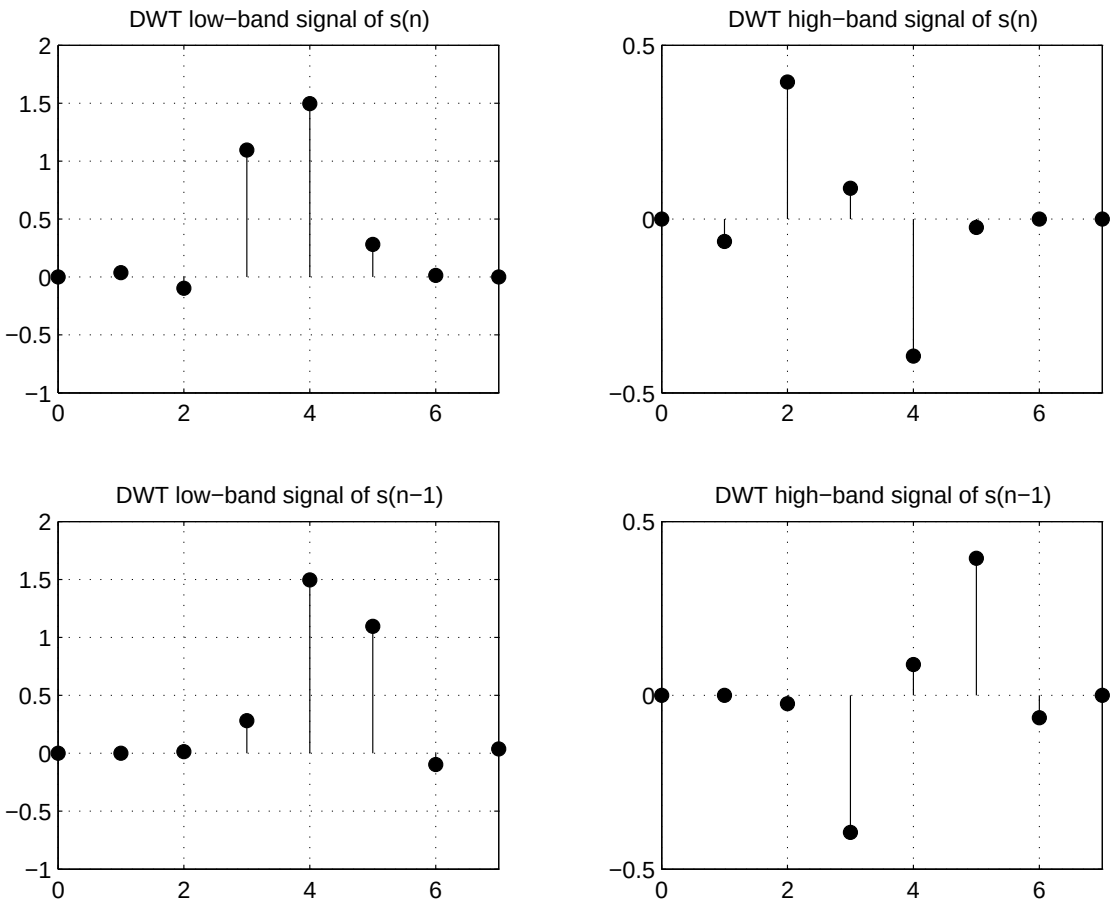


Figure 1.12: Wavelet-domain representations of $s(n)$ and $s(n-1)$.

invariant. This dissertation will consequently explore the use of RDWT in video coding. Park and Kim [15] were the first to incorporate the RDWT into a video coder, using an RDWT-domain reference frame to search for the best match for a block in the DWT of the current frame. A number of other systems [16–24] have been inspired by their coder, but all are essentially built on the same block-based RDWT-domain approach of [15].

As the first contribution of this dissertation, we present the redundant wavelet triangle mesh (RWTM) system which applies a triangle mesh to replace the traditional block-based ME/MC of [15]. This RWTM system, first developed in [25, 26], yields performance superior to that of the block-based system of [15].

As the second contribution of this dissertation, we investigate the combination of RDWT-based ME/MC with multihypothesis MC (MHMC). MHMC, which calls for using several hypothesis predictions of motion, has long been used in video-coding systems to compensate for the inherent inaccuracy of the ME process. In this dissertation, we develop a new class of MHMC, redundant wavelet multihypothesis (RWMH) [28, 29], which exploits redundancy in the RDWT domain to improve motion prediction. Additional investigation is focused on further exploring the performance of RWMH. Initially, we consider the combination of RWMH with other, more traditional forms of multihypothesis. We then explore the use of triangle meshes within RWMH, essentially combining the RWTM system of the first part of the dissertation with the RWMH system of the latter part.

The remainder of this dissertation is organized as follows to describe our work in detail. Chap. II presents theoretical background on the RDWT. Next, prior uses of the RDWT in video coding are overviewed in Chap. III. The RWTM and RWMH systems are then introduced in Chaps. IV and V, respectively, followed by a presentation of experimental results and observations in Chap. VI. Finally, we make some concluding remarks in Chap. VII.

CHAPTER II

THE REDUNDANT DISCRETE WAVELET TRANSFORM (RDWT)

In this chapter, we review the redundant discrete wavelet transform (RDWT). We first overview some theoretical aspects of the transform by comparing it to the ubiquitous DWT in Sec. 2.1 and then review several practical alternatives for RDWT implementation and coefficient representation in Sec. 2.2. We then discuss inversion of the RDWT in Sec. 2.3, and then, finally, we consider the ramifications of the RDWT for motion estimation (ME) by illustrating its shift invariance in Sec. 2.4. The RDWT has a long history of development within the signal-processing community. For greater elaboration on the discussion here, consult [30–34].

2.1 RDWT vs. DWT

The RDWT can be considered to be an approximation to the continuous wavelet transform that removes the downsampling operation from the traditional critically sampled DWT to produce an overcomplete representation. The shift-variance characteristic of the DWT arises from its use of downsampling, while the RDWT is shift invariant since the spatial sampling rate is fixed across scale. The RDWT has been given several appellations over the years, including the "undecimated DWT," the "overcomplete DWT," and the *algorithme à trous*.

To describe the implementation of the RDWT in terms of filter-banks, let us first illustrate the same for the DWT. A 1D DWT and its inverse are illustrated in Fig. 2.1. Here, $f[n]$ is the 1D input signal and $f'[n]$ is the reconstructed signal. $h[-k]$ and $g[-k]$ are the lowpass and highpass analysis filters, while the corresponding lowpass

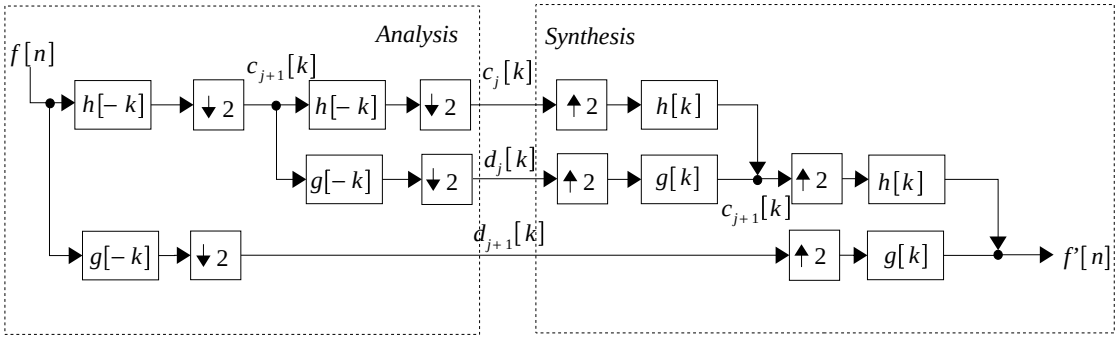


Figure 2.1: Two level 1-D DWT analysis and synthesis filter banks.

and highpass synthesis filters are $h[k]$ and $g[k]$. c_j and d_j are the low-band and high-band output coefficients at level j. DWT analysis, or decomposition, is, mathematically,

$$c_j[k] = (c_{j+1}[k] * h[-k]) \downarrow 2, \tag{2.1}$$

and

$$d_j[k] = (c_{j+1}[k] * g[-k]) \downarrow 2, \tag{2.2}$$

where $*$ denotes convolution, and $\downarrow 2$ denotes downsampling by a factor of two. That is, if $y[n] = x[n] \downarrow 2$, then

$$y[n] = x[2n]. \tag{2.3}$$

The corresponding operation of DWT synthesis, or reconstruction, is

$$c_{j+1}[k] = (c_j[k] \uparrow 2) * h[k] + (d_j[k] \uparrow 2) * g[k], \tag{2.4}$$

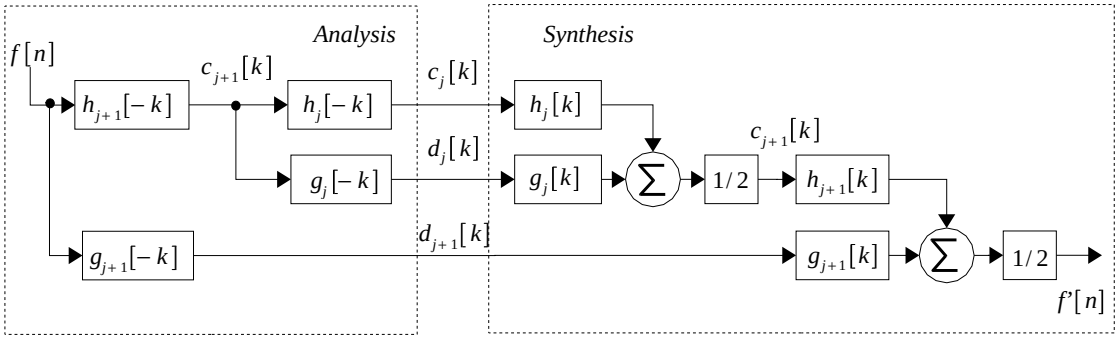


Figure 2.2: Two level 1-D RDWT analysis and synthesis filter banks.

where $\uparrow 2$ denotes upsampling by a factor of two. That is, if $y[n] = x[n] \uparrow 2$, then

$$y[n] = \begin{cases} x[n/2], & n \text{ even}, \\ 0, & n \text{ odd}. \end{cases} \tag{2.5}$$

In contrast, a 1D RDWT and its inverse are illustrated in Fig. 2.2. The RDWT eliminates downsampling and upsampling of coefficients, and at each scale, the number of output coefficients doubles that of the input. The filters themselves are upsampled to fit the growing data length. Specifically, the filters for scale j are

$$h_j[k] = h_{j+1}[k] \uparrow 2, \tag{2.6}$$

and

$$g_j[k] = g_{j+1}[k] \uparrow 2. \tag{2.7}$$

RDWT analysis is then

$$c_j[k] = (c_{j+1}[k] * h_j[-k]), \tag{2.8}$$

and

$$d_j[k] = (c_{j+1}[k] * g_j[-k]), \tag{2.9}$$

while RDWT synthesis is

$$c_{j+1}[k] = \frac{1}{2}(c_j[k] * h_j[k] + d_j[k] * g_j[k]). \tag{2.10}$$

(2.6) through (2.10) are known as the *algorithme à trous* [30], since the filter-upsampling procedure inserts "holes" ("trous" in French) between the filter taps.

2.2 RDWT Implementation and Coefficient Representation

There are several ways to implement the RDWT, and several ways to represent the resulting overcomplete set of coefficients. The most obvious implementation, direct implementation of the *algorithme à trous* as given by (2.6) through (2.9), results in subbands that are exactly the same size as the original signal, as is illustrated for a 1D signal in Fig. 2.3. The advantage of this "spatially coherent" representation is that each RDWT coefficient is located within its subband in its spatially correct position. By appropriately subsampling each subband of an RDWT, one can produce exactly the same coefficients as does a critically sampled DWT applied to the same input signal. In fact, in a J-scale 1D RDWT, there exist 2^J distinct critically sampled DWTs corresponding to the choice between even- and odd-phase subsampling at each scale of decomposition.

As we will see in Chap. III, the most popular coefficient-representation scheme employed in RDWT-based video coders is that of a "coefficient tree," as illustrated in Fig. 2.4 for a 1D signal. This tree representation is easily created by employing filtering and downsampling as in the usual critically sampled DWT; however, all "phases" of downsampled coefficients are retained and arranged as "children" of the signal that was decomposed. The process is repeated on the lowpass bands of all nodes to achieve

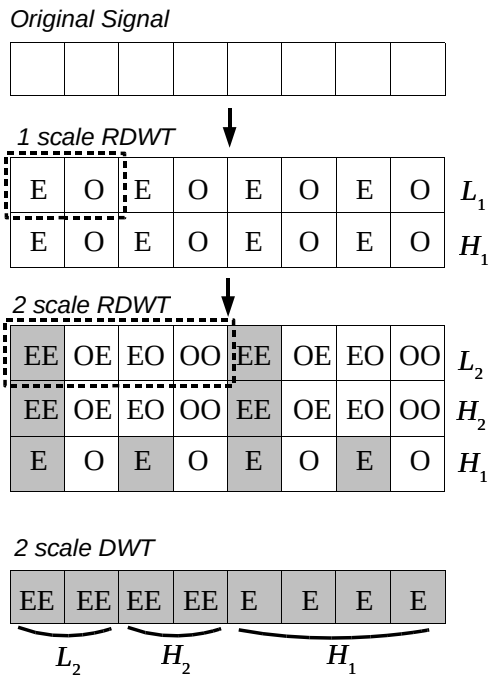


Figure 2.3: Spatially coherent representation of a two-scale 1D RDWT. Coefficients retain their correct spatial location within each subband. Gray coefficients indicate the subsampling pattern necessary to recover one of the 2^J critically sampled DWTs.

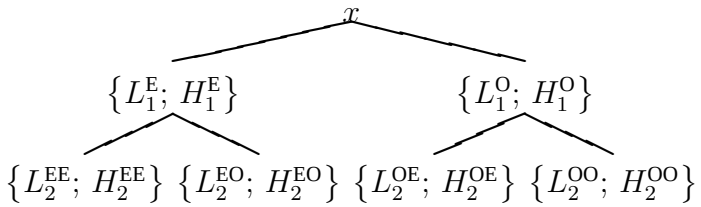


Figure 2.4: Tree representation of a two-scale RDWT of 1D-signal x. Approximation and detail coefficients at scale j are L_j and H_j, respectively. E indicates even-phase subsampling; O indicates odd-phase subsampling. A path from root to leaf indicates a distinct critically sampled DWT; a J-scale RDWT consists of 2^J such DWTs.

multiple decomposition scales. It is straightforward to see that each path from root to leaf in the RDWT tree constitutes a distinct critically sampled DWT, and there are 2^J such critically sampled DWTs in a J-scale decomposition. An alternative, and equivalent, implementation of the RDWT tree representation comes from employing consistent subsampling phase and shifting the lowpass bands by one sample to generate children in the tree. Indeed, this "low-band-shift" [15] method has been a popular implementation for the RDWT-based video coders. It can be shown that the coefficients at a given scale in the tree representation of the RDWT (Fig. 2.4) can be appropriately "interleaved" to produce the subbands of the spatially coherent representation (Fig. 2.3); i.e., the two representations consist of exactly the same coefficient values.

The situation is similar for 2D decompositions implemented with separable 1D transforms, as illustrated in Fig. 2.5. A J-scale 2D RDWT consists of 4^J distinct critically sampled DWTs. An example of RDWT image is shown in Fig. 2.6

2.3 The Inverse RDWT

The RDWT is a perfectly reconstructing transform. To invert the RDWT, one can simply independently invert each of the constituent critically sampled DWTs and average the resulting reconstructions together. However, this implementation of the inverse RDWT incurs unnecessary duplicate synthesis filterings of the highpass bands; thus, one usually alternates between synthesis filtering and reconstruction averaging on a scale-by-scale basis in practical implementations as illustrated in Fig. 2.2. The final reconstruction of this latter implementation, however, is identical to that produced by the conceptually simpler former approach.

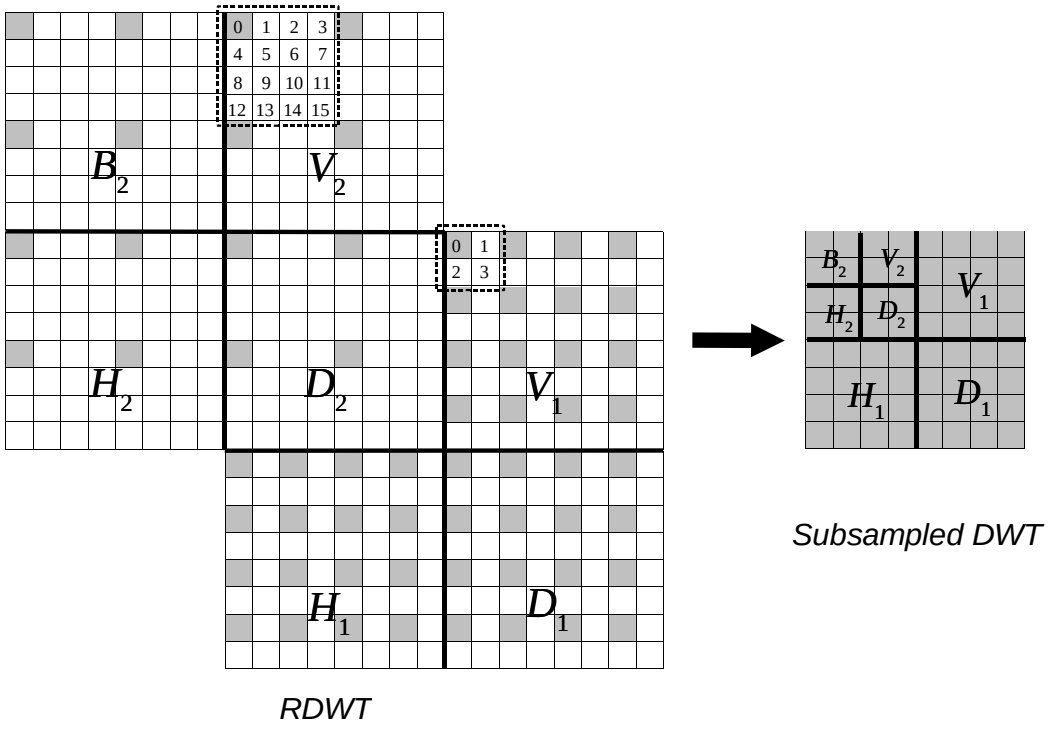


Figure 2.5: Spatially coherent representation of a two-scale 2D RDWT. Coefficients retain their correct spatial location within each subband, and each subband is the same size as the original image. B_j, H_j, V_j, and D_j denote the baseband, horizontal, vertical, and diagonal subbands, respectively, at scale j. This figure shows subsampling recovering one of the 4^J critically sampled DWTs.

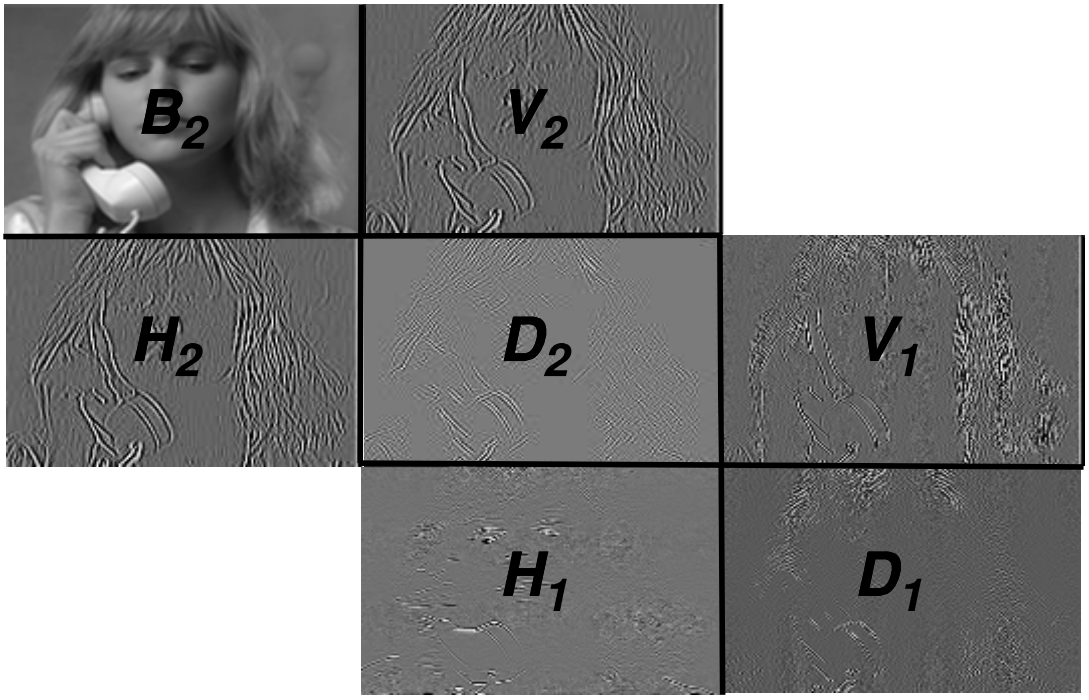


Figure 2.6: An example of a two-scale 2D RDWT applied to the first frame of "Susie" sequence. B_j, H_j, V_j, and D_j denote the baseband, horizontal, vertical, and diagonal subbands, respectively, at scale j.

2.4 Shift Invariance of the RDWT

To demonstrate that the lack of downsampling in the RDWT renders it shift invariant, let us revisit the example of Figs. 1.11 and 1.12 in Chap. I. The RDWT outputs of both the signals $s(n)$ and $s(n-1)$ of Fig. 1.11 are illustrated in Fig. 2.7. Compared to Fig. 1.12 in Chap. I, in which it was impossible to determine the amount of motion in the DWT domain, the RDWT subbands of Fig. 2.7 correctly reflect the one-sample motion. That is, the subbands of the RDWT of $s(n-1)$ are shifted versions of the subbands of $s(n)$, just as $s(n-1)$ is a shifted version of $s(n)$, and the amount of shift in each domain is identical.

The shift invariance of the RDWT implies that ME/MC with an RDWT subband can be performed essentially in the same manner as in the original spatial-domain frame. This observation has spawned a number of RDWT-based video-coding systems. In the next chapter, we survey a number of such systems.

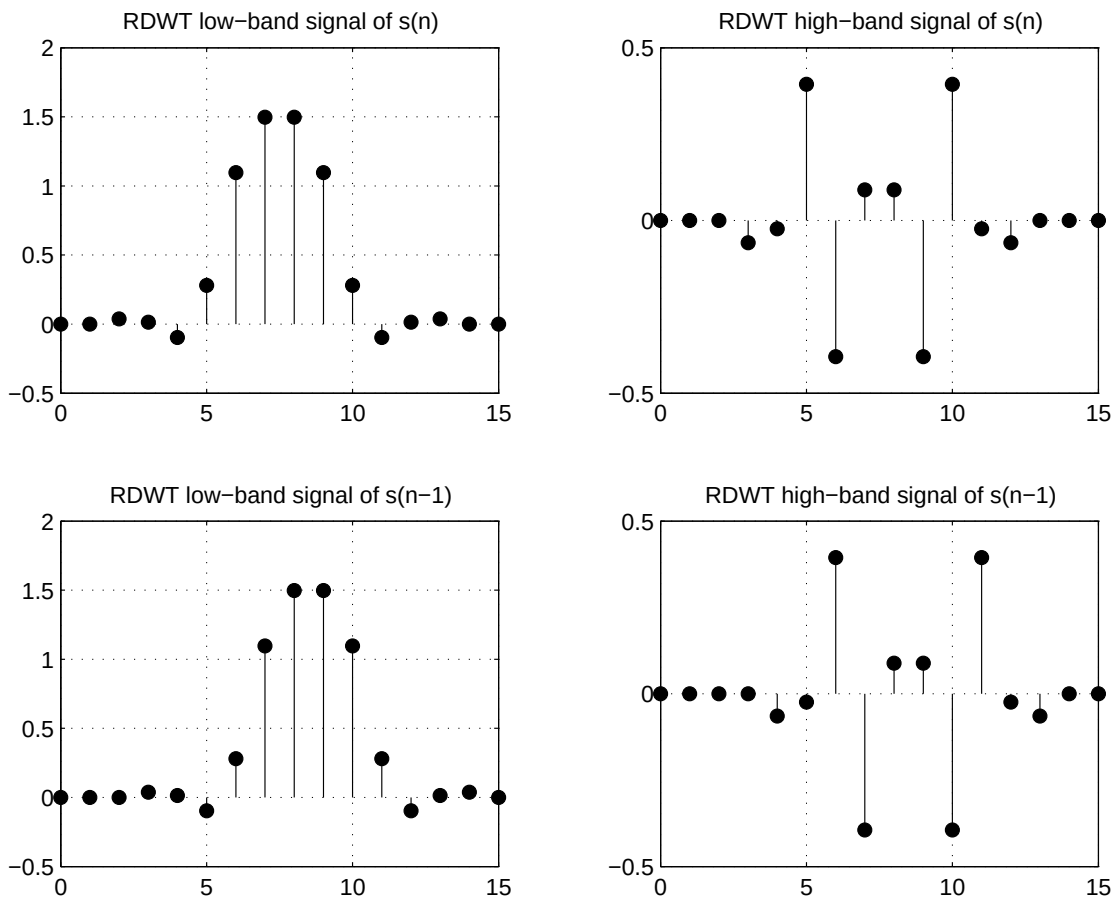


Figure 2.7: RDWT-domain representations of $s(n)$ and $s(n-1)$.

CHAPTER III

PRIOR USE OF THE RDWT IN VIDEO CODING

Previously, we have seen that the redundancy within the RDWT provides shift invariance. In this chapter, we will explore a number of video-coding systems that have been proposed to capitalize upon this shift invariance to implement ME/MC in the wavelet domain. As all these systems have their origins in system of [15], we first review the architecture and performance of this system in Secs. 3.1 and 3.2, respectively. We then consider in Sec. 3.3 a number of refinements that have been proposed to the basic system.

3.1 Overview of the RDWT-Block System

The majority of prior work concerning RDWT-based video coding originates in the work of Park and Kim [15], in which the system shown in Fig. 3.1 was proposed. In this system, the RDWT is implemented with the "low-band shift" procedure and the ME/MC is performed with blocks. Hence, we call this technique "RDWT Block".

In essence, the system of Fig. 3.1 works as follows. An input frame is decomposed with a critically sampled DWT, and the resulting wavelet-domain coefficients are partitioned into blocks. Each block consists of all the coefficients in the DWT that correspond to a particular spatial-domain block in the original image, and thus includes coefficients from all subbands at all scales. A full-search block-matching algorithm then computes motion vectors for each wavelet-domain block; the system uses as the reference for this search an RDWT decomposition of the previous reconstructed frame. Since these reconstructed RDWT coefficients are arranged in the tree representation

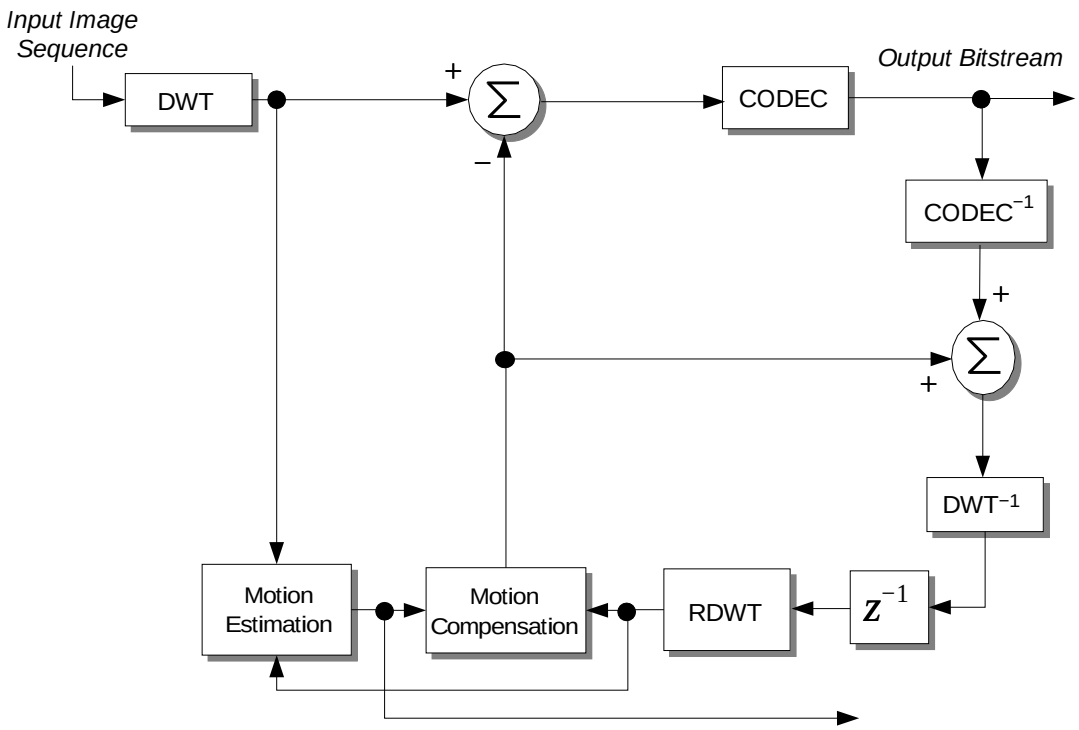


Figure 3.1: The RDWT-based video coder of [15]. z^{-1} = frame delay, *CODEC* is any still-image coder operating in the critically-sampled-DWT domain.

as described in Sec. 2.2, the ME procedure of this system amounts to identifying, for each block of the current frame, a particular critically sampled DWT in the reference-frame tree (a root-to-leaf path), and a displacement within that DWT. Transmission of a single motion vector per block suffices to convey all of this motion information to the decoder. A suitable cross-scale distortion metric that averages distortions incurred in each subband is used to drive the ME search.

Specifically, a $B \times B$ block of DWT coefficients is extracted from the critically sampled DWT of the current frame as illustrated in Fig. 3.2. As shown, this block consists of all the DWT coefficients in the various subbands that correspond to the given spatial location of the block. In the block-matching search of the RDWT-block system, this DWT block is compared to $B \times B$ blocks extracted from the RDWT of the reference frame, as illustrated in Fig. 3.2. In the RDWT of the reference frame, the coefficients are arranged in the tree representation that results from the low-band-shift procedure described in Sec. 2.2. Since the tree representation of the RDWT consists of multiple critically sampled DWTs, the block-matching procedure of the RDWT-block system compares the current-frame DWT block to reference-frame blocks extracted from each critically sampled DWT of the RDWT of the reference frame as illustrated in Fig. 3.2.

Specifically, a block of $B \times B$ coefficients is extracted from the DWT of the current frame and compared to blocks of $B \times B$ coefficients extracted from the RDWT of the reference frame as illustrated in Fig. 3.2. Mathematically, the distortion metric for the ME search is as follows. Let S_j^{cur} be subband S at scale j of the DWT of the current frame, and S_j^{ref} be subband S at scale j of the RDWT of the reference frame, where $1 \leqslant j \leqslant J$, and S is B, H, V, or D, for the baseband, horizontal, vertical, or diagonal subbands, respectively. Let (x, y) be the location of a block in the original

image coordinates. The corresponding motion vector is

$$(\Delta_x, \Delta_y) = \arg \min_{-W \leqslant \Delta_x, \Delta_y \leqslant W} \text{MAE}(x, y, \Delta_x, \Delta_y) \tag{3.1}$$

where the mean absolute error (MAE) is

$$\begin{aligned}
&\text{MAE}(x, y, \Delta_x, \Delta_y) = \\
&\quad \frac{2^{-J}}{B^2} \sum_{k=1}^{B/2^J} \sum_{l=1}^{B/2^J} \left| B_J^{cur}(x/2^J + k, y/2^J + l) - B_J^{ref}(x + 2^J k + \Delta_x, y + 2^J l + \Delta_y) \right| \\
&\quad + \frac{1}{B^2} \sum_{j=1}^{J} 2^{-j} \sum_{k=1}^{B/2^j} \sum_{l=1}^{B/2^j} \Bigg[\Big| V_j^{cur}(x/2^j + k, y/2^j + l) \\
&\quad - V_j^{ref}(x + 2^j k + \Delta_x, y + 2^j l + \Delta_y) \Big| \\
&\quad + \left| H_j^{cur}(x/2^j + k, y/2^j + l) - H_j^{ref}(x + 2^j k + \Delta_x, y + 2^j l + \Delta_y) \right| \\
&\quad + \left| D_j^{cur}(x/2^j + k, y/2^j + l) - D_j^{ref}(x + 2^j k + \Delta_x, y + 2^j l + \Delta_y) \right| \Bigg],
\end{aligned} \tag{3.2}$$

and $W > 0$ is the search-window size. In (3.2), k and l indicate the different subsampling phases in RDWT tree-structure representation. In summary, a single critically sampled DWT of the current frame is predicted in a block-by-block manner from a wavelet-domain reference frame wherein all phases are retained. By using such an overcomplete expansion of the reference frame, the best-matching block from all possible phases is obtained, and the shift-variant nature of the critically sampled DWT is overcome.

We note that, although the original development [15] of the RDWT-block system used the tree representation of the RDWT, it is possible to use the spatially coherent representation as well. That is, as discussed in Sec. 2.2, it is possible to interleave the coefficients from the tree representation of the RDWT to produce the spatially coherent

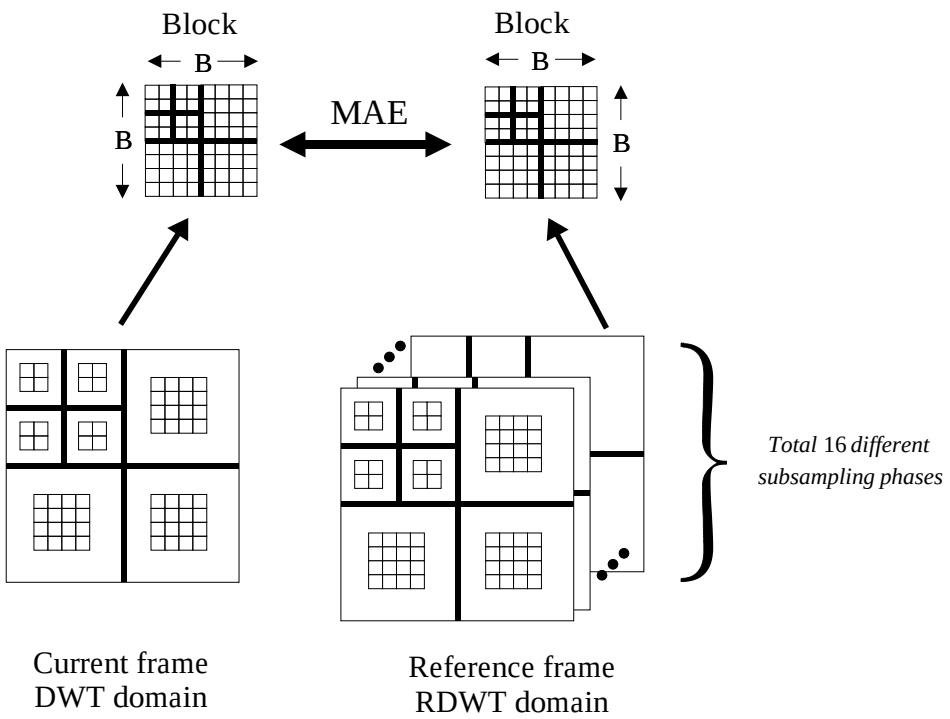


Figure 3.2: The motion estimation procedure of [15], tree representation, where $B \times B$ coefficients are extracted out to build a block in DWT as well as RDWT domain. The MAE is calculated between these blocks.

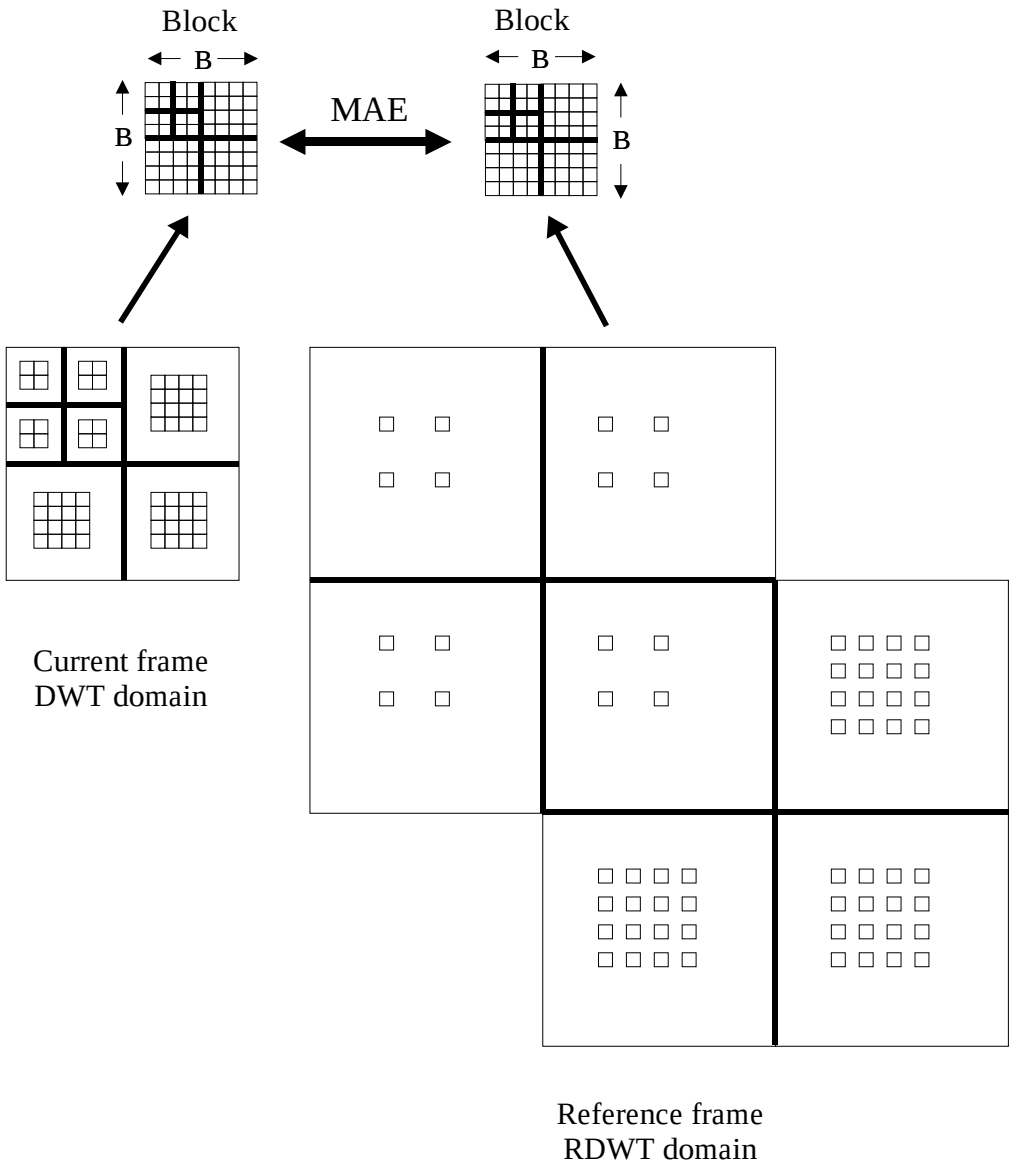


Figure 3.3: The motion estimation procedure of [15], spatially coherent representation, where $B \times B$ coefficients are extracted out to build a block in DWT as well as RDWT domain. The MAE is calculated between these blocks.

representation. In this case, the block-matching search of the RDWT-block system then becomes as illustrated in Fig. 3.3. Although equivalent algorithmically to the search of Fig. 3.2, this alternative implementation has certain conceptual advantages that will facilitate the introduction of RDWT-based coders that we will develop in subsequent chapters.

3.2 Performance of the of RDWT-Block System

In these experiments, we compare the RDWT block with direct wavelet-domain block-based ME/MC (DWT Block). In the DWT-Block system, both the current and reference frames are in the critically subsampled DWT domain. Consequently the ME/MC in this system suffers from shift-variance problem. We use the 100-frame "Football" SIF sequence, the 70-frame "Susie" SIF sequence, the 300-frame "Mother & Daughter" CIF sequence, and the 300-frame "Coastguard" CIF sequence. All sequences are grayscale. The first frame is intra-encoded (I-frame) while all subsequent frames use ME/MC (P-frames). Both wavelet transforms (DWT and RDWT) use the Cohen-Daubechies-Feauveau 9-7 filter [14] with symmetric extension and a decomposition of $J = 3$ levels. Both ME/MC methods use integer-pixel accuracy and approximately the same number of motion vectors per frame.

The average PSNRs are shown in Table 3.1 and indicate at least 3-dB gain over all sequences. Thus, driving ME/MC in the RDWT domain instead of critically sampled DWT domain yields significantly better motion prediction. Frame-by-frame PSNR profiles for the "Football" and "Susie" sequences are shown in Figs. 3.4 and 3.5. Fig. 3.6 gives the reconstructed images of frame 6 of "Football", where we can easily see that the RDWT-block system significantly outperforms the DWT-block system.

Table 3.1: Distortion averaged over all frames of the sequence.

	PSNR (dB)			
	Football†	Susie	Mother & Daughter	Coastguard
DWT Block	24.4	33.5	33.4	24.0
RDWT Block	27.9	37.4	40.8	28.9

Rate is 0.25 bpp except †, which is 0.5 bpp.

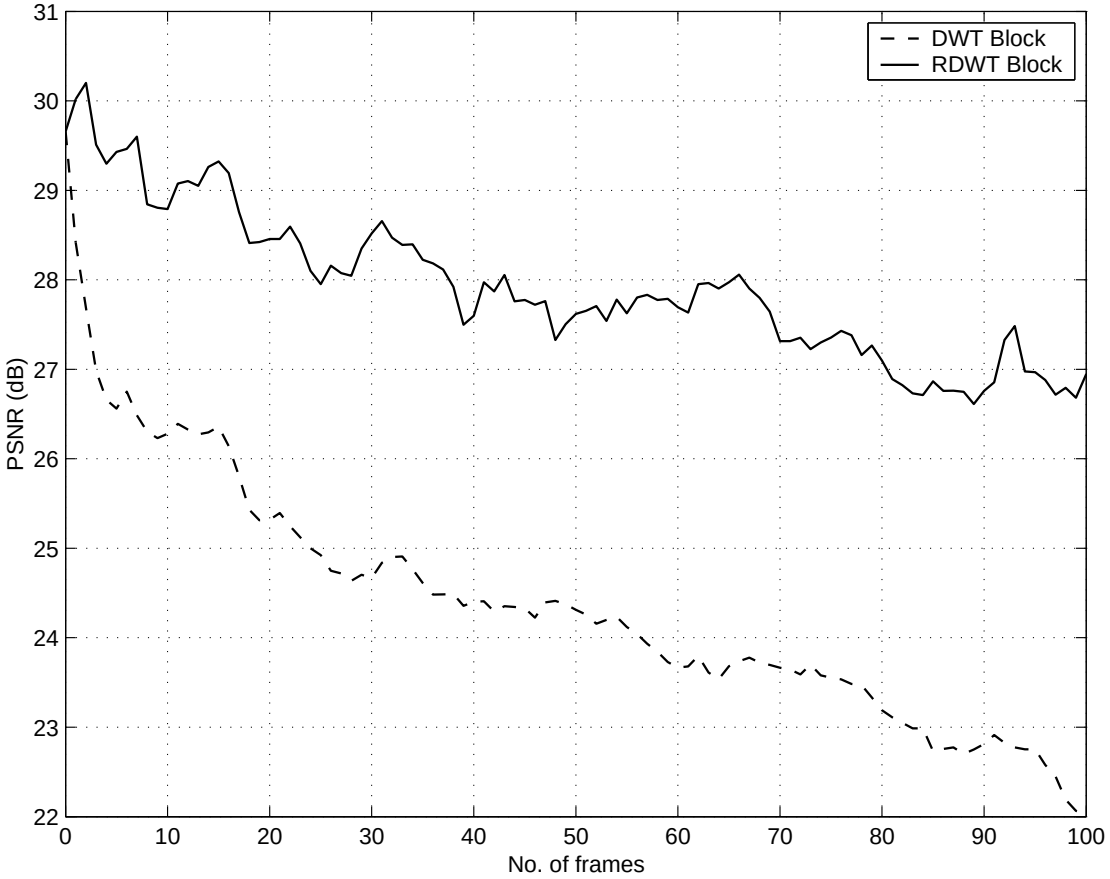


Figure 3.4: Comparison of DWT Block to RDWT Block—frame-by-frame PSNR for "Football" at 0.5 bpp (1.3 Mbps).

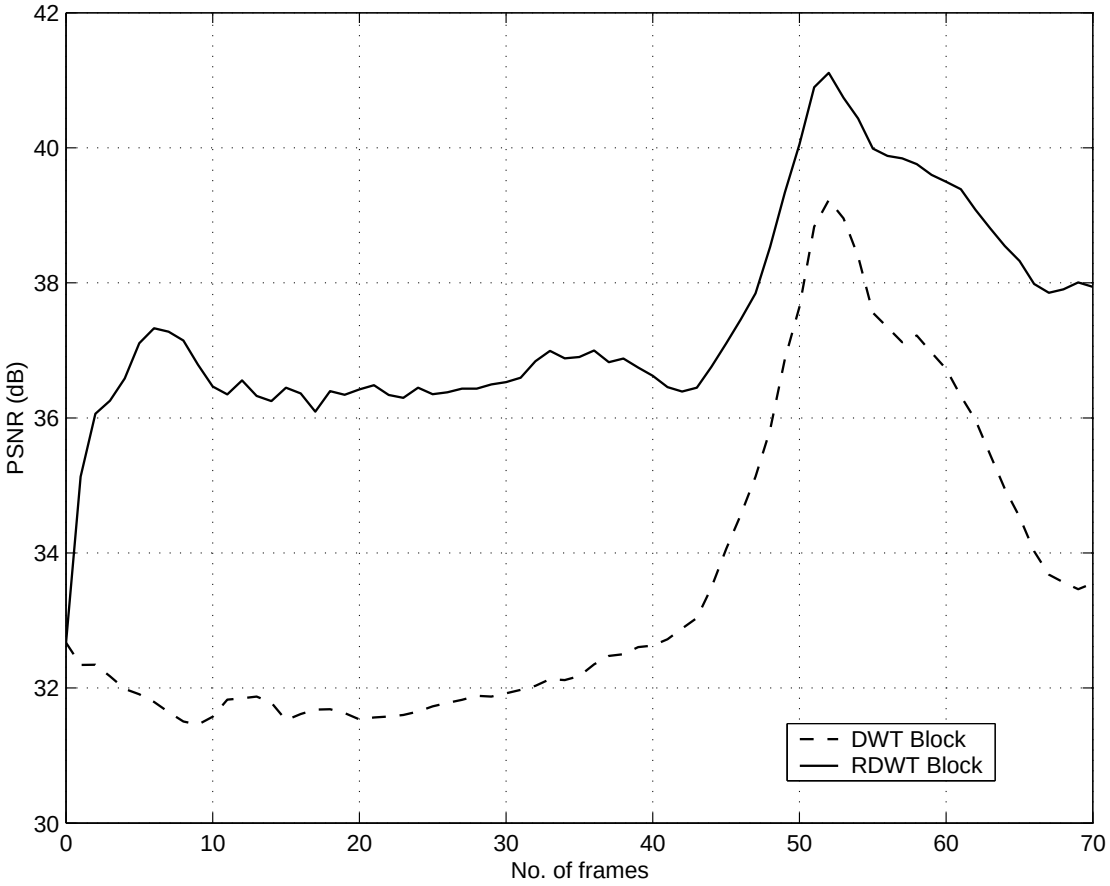


Figure 3.5: Comparison of DWT Block to RDWT Block—frame-by-frame PSNR for "Susie" at 0.25 bpp (634 kbps).

(a)

(b)

(c)

Figure 3.6: Original and reconstructed images for frame 6 of “Football”. (a) Original, (b) DWT Block, (c) RDWT Block.

3.3 Other RDWT Video-coding Systems

Subsequent work has further refined the system depicted in Fig. 3.1. In particular, in [16, 23, 24], multiple motion vectors are transmitted for each current-frame block by estimating motion in each subband independently. By doing so, a fast algorithm for calculating the level-by-level RDWT coefficients is achieved. The system of [17] employs interpolation between the coefficients in distinct root-to-leaf paths of the RDWT tree to enable motion compensation to be performed with sub-pixel accuracy. Additionally, resolution-scalable video coders [18, 21, 23, 24] have been devised that constrain the ME/MC procedure to process each scale of the wavelet decomposition independently. Each of these systems built upon the architecture of Fig. 3.1 retains its block-based ME/MC procedure and its system structure. That is, the current frame is decomposed into DWT coefficients, the reference frame is decomposed into RDWT coefficients, and in the ME procedure, the current DWT block is matched to an overcomplete RDWT reference block. Next, we will look at two important refinements proposed to the RDWT-block system: in-band prediction and half-pixel accuracy.

3.3.1 In-band Prediction

In the RDWT domain, there are a total of $3 \times J + 1$ subbands for a J-level decomposition. In the RDWT-block system described above, one set of motion vectors describes motion in all subbands simultaneously. In order to support resolution, quality, and frame-rate scalability, ME/MC can be performed level-by-level [16, 21, 23, 24]. In this case, although the current and the reference frames are decomposed into J levels of wavelet decomposition, ME is first employed on the highest level, level J, consisting of four subbands, B_J, H_J, V_J, and D_J. Block-based ME then finds the motion vectors at level J, and the motion vectors along with the residual image at level J are transmitted. If the target bitrate is larger than the bit rate used to code the level-J motion vectors and

residual image, ME is carried out in level $J-1$, and so on. Coincident with completion of the encoding at the encoder side and decoding at the decoder side at each level, the reference image is refined. Thus, the reference image is updated upon receiving the motion vectors level-by-level. This in-band prediction introduces resolution scalability into RDWT ME/MC, at the cost of the increased overhead for motion vectors.

3.3.2 Half-pixel Accuracy

Another refinement to the RDWT-block system is to extend the integer-pixel accuracy used in [15] to half-pixel accuracy [17]. In this approach, the RDWT reference frame is bilinearly interpolated to obtain a new reference frame in sub-pixel accuracy. This half-pixel interpolation is illustrated in (3.3) – (3.5) and Fig. 3.7, where A, B, C and D indicate the integer pixels, while a, b and c are the interpolated half pixels. a, b and c are obtained by bilinear interpolation from A, B, C and D as

$$a = (A + B)/2, \tag{3.3}$$

$$b = (A + C)/2, \tag{3.4}$$

$$c = (A + B + C + D)/4. \tag{3.5}$$

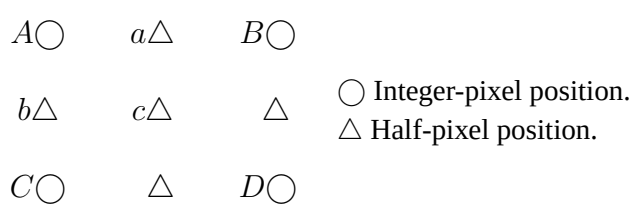


Figure 3.7: Half-pixel accuracy obtained by interpolation.

Table 3.2: Distortion averaged over all frames of the sequence.

	PSNR (dB)			
	Football†	Susie	Mother & Daughter	Coastguard
RDWT Block Integer Accuracy	27.9	37.4	40.8	28.9
RDWT Block Half Accuracy	29.1	38.1	39.4	30.1

Rate is 0.25 bpp except †, which is 0.5 bpp.

The RDWT-block system is then modified so that the search as illustrated in Fig. 3.3 is carried out with half-pixel accuracy in the interpolated RDWT reference frame. This incurs the addition of one bit of precision to each component of the motion vectors. The average PSNRs are shown in Table 3.2 and frame-by-frame PSNR profiles for the "Football" and "Susie" sequences are shown in Figs. 3.8 and 3.9. We see that the performance is improved significantly for the "Football", "Susie", and "Coastguard" sequences when half-pixel accuracy is used.

In this chapter, we have reviewed a number of video-coding systems that employ the RDWT to provide shift invariance, thus enabling ME/MC to take place in the wavelet domain. However, as we will see in the following chapters, the redundancy inherent in the RDWT can be employed for ends other than just shift invariance. Specifically, in the next chapter we will introduce a system that exploits the redundancy of the RDWT to enable ME/MC with geometry more general than that of the blocks used in the systems we have thus far considered.

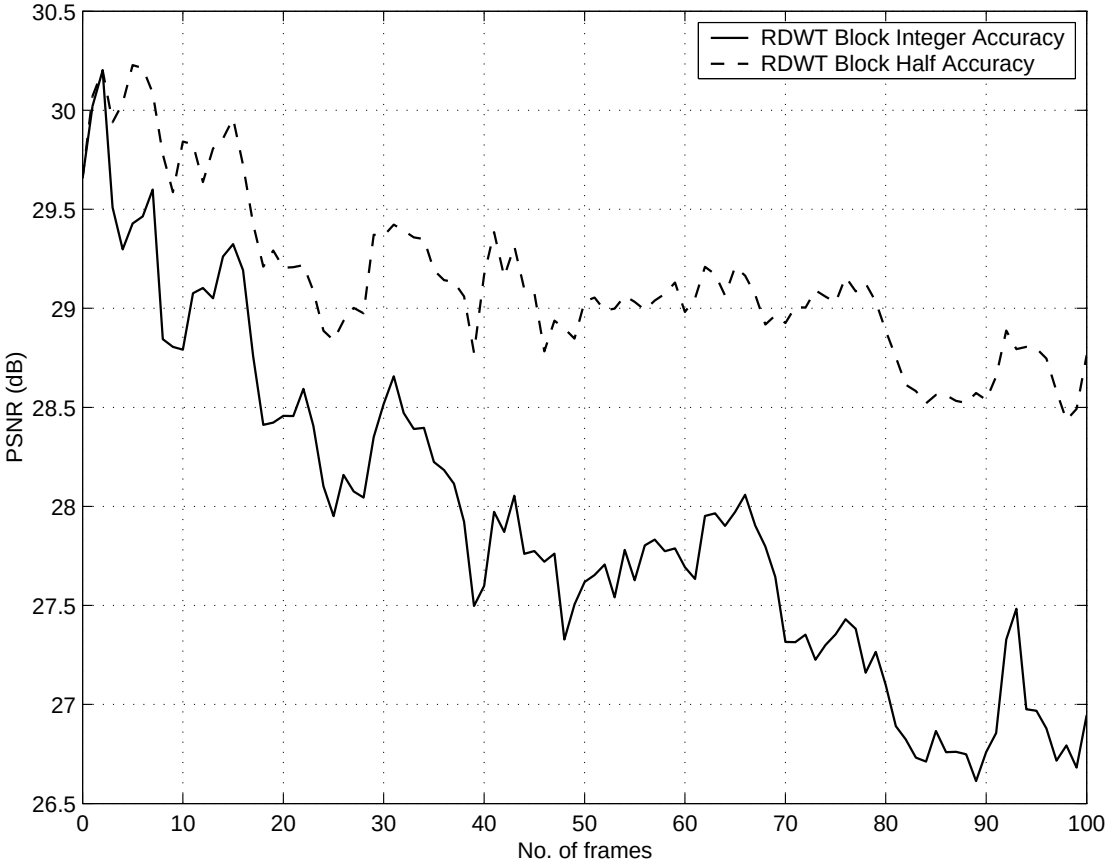


Figure 3.8: Comparison of RDWT Block integer to half accuracy—frame-by-frame PSNR for “Football” at 0.5 bpp (1.3 Mbps).

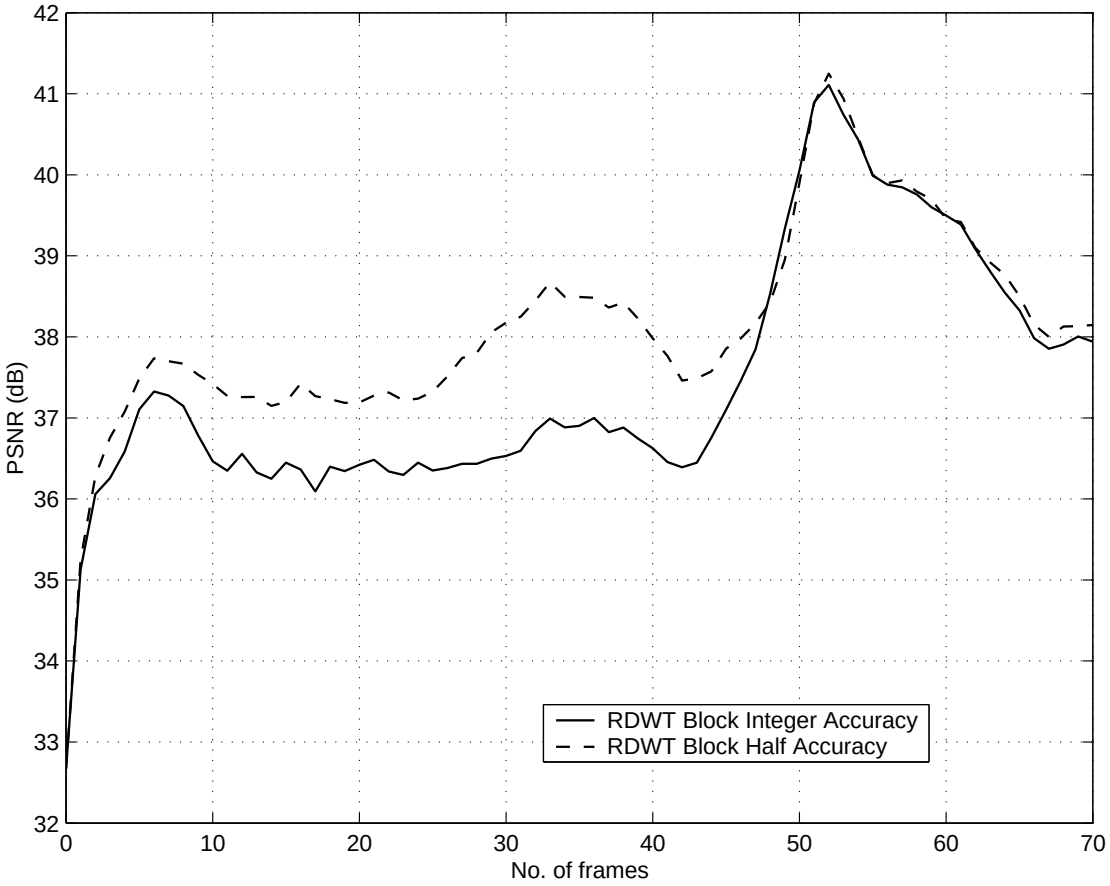


Figure 3.9: Comparison of RDWT Block integer to half accuracy—frame-by-frame PSNR for “Susie” at 0.25 bpp (634 kbps).

CHAPTER IV

REDUNDANT WAVELET TRIANGLE MESH (RWTM)

As has been illustrated in the previous chapter, the RDWT is shift-invariant; consequently the RDWT domain is much more amemable to ME/MC than the critically sampled DWT domain. The system by Park and Kim [15] and other related systems [16–18, 21, 23, 24] demonstrate the efficiency of this approach. These systems eliminate high-frequency artifacts associated with a wavelet transform of MC residuals by moving the ME/MC into the wavelet domain while using a redundant transform to overcome the problems of shift variance. However, all these prior systems still rely on the traditional block-based ME/MC architecture. In this chapter, we move beyond this block structure to explore the benefits of more general ME/MC geometries.

Specifically, we drive ME/MC with an irregular triangle mesh rather than the traditional block-based structure to build the redundant-wavelet-triangle-mesh (RWTM) system. The motivation for mesh-based ME/MC is that a mesh structure can oftentimes better match the motion of objects in video than can fixed-sized blocks. For example, highly detailed areas should be divided into many small irregularly shaped regions to be individually compensated, whereas larger ME/MC regions can suffice for areas with little detail. This fine-tuning of ME/MC is impossible in traditional block-based approaches since the size of the block is fixed. However, in mesh-based approaches, such as triangle-mesh ME/MC [35], the regions are sized and shaped according to the local level of detail in the image. Specifically, in triangle-mesh ME/MC, triangle vertices, or “control points,” are selected to track edges of objects in the image.

In this chapter, we describe our RWTM system in detail. We first overview the system architecture in Sec. 4.1, and then describe details of the ME/MC process in Secs. 4.2 – 4.4. We will defer experimental evaluation of the performance of the RWTM until Chap. VI. The discussion in this chapter elaborates on our previous publication [25, 26] in which the RWTM system was first developed.

4.1 Overview of the RWTM System

The encoder of our RWTM video-coding system is depicted in Fig. 4.1 and operates as follows. The input image is first transformed using a RDWT, and control points are identified in the previous reference frame by locating the most salient image edges. The motion of these control points from the reference frame to the current frame is estimated in the RDWT domain, and motion vectors are transmitted to the decoder to allow it to track control-point motion. MC is accomplished by first using a triangulation algorithm to generate a triangle mesh on the control points in the reference frame and then using affine transformations to predict, subband by subband, triangles in the current frame from triangles in the reference frame. Residing in the RDWT domain, the motion-compensated residual is itself redundant; consequently, it is downsampled before coding. The final encoding step consists of a wavelet-domain still-image coder; for the experiments presented later in Chap. VI, we use SPIHT [8], but any wavelet-domain still-image coder would suffice.

At the decoder side, motion of the control points is tracked, and a triangulation in the reference frame identical to that used in the encoder is produced. A reconstructed spatial-domain image is produced by inverting the still-image coding, adding on a subsampled RDWT-domain prediction, and inverting the DWT. Finally, a RDWT operation produces the reference-frame subbands for generating the prediction of the next-frame subbands in the RDWT domain. Below, we explore the various components

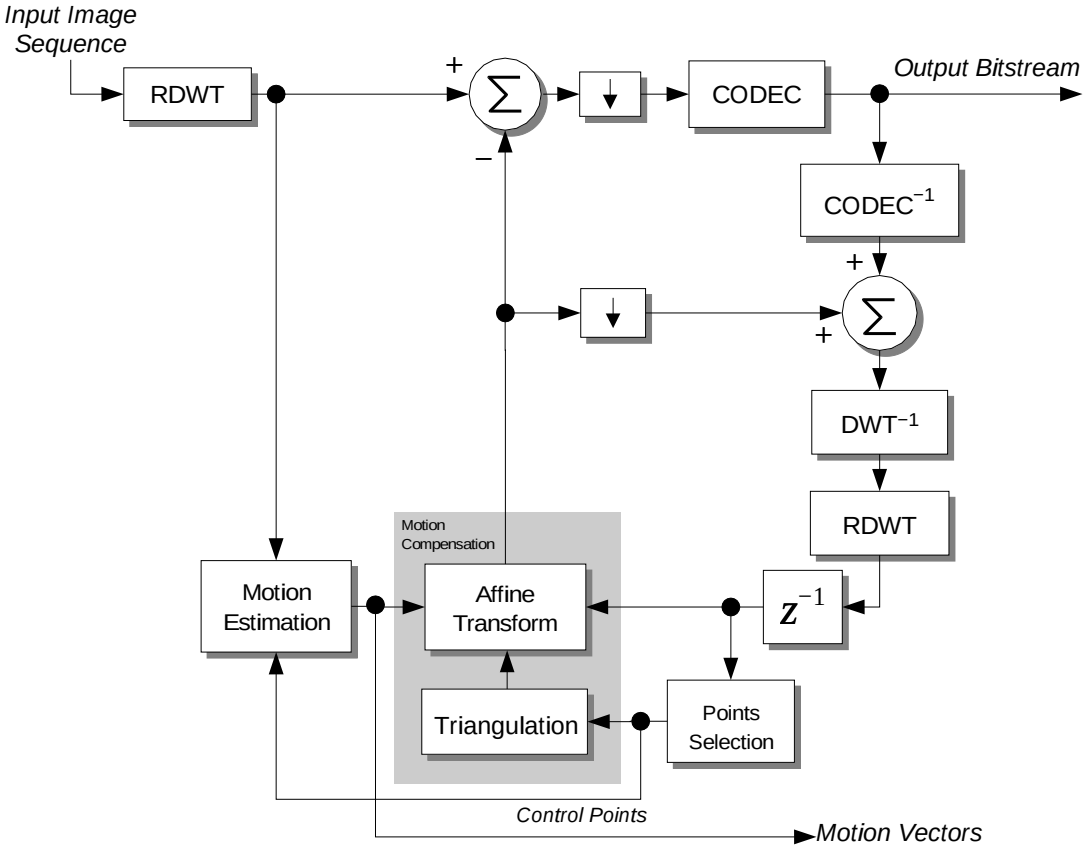


Figure 4.1: The RWTM system.

of our proposed system in greater detail. To a certain extent, our RWTM coder adopts the triangle-mesh ME/MC approach of [35], originally developed in the spatial domain, to the RDWT domain and uses the redundancy inherent in the RDWT to guide mesh placement.

4.2 Selection of Control Points

The choosing of proper control points is crucial to the accuracy of triangle-mesh ME. Typically, one wants control points to track salient image features (e.g., edges). The redundancy of the RDWT facilitates the identification of salient features in an image, especially image edges, since a simple correlation operation can easily accomplish edge identification [36]. Specifically, the direct multiplication of the RDWT coefficients at adjacent scales distinguishes important features from the background due to the fact that wavelet-coefficient magnitudes are correlated across scales. Coefficient-magnitude correlation is well known to exist in the usual critically sampled DWT also; however, the changing temporal sampling rate of the critically sampled DWT makes the calculation of an explicit correlation mask across scales much more difficult [36].

To create the correlation mask for the reference frame, we multiply the vertical (V), horizontal (H), and diagonal (D) bands together across scales and combine the products; i.e.,

$$\begin{aligned}\text{mask}(x,y) = &\left|\prod_{j=J_0}^{J_1} V_j(x,y)\right| + \left|\prod_{j=J_0}^{J_1} H_j(x,y)\right| \\ &+ \left|\prod_{j=J_0}^{J_1} D_j(x,y)\right|, \end{aligned} \tag{4.1}$$

where J_0 and J_1 are the starting and ending scales, respectively, of the correlation operation. We note that calculation of the correlation mask in this manner is possible

due to the fact that each RDWT subband is the same size as the original image. Fig. 4.2 shows the correlation mask for the first frame of the sequence "Susie," where we use the subbands from the two highest-frequency scales in the products above.

To identify control points within the correlation mask, we have devised the following procedure which attempts to place control points on the most salient image edges while ensuring a somewhat uniform spatial spread of the control points across the image. We first determine the global maximum of the mask,

$$\text{mask}_{\text{max}} = \max_{x,y} \text{mask}(x, y), \tag{4.2}$$

and set a threshold, τ, as

$$\tau = \alpha \cdot \text{mask}_{\text{max}}, \tag{4.3}$$

where the threshold parameter α, $0 \leq \alpha \leq 1$, is tailored to a specific sequence for best performance—sequences with faster motion or smaller objects need more control points and thus a smaller value of α. We next divide the mask into $M \times M$ blocks and select at most one point in each block as a control point, processing the $M \times M$ blocks in raster-scan order. Specifically, in each block, we select the point with the largest mask value that is located a distance of d_{min} or greater from an already identified control point. We then compare the mask value of this candidate point to τ—if greater than or equal to τ, we add this candidate point to the set of selected control points. As an example, consider Fig. 4.3, in which four points marked 1 through 4 have the mask values $p1 > p2 > p3 > p4$. 'The 'X" marks two previously selected control points in nearby blocks. The shaded circles are the areas that do not satisfy the minimum-distance criterion, while the raster-scan order is shown by the arrows. Although $p1 > p2 > p3$, points 1 and 2 reside in the shaded areas, and so are discarded. Thus, point 3 is selected

Figure 4.2: Correlation mask for the first frame of “Susie”.

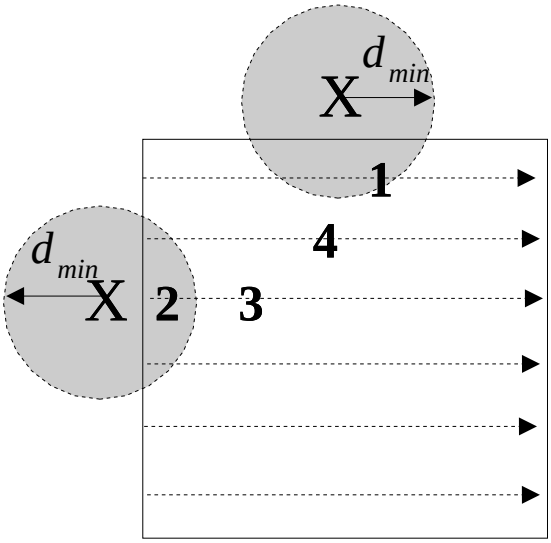


Figure 4.3: Selection of control points in a block.

as the candidate point to compare to τ. However, if point 3 and 4 have the same mask value, i.e., if $p3 = p4$, point 4 will be chosen because of the raster-scan order. Note that we usually end up with each block containing one control point, although it is possible that, because of the thresholding operation, any given block might not contain a control point.

Finally, we add control points equally spaced along the image border to the points chosen via the correlation mask so that the meshed area covers the entire image. These border points always have zero motion vectors and thus are not included in the motion-vector information transmitted by the encoder.

4.3 Motion Estimation

Each non-border control point identified in the reference frame via the correlation mask has an associated motion vector describing the movement of that control point from the reference frame to the current frame. These motion vectors are obtained by finding the best matching point in the current frame for each control point in the reference frame. This match is accomplished by calculating the absolute difference of a $B \times B$ block centered at the control point in the reference frame and blocks in a search window about the control-point location in the current frame, similar to the usual block-based ME process. Our triangle-mesh ME is quite similar to the triangle-mesh ME proposed in [35] in the spatial domain. However, because our ME takes place in the RDWT domain, for a given vector in the search window, we calculate absolute differences for all the subbands at all scales and sum them together to produce a cross-subband, cross-scale distortion, as was proposed in [15] for block-based ME in the RDWT domain. We choose the vector that minimizes this cross-subband, cross-scale distortion as the motion vector for the current control point.

Specifically, the motion vector, (Δ_x, Δ_y), for control point (x, y) in the reference frame is the vector in the search window about (x, y) in the current frame that minimizes the mean absolute error (MAE). Specifically,

$$(\Delta_x, \Delta_y) = \arg \min_{-W \leqslant \Delta_x, \Delta_y \leqslant W} \text{MAE}(x - B/2, y - B/2, \Delta_x, \Delta_y) \tag{4.4}$$

where

$$\text{MAE}(x, y, \Delta_x, \Delta_y) = \frac{1}{B^2} \sum_{k=1}^{B} \sum_{l=1}^{B} \text{AE}(x + k, y + l, \Delta_x, \Delta_y), \tag{4.5}$$

and the absolute error (AE) is

$$\begin{aligned} \text{AE}(x, y, \Delta_x, \Delta_y) = \sum_{j=1}^{J} 2^{-j} \Big\{ & \left| V_j^{cur}(x + \Delta_x, y + \Delta_y) - V_j^{ref}(x, y) \right| \\ & + \left| H_j^{cur}(x + \Delta_x, y + \Delta_y) - H_j^{ref}(x, y) \right| \\ & + \left| D_j^{cur}(x + \Delta_x, y + \Delta_y) - D_j^{ref}(x, y) \right| \Big\} \\ & + 2^{-J} \left| B_J^{cur}(x + \Delta_x, y + \Delta_y) - B_J^{ref}(x, y) \right|, \end{aligned} \tag{4.6}$$

where *cur* and *ref* denote subbands from the current and reference frames, respectively, and B_j, H_j, V_j, and D_j are the baseband, horizontal, vertical, and diagonal subbands, respectively, at scale j. In the search, motion vectors are chosen from a window of size $W > 0$ such that $-W \leq \Delta_x, \Delta_y \leq W$, and the block size, B is assumed to be odd.

4.4 Triangulation and Affine Transform

As in the spatial-domain triangle-mesh ME/MC of [35], after the control points are selected in the reference frame, a triangle mesh is computed using Delaunay triangulation [37]. A single triangle mesh is used for all subbands of the RDWT as depicted in Fig. 4.4; this is possible since each RDWT subband has the same size. MC

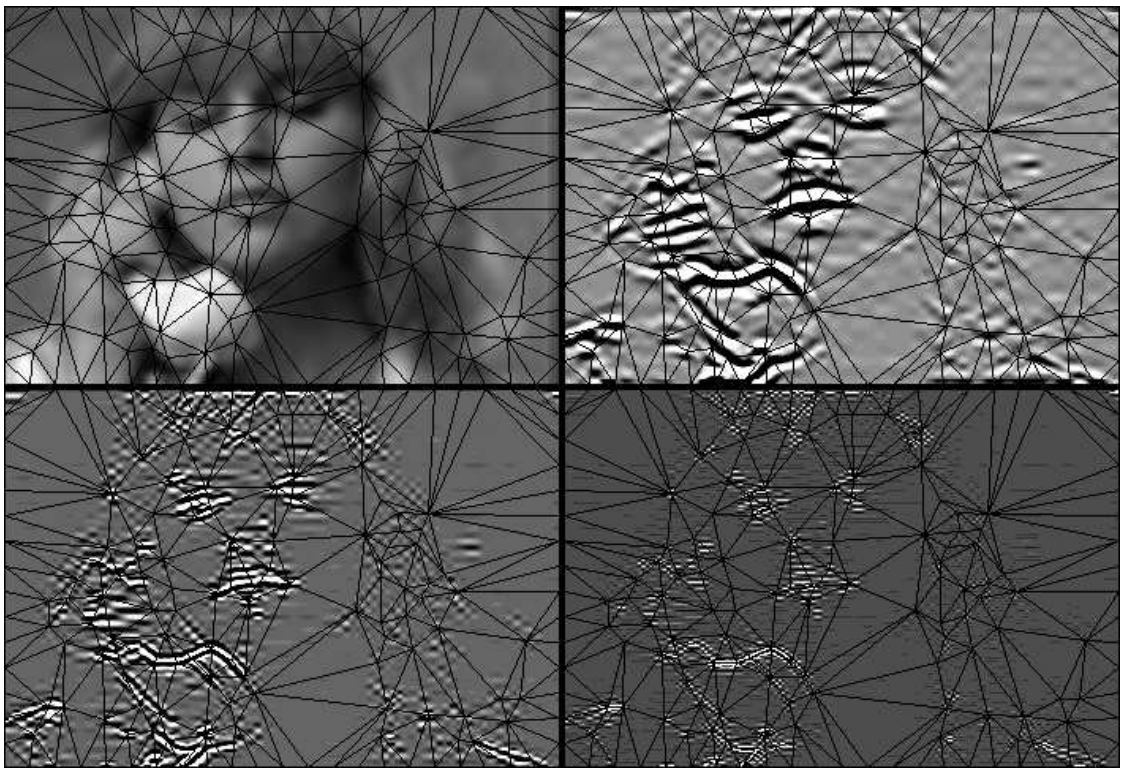


Figure 4.4: RDWT subbands and triangle mesh for the first frame of “Susie”. Clockwise from upper-left: baseband, B_3; vertical subband V_3; subband V_1; and subband V_2. A single triangle mesh is applied to all subbands at all orientations and scales, even though only the vertical subbands are shown here.

proceeds by mapping each triangle in the reference frame into the current frame using an affine six-parameter model as described in [38]; this affine mapping is performed for each triangle in each subband separately.

Affine transforms are widely used in computer graphics. In homogeneous coordinates, affine transforms can represent translation, rotation, and scaling. Consequently, an affine transform can map a point inside one triangle to point inside another triangle. The affine transform is a vector-matrix equation,

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} a_1 & a_2 & a_3 \\ b_1 & b_2 & b_3 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}, \tag{4.7}$$

where x and y are the coordinates of a coefficient in a triangle in the current frame, x' and y' are the corresponding coordinates in the reference-frame triangle, and a_1, a_2, a_3, b_1, b_2, and b_3 are the six parameters of an affine transform that is determined for each pair of current- and reference-frame triangles independently. To determine the transform parameters, we evaluate (4.7) for each of the three vertices of the triangle in the current frame using the known relation between the current- and reference-frame vertices,

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} x \\ y \end{bmatrix} - \begin{bmatrix} \Delta_x \\ \Delta_y \end{bmatrix}, \tag{4.8}$$

to yield six equations in six unknowns. Once the parameters of the transform are determined, it is applied to a coefficient location in the current frame to determine the corresponding location in the reference frame, from which a prediction of the coefficient is determined. Bilinear interpolation is employed to calculate predictions for locations that lie off the RDWT-coefficient grid in the reference frame. In order

to maximize computational efficiency, the affine transformation is carried out only for those coefficients in the current frame that will survive the subsequent RDWT-to-DWT downsampling operation.

In this chapter, we have presented a video-coding system that exploits the RDWT not only for its shift invariance, but also for its ability to facilitate the placement of a triangular mesh for ME/MC via a simple correlation operation. In the next chapter, we develop another use for the redundancy of the RDWT—we use the redundancy of the RDWT to provide multihypothesis prediction for ME/MC.

CHAPTER V

REDUNDANT WAVELET MULTIHYPOTHESIS (RWMH) MOTION COMPENSATION

In the previous chapters, we have seen the RDWT used in a number of video-coding systems, including the RWTM system we developed in Chap. IV. In most of those systems, the redundancy inherent in the RDWT is used exclusively to permit ME/MC in the wavelet domain by overcoming the well known shift variance of the critically sampled DWT ubiquitous to wavelet-based compression methods. The one exception is our RWTM system which additionally exploits the redundancy in the transform to facilitate the fitting of a triangle mesh to the images.

In this chapter, we present an entirely new use for the redundancy in the RDWT. Specifically, we present a system in which transform redundancy is employed to yield multiple predictions of motion that are combined into a single multihypothesis prediction. This system represents a new paradigm in multihypothesis MC (MHMC) wherein diversity in transform phase yields multihypothesis predictions that significantly enhance coding performance.

We first overview the general technique of MHMC in Sec. 5.1, and then present the architecture of our redundant-wavelet multihypothesis (RWMH) system in Sec. 5.2. In Secs. 5.3 – 5.5, we consider a number of refinements to the basic RWMH system, namely a more sophisticated ME/MC search process (Sec. 5.3), and the combining of RWMH with other types of multihypothesis (Secs. 5.4 and 5.5). Finally, in Sec. 5.6, we consider the deployment of triangle meshes as developed in Chap. IV for the RWTM

system with the RWMH framework. The discussion in this chapter elaborates on our previous publications [28, 29] in which the RWMH system was first developed.

5.1 Multihypothesis Motion Compensation (MHMC)

Multihypothesis MC (MHMC) [39] forms a prediction of pixel $s(x, y)$ in the current frame as a combination of multiple predictions in an effort to combat the uncertainty inherent in the ME process. Assuming that the combination of these hypothesis predictions is linear, we have that the prediction of $s(x, y)$ is

$$\tilde{s}(x, y) = \sum_i w_i(x, y)\tilde{s}_i(x, y), \tag{5.1}$$

where the multiple predictions $\tilde{s}_i(x, y)$ are combined according to some weights $w_i(x, y)$. A number of MHMC techniques have been proposed over the last decade. One approach to MHMC is to implement multihypothesis prediction in the spatial dimensions; i.e., the predictions $\tilde{s}_i(x, y)$ are culled from spatially distinct locations in the reference frame. Included in this class of MHMC would be fractional-pixel MC [40] and overlapped block motion compensation (OBMC) [41, 42]. Another approach is to deploy MHMC in the temporal dimension by choosing predictions $\tilde{s}_i(x, y)$ from multiple reference frames. Examples of this class of MHMC are bidirectional prediction (B-frames) as used in MPEG-2 and H.263 and long-term-memory motion compensation (LTMMC) [43]. Of course, it is possible to combine these two classes by choosing multiple predictions that are diverse both spatially and temporally [44]. Note that the calculation of (5.1) in the decoder must be identical to that in the encoder; consequently, it will be necessary to transmit the weights $w_i(x, y)$ to the decoder as side information in the case that the weights are not fixed or not determinable from information already possessed by the decoder. Although implementation dependent, B-frames and LTMMC

typically incur this additional side-information burden while fractional-pixel MC and OBMC do not.

In this chapter, we develop a new class of MHMC by extending the multihypothesis-prediction concept into the transform domain. Specifically, we perform ME/MC in the domain of a redundant, or overcomplete, wavelet transform, and use multiple predictions that are diverse in transform phase. First, we observe that each of the critically sampled DWTs within a RDWT will "view" motion from a different perspective. Consequently, if motion is predicted in the RDWT domain, the inverse RDWT forms a multihypothesis prediction in the form of (5.1). Specifically, for a J-scale RDWT, the reconstruction from DWT i of the RDWT is $\tilde{s}_i(x, y)$, $0 \leq i < 4^J$, while $w_i(x, y) = 4^{-J}$, $\forall i$. Below, we present our RWMH video-coding system [28] that performs MHMC in precisely this fashion.

An interesting aspect of the phase-diversity approach to MHMC is that low-resolution information is inherently predicted with a greater number of hypotheses which corresponds to the greater difficulty inherent in estimating motion in signals with spatially low resolution. Additionally, since the weighting of the individual predictions is carried out implicitly in the form of an inverse transform, no side information need be sent to the decoder. Finally, we show below that our phase-diversity MHMC functions complementary to other forms of MHMC; specifically, we combine RWMH with two forms of spatial-diversity MHMC to achieve performance superior to that of either class of MHMC operating alone.

5.2 Overview of the RWMH System

The encoder of our RWMH video-coding system is depicted in Fig. 5.1. The current and reference frames are transformed into RDWT coefficients, and both ME and MC take place in this redundant-wavelet domain.

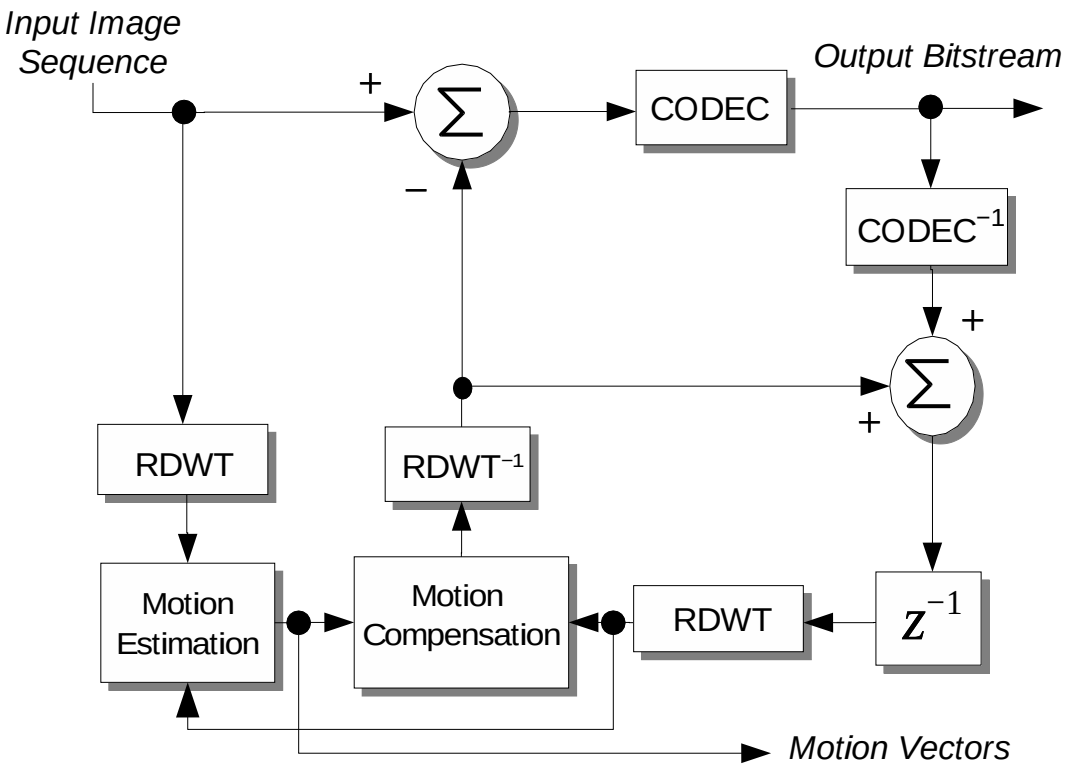


Figure 5.1: The RWMH coder. z^{-1} = frame delay, *CODEC* is any still-image coder.

In a J-scale RDWT decomposition, each $B \times B$ block in the original spatial domain corresponds to $3J + 1$ blocks of the same size, one in each subband. The collection of these co-located blocks is called a set. Each set contains all the different phases of RDWT coefficients. In the ME procedure, block matching is used to determine the motion of each set as a whole. Specifically, a block-matching procedure uses a cross-subband distortion measure that sums absolute errors for each block of the set similar to the cross-subband ME procedure of [15]. However in our metric, the coefficients from all phases in both current and reference frames contribute to the distortion measurement, in contrast to the metric of [15], in which only coefficients from a single critically subsampled DWT in the current frame contribute. Specifically, the motion vector for the set located at (x, y) is

$$(\Delta_x, \Delta_y) = \arg \min_{-W \leqslant \Delta_x, \Delta_y \leqslant W} \text{MAE}(x, y, \Delta_x, \Delta_y), \tag{5.2}$$

where

$$\text{MAE}(x, y, \Delta_x, \Delta_y) = \frac{1}{B^2} \sum_{k=1}^{B} \sum_{l=1}^{B} \text{AE}(x + k, y + l, \Delta_x, \Delta_y). \tag{5.3}$$

The absolute error (AE) is

$$\begin{aligned} \text{AE}(x, y, \Delta_x, \Delta_y) = \frac{1}{2} \Big\{ & \left| V_1^{cur}(x, y) - V_1^{ref}(x + \Delta_x, y + \Delta_y) \right| \\ & + \left| H_1^{cur}(x, y) - H_1^{ref}(x + \Delta_x, y + \Delta_y) \right| \\ & + \left| D_1^{cur}(x, y) - D_1^{ref}(x + \Delta_x, y + \Delta_y) \right| \\ & + \left| B_1^{cur}(x, y) - B_1^{ref}(x + \Delta_x, y + \Delta_y) \right| \Big\}, \end{aligned} \tag{5.4}$$

where *cur* and *ref* denote subbands from the current and reference frames, respectively, and B_j, H_j, V_j, and D_j are the baseband, horizontal, vertical, and diagonal subbands, respectively, at scale j. A window $[-W, W]$ is used for the block search, and, to speed

the search, a 1-scale RDWT, rather than the full J-scale transform, is used for the block-matching ME procedure.

After the ME search has determined motion vectors for each set, a motion-compensated frame is then created in the RDWT domain using the same motion vector for each block of the set. The inverse RDWT is performed on this RDWT-domain motion-compensated frame, combining the multiple phases into a spatial-domain multihypothesis prediction. This spatial-domain prediction is subtracted from the current frame, and the residual is coded. This final encoding step consists of a still-image coder; for the experiments later in Chap. VI, we use SPIHT [8], but any still-image coder, wavelet-based or otherwise, would suffice.

At the decoder side, a spatial-domain residual image is produced by inverting the still-image coding. The reconstructed image is obtained by adding the prediction image, which is the same as that at the encoder side, to the residual image. Reconstruction is necessarily followed by a RDWT operation to produce the reference-frame subbands for generating the prediction for the next frame in the RDWT domain.

5.3 Phase-optimal Vector Search

In the system as described above, each critically sampled DWT in the RDWT yields a different prediction of the motion of the frame, and these separate predictions are combined into a single multihypothesis prediction via the inverse-RDWT operation. However, all of the constituent DWTs use the same motion-vector field to describe the motion. More accurate prediction results when motion fields are optimized to each DWT, albeit at the expense of additional rate.

Specifically, we propose a multiscale hierarchical ME scheme which assigns to each phase at each scale a different motion-vector field. This hierarchical ME approach bears some resemblance to traditional hierarchical ME/MC [45]; however, in our case, the

hierarchy starts at high resolution and proceeds toward low resolution. That is, we refine, for each phase at each scale, the motion vectors resulting from the block search described above starting at scale 1 and continuing to scale J. Consider a block of size $B \times B$ at scale 1 and call the motion vector V determined for this block using the procedure above the "all-phase" motion vector. We perform a block search with a small window of $[-W', W']$ about the location indicated by the all-phase motion vector for this block. However, in the cross-subband distortion metric for this search, we include only those coefficients belonging to phase 0; additionally, this distortion metric is limited to only the subbands at scale 1. This search will yield a "single-phase" motion vector, $V_{1,0}$. We repeat this process for the other three phases yielding single-phase vectors $V_{1,1}$, $V_{1,2}$, and $V_{1,3}$. In addition to V, for each block, we transmit "refinement" vectors

$$v_{1,i} = V_{1,i} - V \tag{5.5}$$

for each phase i.

For scales $j > 1$, we can use the vector $V_{1,i}$ for all the phases that are descendants of phase i at scale 1. Alternatively, we can apply the above procedure to further refine the motion estimate for higher scales. For example, in scale 2, we search in a $[-W', W']$ window about $V_{1,0}$ to find the four motion vectors for the four phases at scale 2 that are children of phase 0 at scale 1. Note that, for each additional scale of refinement, the number of additional refinement vectors that need to be sent increases by a factor of 4—there will be 4 refinement vectors per set for one scale of refinement, 16 for two scales of refinement, etc. Fig. 5.2 illustrates this multiscale motion-vector refinement procedure. After this hierarchical search, for each set, we will obtain an "all-phase" search vector V followed by a number of refinement vectors for each phase at each scale.

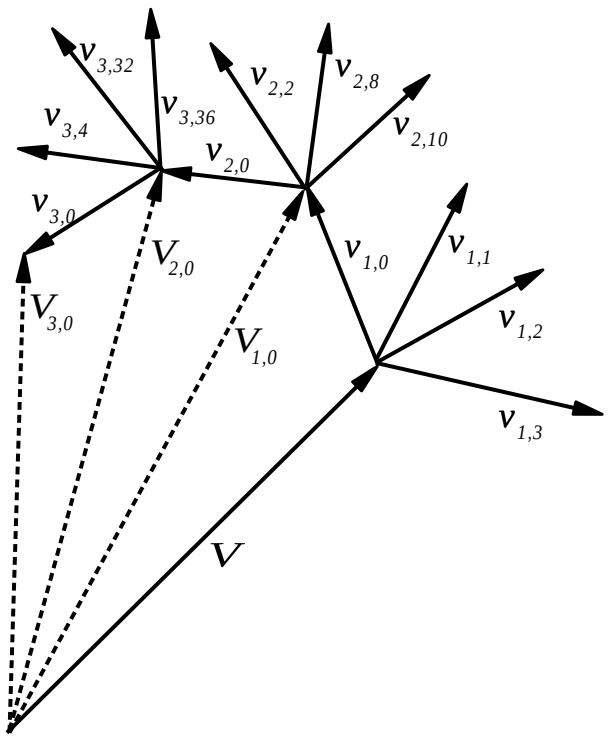


Figure 5.2: Hierarchical refinement of motion vectors.

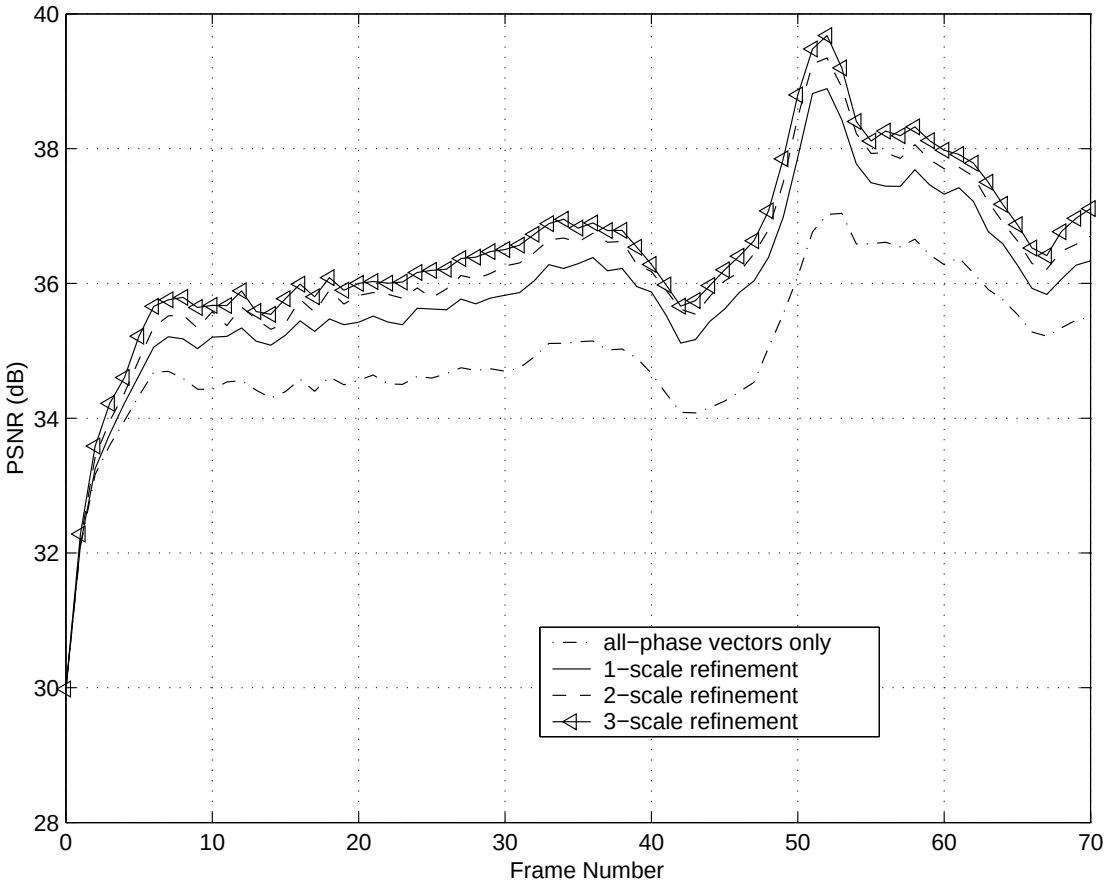


Figure 5.3: PSNR performance for “Susie” at 0.12 bpp using different numbers of scales of motion-vector refinement (refinement vectors not included in the rate).

The PSNR performance of the RWMH system improves as more scales of refinement vectors are used; Fig. 5.3 illustrates this improvement for several scales of refinement for the "Susie" sequence at a fixed rate. We observe diminishing returns—the amount of PSNR improvement decreases with each additional scale of refinement. However, since the number of refinement vectors grows dramatically with each additional scale of refinement, we have concluded that the cost in rate does not justify the incremental increase in PSNR performance beyond one scale of refinement. Thus, for the experiments later in Chap. VI, we transmit for each set of blocks one all-phase motion vector and four single-phase refinement vectors. W' is chosen so that $W' \ll W$ in order to minimize the rate burden associated with the refinement vectors.

5.4 Combining RWMH with Spatial-diversity Multihypothesis

The RWMH system is a generalization to the wavelet-domain ME/MC approaches based on [15] which are based on single-hypothesis prediction. In this section, we further enhance performance by increasing the number of hypotheses. That is, we combine our RWMH technique with other multihypothesis methods, specifically, serveral that employ spatial-diversity. The results of Chap. VI will show that the two classes of multihypothesis prediction—phase-diversity and spatial-diversity—complement each other such that their combination yields performance superior to that of either class alone. This synergy is possible since the RDWT preserves the spatial relation of the original image. Two prominent paradigms for spatial-diversity multihypothesis are overlapped block MC (OBMC) and sub-pixel accuracy. In order to reduce computation complexity and avoid transmitting excessive overhead information, we choose not to use refinement vectors as described in the previous section in conjuction with the spatial-diversity multihypothesis approaches.

5.4.1 Overlapped Block Motion Compensation (OBMC)

In conventional block-based motion prediction, each block is motion-compensated independently of other blocks. Consequently, the motion vector for a given block is not necessarily the same as the vectors of its adjacent blocks even though it is likely that the motion of the neighboring blocks is similar. This disparity causes discontinuity among consecutive blocks in the motion-compensated frame, a major cause of blocking artifacts. To mitigate this effect, OBMC was proposed in [41]. In OBMC, a weighted sum of multiple predictions is used to motion-compensate each block. Let $P_i(x, y)$ be a prediction of the current block obtained from a reference block, which is weighted by matrix $W_i(x, y)$. In OBMC, the P_i predictions of the current block are generated by using the motion vectors of neighboring blocks. Then, the weighted prediction is,

$$\tilde{P}_i(x, y) = P_i(x, y) \times W_i(x, y), \tag{5.6}$$

where $\times$ represents element-by-element multiplication. The final prediction of the current block is

$$P(x, y) = \sum_i \tilde{P}_i(x, y), \tag{5.7}$$

which is a form of MHMC when compared to (5.1) in Sec. 5.1.

Since we drive our RWMH with a block-based search, blocking artifacts will occur in the RDWT-domain motion-compensated frame, causing coding inefficiency in the corresponding residual image. OBMC as developed in [41, 42] is a simple and straightforward solution to this problem. It is well known that OBMC in the spatial domain can increase performance greatly; thus, it has been adopted in the H.263 standard [5, 6]. Since RDWT coefficients retain the "spatial coherence" of the original image (Sec.2.2), OBMC in the RDWT domain is straightforward. Since there are $3J+1$

subbands for a J-scale decomposition, we must deploy OBMC in all the subbands in the RDWT domain following the same procedure.

We follow the simple OBMC scheme of H.263 [5, 6] in order to implement OBMC within RWMH. In each subband, we define 16×16 macroblocks which are further divided into four 8×8 blocks. As illustrated in Fig. 5.4, the vectors of the four blocks within a macroblock and the neighboring eight blocks are used to form a prediction of the current macroblock. The prediction of the current block from the reference frame is a weighted sum of three blocks obtained through the motion vector for the current block (Δ_x, Δ_y) and the motion vectors of the two nearest neighboring blocks, one from the vertical direction (Δ_x^V, Δ_y^V) and one from the horizonal direction (Δ_x^H, Δ_y^H). As illustrated in Fig. 5.4, according to different location of those prediction blocks, there are three 8×8 matrices of weighting values illustrated in Figs. 5.5, 5.6, and 5.7. The prediction $P(x, y)$ is an 8×8 block,

$$P(x, y) = \tilde{P}(x, y) \times W(i, j) + \tilde{P_V}(x, y) \times W_V(i, j) + \tilde{P_H}(x, y) \times W_H(i, j)/8, \quad (5.8)$$

where $p(x + \Delta_x^k, y + \Delta_y^k)$ is the prediction value at position $(x + \Delta_x^k, y + \Delta_y^k)$ in the reference frame, and

$$\tilde{P}(x, y) = p(x + \Delta_x, y + \Delta_y), \quad (5.9)$$

$$\tilde{P_V}(x, y) = p(x + \Delta_x^V, y + \Delta_y^V), \quad (5.10)$$

$$\tilde{P_H}(x, y) = p(x + \Delta_x^H, y + \Delta_y^H). \quad (5.11)$$

The resulting system is denoted as RWMH-OBMC.

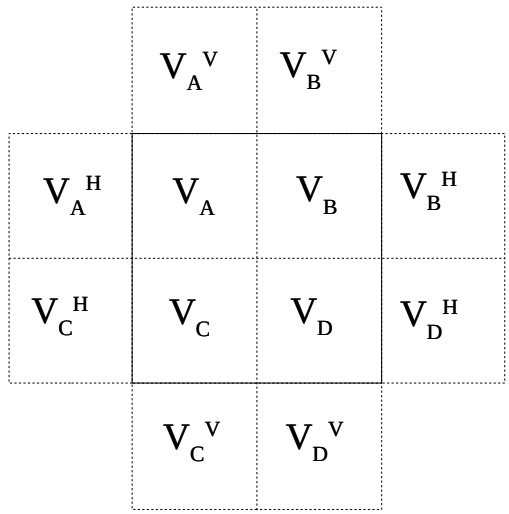


Figure 5.4: Block V_x is predicted using motion vector for block $V_x(\Delta_x, \Delta_y)$, and the motion vectors for blocks V_x^V and V_x^H ((Δ_x^V, Δ_y^V) and (Δ_x^H, Δ_y^H), respectively). Here, $x \in \{A, B, C, D\}$.

4	5	5	5	5	5	5	4
5	5	5	5	5	5	5	5
5	5	6	6	6	6	5	5
5	5	6	6	6	6	5	5
5	5	6	6	6	6	5	5
5	5	6	6	6	6	5	5
5	5	5	5	5	5	5	5
4	5	5	5	5	5	5	4

Figure 5.5: Weighting values, W, for prediction with motion vector of current block.

2	2	2	2	2	2	2	2
1	1	2	2	2	2	1	1
1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1
1	1	2	2	2	2	1	1
2	2	2	2	2	2	2	2

Figure 5.6: Weighting values, W_V, for prediction with motion vectors of the blocks on top or bottom of current block.

2	1	1	1	1	1	1	2
2	2	1	1	1	1	2	2
2	2	1	1	1	1	2	2
2	2	1	1	1	1	2	2
2	2	1	1	1	1	2	2
2	2	1	1	1	1	2	2
2	2	1	1	1	1	2	2
2	1	1	1	1	1	1	2

Figure 5.7: Weighting values, W_H, for prediction with motion vectors of the blocks to the left or right of current block.

5.4.2 Sub-pixel Accuracy

The modern generation of ME/MC algorithms specify motion vectors with an increased resolution, that is, with fractional-pixel accuracy. Although increased motion-vector resolution entails a larger bit-rate overhead, the increased accuracy yields better motion prediction, a small MC residual, and a reduced reconstruction distortion. Usually, the increased distortion performance will more than offset the added rate overhead for a net coding gain. For example, half-pixel accuracy has been successfully used in MPEG-1 [3], MPEG-2 [4], and H.263 [5, 6]. In half-pixel mode, the motion vectors take full- and half-pixel values. In the case of a half-pixel motion vector, the

position in the reference frame to which the vector points is between integer-pixel positions. Since pixels are assumed to lie only at integer positions, the reference frame has no pixel value associated with half-pixel positions; consequently, interpolation is used to construct such values off the integer-pixel grid. Extensive literature has shown that a simple bilinear interpolation can achieve good performance for half-pixel accuracy. However, to further increase the accuracy to quarter-pixel, bilinear interpolation of the half-pixel values will not improve performance since the additional motion-vector overhead usually outweights the potential reduction in distortion. Instead of mere interpolation of only the nearby half-pixel values, an improved sample-interpolation process adopted in MPEG-4 [7] increases the coding efficiency by taking into account aliasing components.

Half-pixel accuracy in the wavelet domain has been implemented in [17], and performance superior to that of full-pixel ME/MC was observed. In our work, we investigate increasing the resolution of the RWMH-OBMC system to quarter-pixel accuracy and find that the quarter-pixel technique employed in MPEG-4 [7] in the spatial domain can be directly applied to RDWT coefficients. Specifically, the two-step procedure in MPEG-4 [7] for the quarter-pixel interpolation is illustrated in Fig. 5.8. First, a 1D 8-tap filter is applied on the integer-pixel values to generate values on the half-pixel grid. Let the integer-grid RDWT coefficients be S_j, where scale j is $1 \leqslant j \leqslant J$, and $S \in \{B, H, V, D\}$. The 8-tap interpolation filter is $f[n]$,

$$f[0] = \frac{160}{256}, \tag{5.12}$$

$$f[1] = \frac{-48}{256}, \tag{5.13}$$

$$f[2] = \frac{24}{256}, \tag{5.14}$$

$$f[3] = \frac{-8}{256}, \tag{5.15}$$

$$f[n] = 0, \qquad n \geqslant 4, \tag{5.16}$$

$$f[-n] = f[n-1]. \tag{5.17}$$

Assume subband S_j is of size $M \times N$. The interpolation filter is first applied horizontally to subband S_j to produce $\tilde{S}_j$ of size $M \times 2N$,

$$\tilde{S}_j(x,y) = \begin{cases} S_j(\frac{x}{2}, y) & x \text{ even}, \\ r_y[n] * f[n]\Big|_{n=\lceil \frac{x}{2} \rceil} & x \text{ odd}, \end{cases} \tag{5.18}$$

where $r_y[n]$ is the yth row of $S_j(x,y)$. Next, the filter is applied vertically to produce $\hat{S}_j$ of size $2M \times 2N$,

$$\hat{S}_j(x,y) = \begin{cases} \tilde{S}_j(x, \frac{y}{2}) & y \text{ even}, \\ c_x[n] * f[n]\Big|_{n=\lceil \frac{y}{2} \rceil} & y \text{ odd}, \end{cases} \tag{5.19}$$

where $c_x[n]$ is the xth column of $\tilde{S}_j(x,y)$. Next, the quarter-pixel coefficients are calculated by bilinear interpolation of the half-pixel coefficients. Again, this process is carried out identically in each RDWT subband.

After expanding the reference frame to the quarter-pixel accuracy, we search for the best match for each macroblock to obtain quarter-pixel accurate motion vectors. The integer part of these vectors is transmitted using Table 3 of H.261 (VLC table for MVD) [2], and the fractional part of the vectors is sent by appending a two-bit fixed-length binary code to the Huffman codeword. The overhead bits needed to code the vectors in this manner is nearly the same as in H.263 [5].

◯	+	△	+	◯	
+	+	+	+	+	◯ Integer-pixel position.
△	+	△	+	△	△ Half-pixel position.
+	+	+	+	+	+ Quarter-pixel position.
◯	+	△	+	◯	

Figure 5.8: Quarter-pixel accuracy obtained by filtering and interpolation.

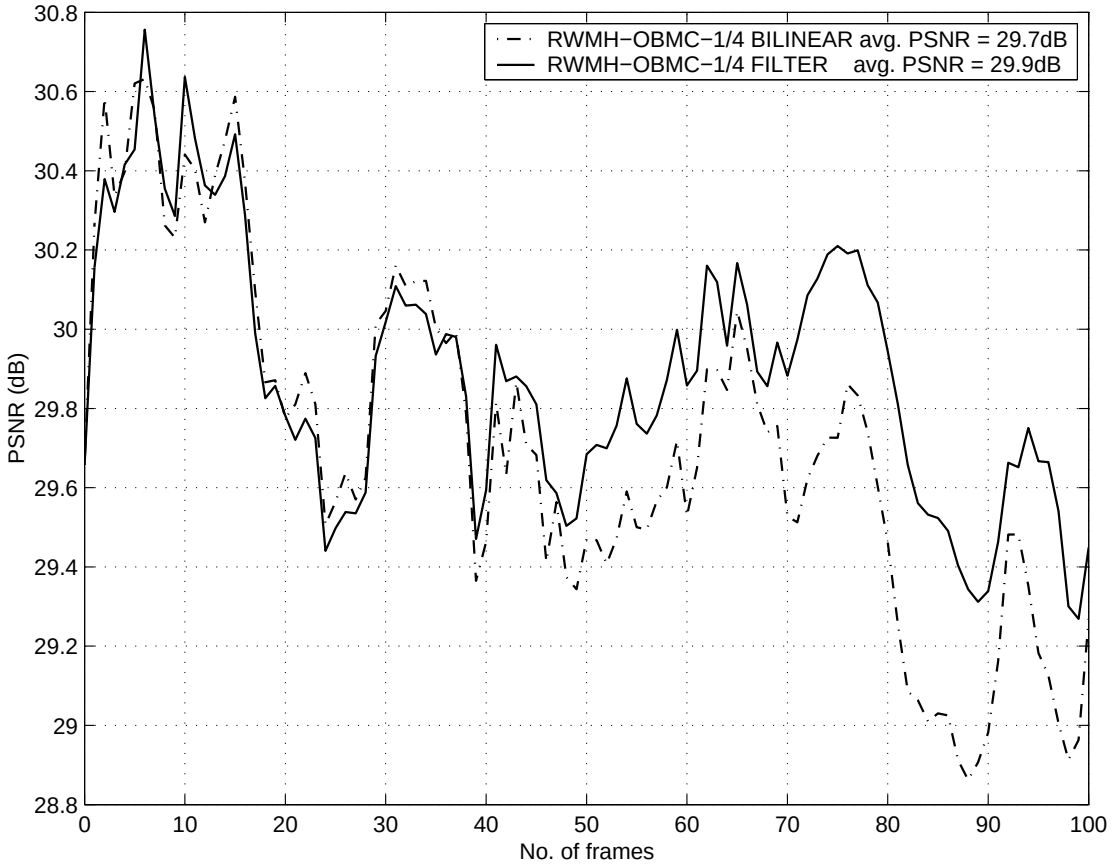


Figure 5.9: Frame-by-frame PSNR for “Football” at 0.5 bpp (1.3 Mbps). Comparing quarter-pixel accuracy ME/MC implemented via bilinear interpolation and the MPEG-4 [7] filter procedure.

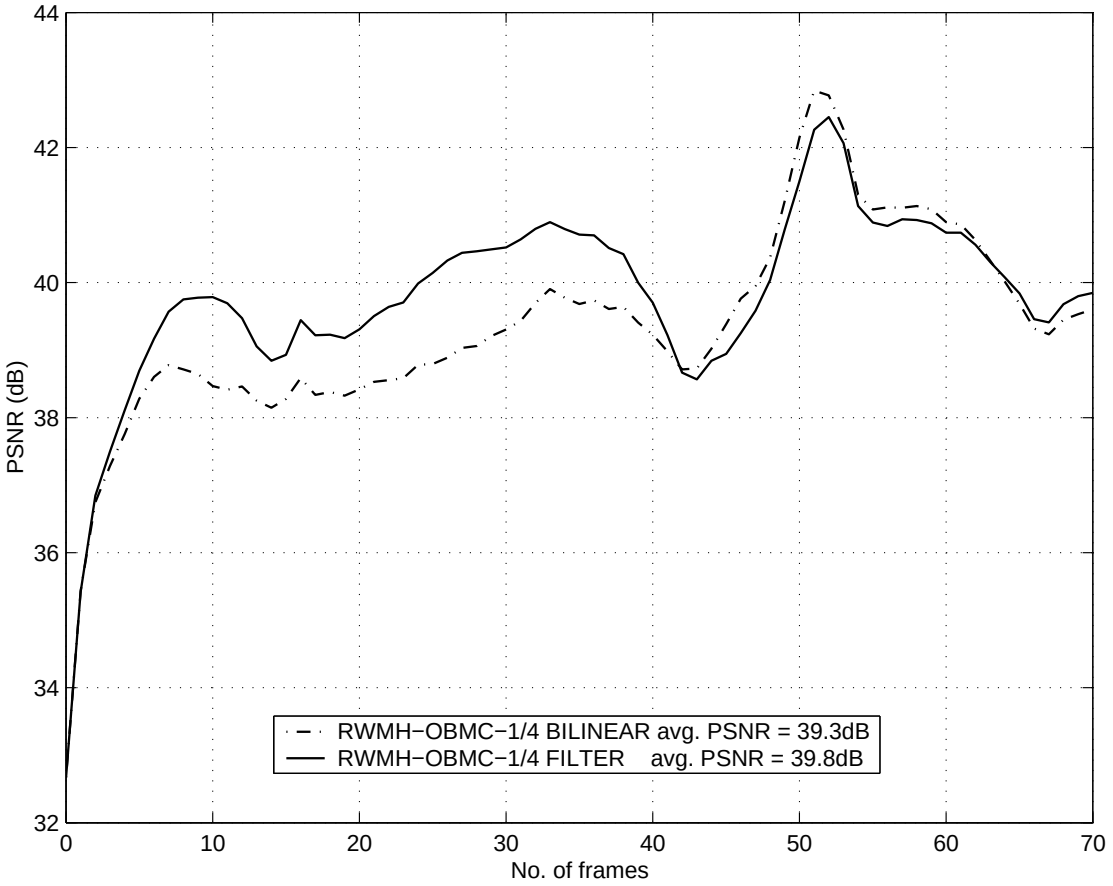


Figure 5.10: Frame-by-frame PSNR for "Susie" at 0.25 bpp (634 kbps). Comparing quarter-pixel accuracy ME/MC implemented via bilinear interpolation and the MPEG-4 [7] filter procedure.

As in [17], simple bilinear interpolation between adjacent coefficients produces values on the half-pixel grid. Bilinear interpolation applied directly on the integer-grid RDWT coefficients can also be employed to produce values on the quarter-pixel grid; however the efficiency of this approach is less than that achieved through use of interpolation filters as described above, as demonstrated in Figs. 5.9 and 5.10.

5.5 Combining RWMH with Temporal-diversity Multihypothesis

Like as with spatial-diversity multihypothesis, RWMH can also be deployed in conjunction with temporal-diversity approaches. Specifically, we choose long-term-memory motion compensation (LTMMC) [43] to combine with our RWMH system. The new system is denoted as RWMH-LT.

LTMMC uses multiple reference frames to predict the current frame as illustrated in Fig. 5.11. One approach to LTMMC is to find the best prediction of a block from a number of reference frames, as shown in Fig. 5.12(a), in which case, the index of the chosen frame is transmitted as overhead information. Another approach is a multihypothesis LTMMC which invokes a combination of several reference frames to predict the current frame, as shown in Fig. 5.12(b). We use this latter approach to generate the predicted image, using the three previous frames. In order to save bits in coding overhead information, we set the weights in MHMC equation (5.1) as

$$\begin{aligned} w_1(x,y) &= 0.5, \\ w_2(x,y) &= 0.25, \\ w_3(x,y) &= 0.25. \end{aligned} \tag{5.20}$$

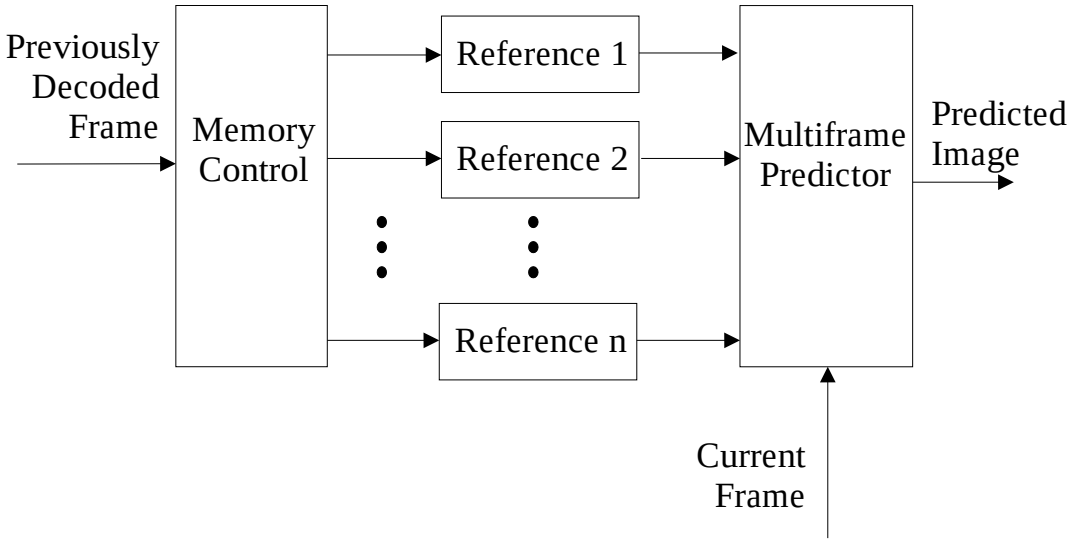


Figure 5.11: Long-term-memory motion compensation (LTMMC) [43] predictor.

Consequently there is no need to transmit the weights, but we still need to transmit a total of three sets of motion vectors, one for each reference frame.

5.6 Combining RWMH with RWTM

As a final refinement of the RWMH approach, we revisit the RWTM system of Chap. IV. That is, the RWTM system employed the redundancy of the RDWT to facilitate triangle-mesh ME/MC, whereas the RWMH systems considered thus far in this chapter employ the traditional block geometry. In this section we build a new system, RWTMMH, by combining RWMH and RWTM. The encoder of our RWTMMH video-coding system is depicted in Fig. 5.13. After the triangle-mesh ME/MC in the RDWT domain, we apply an inverse RDWT to form a multihypothesis prediction which averages the phase-diversity predictions. Later, we will see that this multihypothesis RWTM approach outperforms our former single-hypothesis approach introduced in Chap. IV.

In this chapter, we introduced the concept of phase-diverstiy multihypothesis which exploits the redundancy of the RDWT to increase prediction accuracy of ME/MC. We developed a number of video-coding systems based on this notion of RWMH, employing other multihypothesis strategies in conjunction with our proposed approach. In the next chapter, we evaluate the performance of these systems against other RDWT-based video coders.

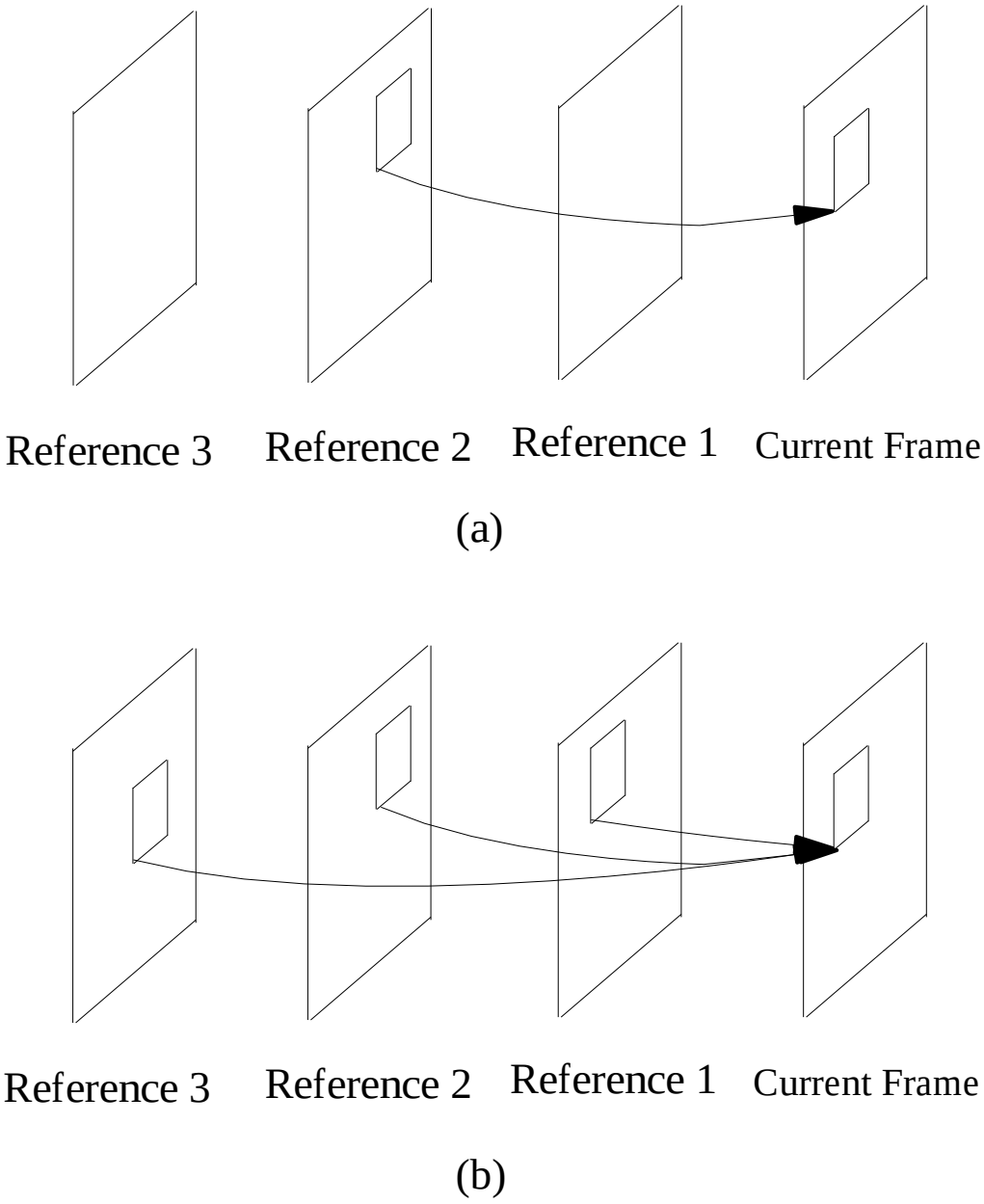


Figure 5.12: (a) Long-term-memory motion compensation. One previous frame is chosen to predict the current block. (b) Multihypothesis long-term-memory motion compensation. Three previous frames are linearly combined to predict the current block.

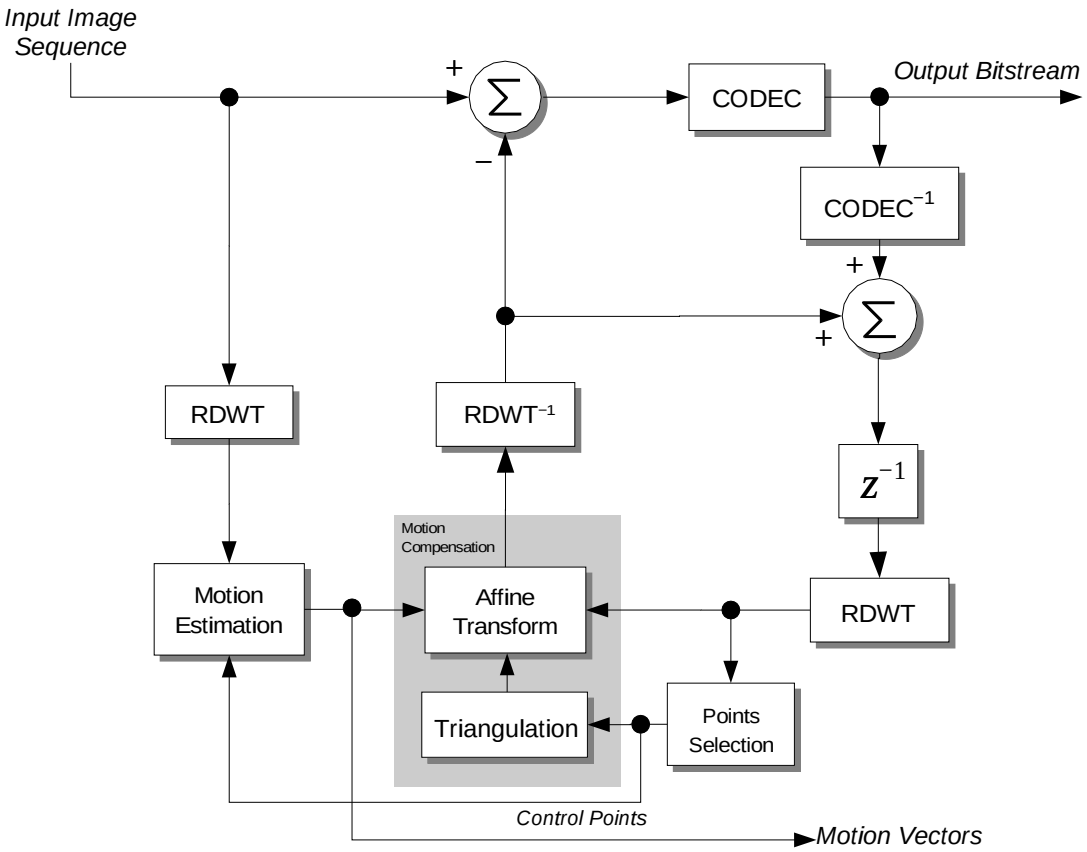


Figure 5.13: The RWTMMH coder. z^{-1} = frame delay, *CODEC* is any still-image coder.

CHAPTER VI

RESULTS AND OBSERVATIONS

In this chapter, we present a body of experimental results to evaluate the effectiveness of the RWTM system proposed in Chap. IV and the RWMH system proposed in Chap. V. We first show, in Secs. 6.1 and 6.2, respectively, that the RWTM and RWMH systems offer performance significantly superior to the RDWT-Block system of [15] which was described in Chap. III as the foundation of all prior proposed uses of the RDWT in video coding. Then, in Secs. 6.2 and 6.3, we investigate the use of spatial and temporal diversity, respectively, in conjunction with the phase diversity of the RWMH system. Finally, in Sec. 6.4, we evaluate the performance gains possible through the merging of RWMH with RWTM.

6.1 The RWTM System

Experimental results use the 100-frame "Football" SIF sequence, the 70-frame "Susie" SIF sequence, the 300-frame "Mother & Daughter" CIF sequence, and the 300-frame "Coastguard" CIF sequence. All sequences are grayscale and have a temporal sampling of 30 frames/sec. (noninterlaced). The first frame is intra-encoded (I-frame) while all subsequent frames use ME/MC (P-frames). All wavelet transforms (DWT and RDWT) use the Cohen-Daubechies-Feauveau 9-7 filter [14] with symmetric extension and a decomposition of $J = 3$ levels. Unless otherwise indicated, all ME/MC methods use integer-pixel accuracy and approximately the same number of motion vectors per frame. The core compression engine in all experiments is the QccPack [46]

implementation of SPIHT [8]; since SPIHT produces an embedded coding, each frame of the sequence is coded at exactly the specified target rate.

For our proposed RWTM system, we calculate the correlation mask of (4.1) using $J_1 = J = 3$ for all sequences. We use $J_0 = 1$ for "Football" and $J_0 = 2$ for the other sequences. We select control points in the mask using $M \times M$ blocks, ensuring compliance with a minimum distance of d_{min} and a threshold as in (4.3). For the experiments here, we use $M = 16$ and $d_{\text{min}} = 8$ for all sequences. We use a threshold parameter of $\alpha = 0$ for "Football" and $\alpha = 0.972$ for the other sequences. To estimate motion of the control points, we use a block of size $B \times B$ centered around the control point in the reference and search in a window of $\pm W$ in the current frame. For the results here, we use $B = 17$ and $W = 15$ for all sequences.

We compare our proposed RWTM technique to both block- and mesh-based ME/MC in both the spatial and wavelet domains. Specifically, in these results, "Spatial Block" refers to block-based ME/MC in the spatial domain, the traditional method employed in video-coding standards, followed by a full-frame, critically sampled DWT and SPIHT coding. "Spatial Mesh" is an irregular triangle-mesh ME/MC in the spatial domain [35], followed by full-frame, critically sampled DWT and SPIHT. "RDWT Block" is the technique proposed in [15] and used subsequently in [16–18, 21] which employs block-based ME/MC to locate DWT blocks in the RDWT domain.

Frame-by-frame PSNR profiles for "Susie" and "Football" are shown in Figs. 6.1 and 6.2. Original and reconstructed frames are shown for "Football" in Fig. 6.3. Finally, PSNR values averaged over all frames of the sequences are tabulated in Table 6.1 for a fixed bit rate.

The experimental results shown in Table 6.1 and Figs. 6.1 and 6.2 indicate that our proposed RWTM method outperforms other ME/MC techniques operating in both the spatial and wavelet domains. In terms of average PSNR performance (Tab. 6.1), RWTM

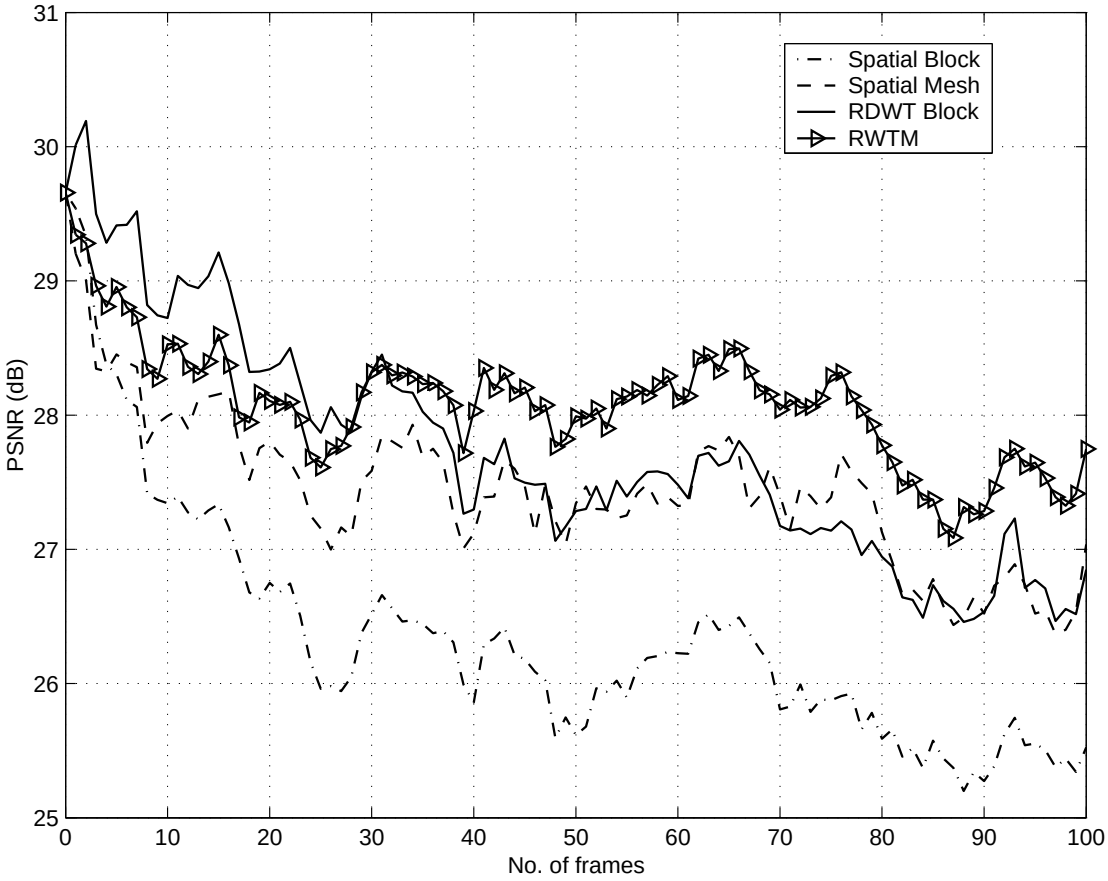


Figure 6.1: Comparison of RWTM to other methods—frame-by-frame PSNR for "Football" at 0.5 bpp (1.3 Mbps).

Table 6.1: Distortion averaged over all frames of the sequence.

	PSNR (dB)			
	Spatial Block	Spatial Mesh	RDWT Block	RWTM
Football†	26.3	27.4	27.9	28.3
Susie	36.0	37.5	37.4	37.8
Mother & daughter	40.2	41.6	40.8	41.7
Coastguard	28.1	28.0	28.9	28.7

Rate is 0.25 bpp except †, which is 0.5 bpp.

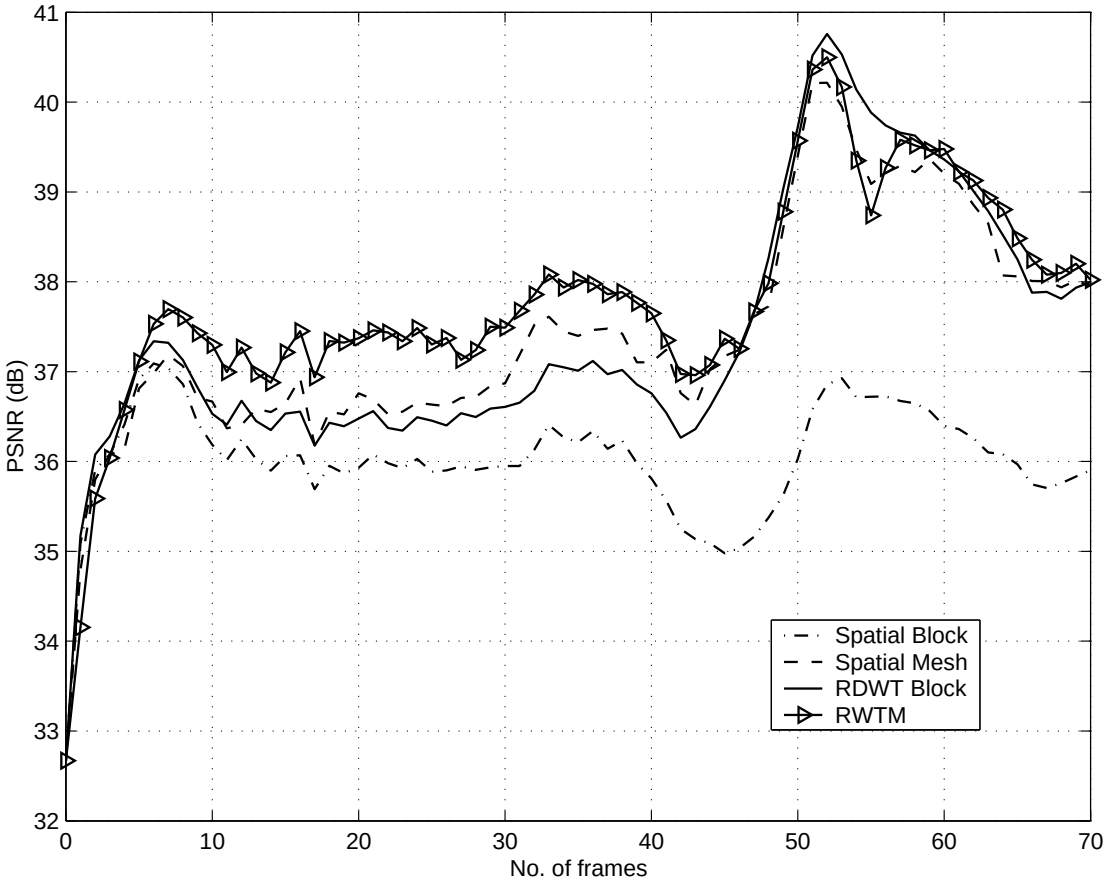
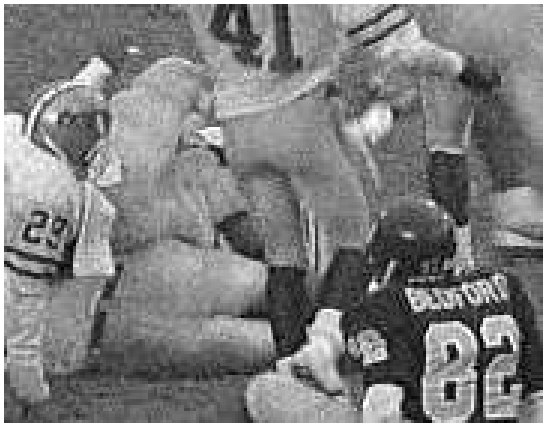


Figure 6.2: Comparison of RWTM to other methods—frame-by-frame PSNR for "Susie" at 0.25 bpp (634 kbps).

(a)



(b)

(c)

(d)

(e)

Figure 6.3: Original and reconstructed images for frame 66 of “Football” (cropped to show detail). (a) Original, (b) Spatial Block, (c) Spatial Mesh, (d) RDWT Block, (e) RWTM.

outperforms its nearest competitor ("RDWT Block") by 0.4 dB for both the fast-motion "Football" and the slow-moving "Susie" sequences. It is interesting to note that our combination of triangle-mesh ME/MC and RDWT-based ME/MC outperforms either technique applied alone.

The success of our approach lies in that the shift invariance of the RDWT makes it an ideal candidate for the implementation of ME/MC in the wavelet domain. This fact has been exploited previously by others [15–18, 21] using the coefficient-tree representation of the RDWT wherein each root-to-leaf path represents a distinct critically sampled DWT of a different phase. In these techniques, the ME/MC procedure "switches" between root-to-leaf paths in the RDWT coefficient tree as the phase of the motion under consideration changes. In our system, on the other hand, we preserve the spatial coherence of the coefficients, thereby permitting easy identification of control points through a simple correlation operation—spatial-domain mesh-based techniques typically employ a more costly convolution operator to accomplish this same task. In addition, we exploit all phase information in the current as well as reference frames to determine motion, whereas other RDWT techniques use a critically sampled wavelet-domain representation of the current frame.

6.2 The RWMH System

In this section the test sequences, wavelet filter, and coding engine are the same as for the RWTM system. The RDWT-based MHMC procedure uses $B = 16$, $W = 15$, and $W' = 1$. All rate figures include all motion-vector overhead.

We illustrate that our proposed RWMH system yields significant performance improvement over the system of [15], which is a single-phase equivalent to our RWMH system. In the system of [15], ME is executed within the RDWT domain; however, only a single critically sampled DWT is predicted, and the ME is optimized to that

Table 6.2: Distortion averaged over all frames of the sequence.

	PSNR (dB)		
	Spatial Block	RDWT Block	RWMH
Football†	26.3	27.9	28.6
Susie	36.0	37.4	37.8
Mother & daughter	40.2	40.8	41.2
Coastguard	28.1	28.9	29.5

Rate is 0.25 bpp except †, which is 0.5 bpp.

single phase. Average PSNR figures for fixed bit rate are tabulated in Table 6.2, and frame-by-frame PSNR profiles for two sequences "Football" and "Susie" are shown in Figs. 6.4 and 6.5. In these results, "RDWT Block" and "Spatial Block" refers to the same methods specified in the previous section.

These results illustrate that multihypothesis prediction in the form of our RWMH system achieves at least a 0.4-dB gain over single-phase prediction. For sequences with complex motion, our RWMH system achieves even larger performance gains. For example, RWMH exhibits a gain of nearly 1 dB over the system of [15] for the "Football" sequence, and a gain of over 2 dB over the spatial-domain system.

The observed performance gain lies in the fact that RWMH extends the idea of MHMC into transform domain. Recognizing that different phases in RDWT coefficients view the motion from different perspectives, we treat each critically sampled DWT within the RDWT as a separate hypothesis prediction. An inverse RDWT operation implicitly combines the multiple predictions with no need for side information concerning prediction weights. Additionally, we use a hierarchical search to tailor the motion-vector field to individual phases. Substantial gains are obtained in comparison to an equivalent single-phase prediction.

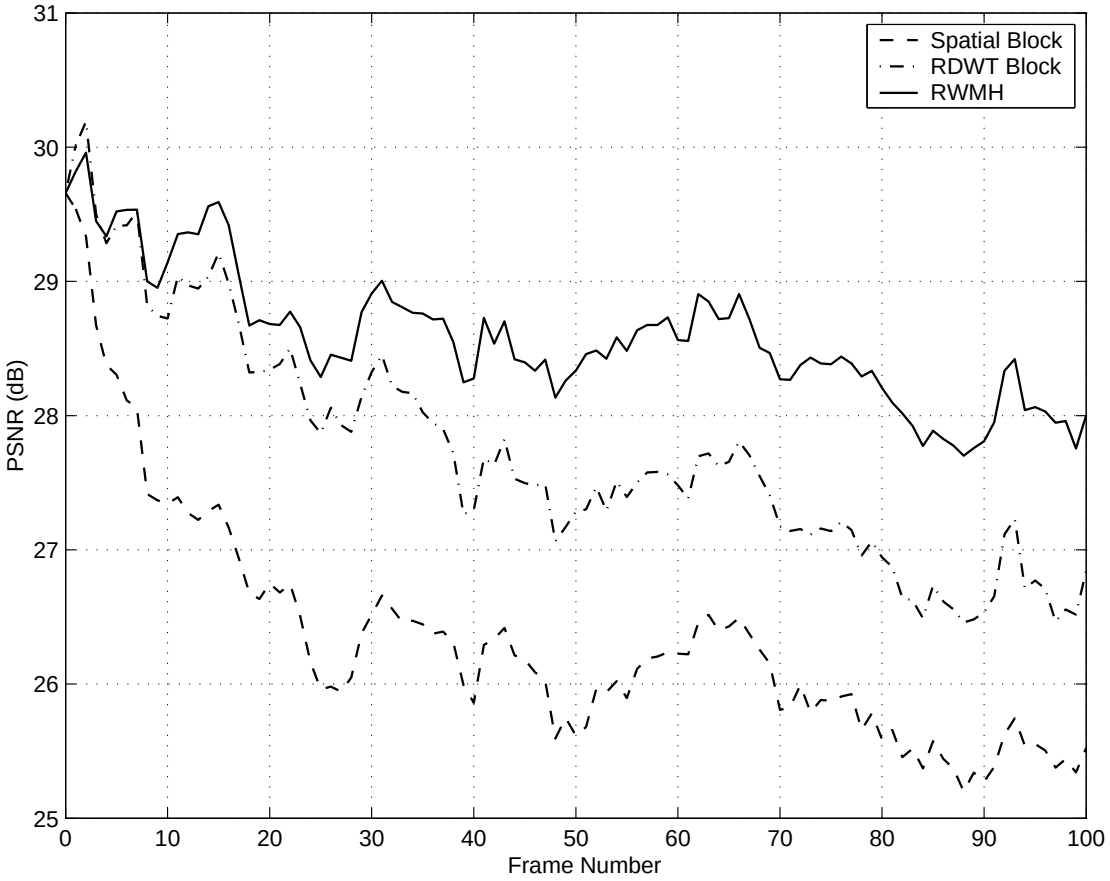


Figure 6.4: Comparison of RWMH to RDWT Block—frame-by-frame PSNR for "Football" at 0.5 bpp (1.3 Mbps).

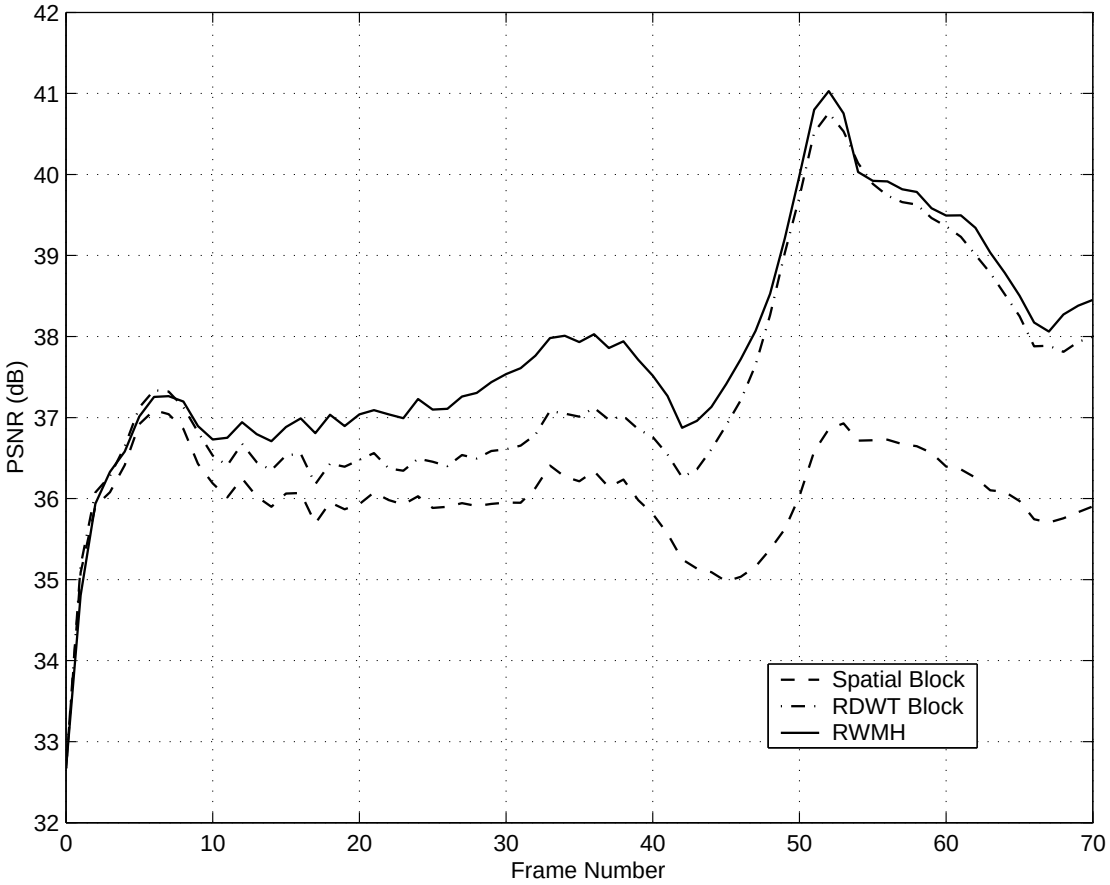


Figure 6.5: Comparison of RWMH to RDWT Block – frame-by-frame PSNR for "Susie" at 0.25 bpp (634 kbps).

Table 6.3: Distortion averaged over all frames of the sequence.

	PSNR (dB)		
	RWMH	RWMH-OBMC	RWMH-OBMC-1/4
Football†	28.6	29.2	29.9
Susie	37.8	38.9	39.8
Mother & daughter	41.2	42.3	43.9
Coastguard	29.5	30.1	31.1

Rate is 0.25 bpp except †, which is 0.5 bpp.

Further improvement can be obtained by combining phase-diversity multihypothesis as represented by RWMH with spatial-diversity multihypothesis in the form of OBMC and fractional-pixel ME/MC. In Table 6.3, we compare average PSNRs of RWMH with integer-pixel accuracy, RWMH coupled with OBMC with integer-pixel accuracy (RWMH-OBMC), and RWMH coupled with OBMC with quarter-pixel accuracy (RWMH-OBMC-1/4). The combination of spatial- and phase-diversity multihypothesis as represented by RWMH-OBMC-1/4 gains at least 0.7-dB over the other approaches for both low-motion sequences ("Susie") as well as high-motion sequences ("Football").

Frame-by-frame PSNR profiles for two sequences are shown in Figs. 6.6 and 6.7. In Fig. 6.8, we examine frame 76 of the "Football" sequence to compare the reconstructed images. We see that, while the addition of OBMC, which eliminates blocking artifacts resulting from the block-based search, produces increased performance over all frames of these sequences, the addition of quarter-pixel accuracy is most effective when motion is slow (e.g., the first 40 frames of "Susie" in Fig. 6.7).

The results indicate that adding both OBMC and fractional-pixel accuracy to RWMH produces significant performance gains. Additionally, we have found that both of these spatial-diversity multihypothesis techniques can be deployed within RDWT subbands in essentially the same form as their original spatial-domain implementations.

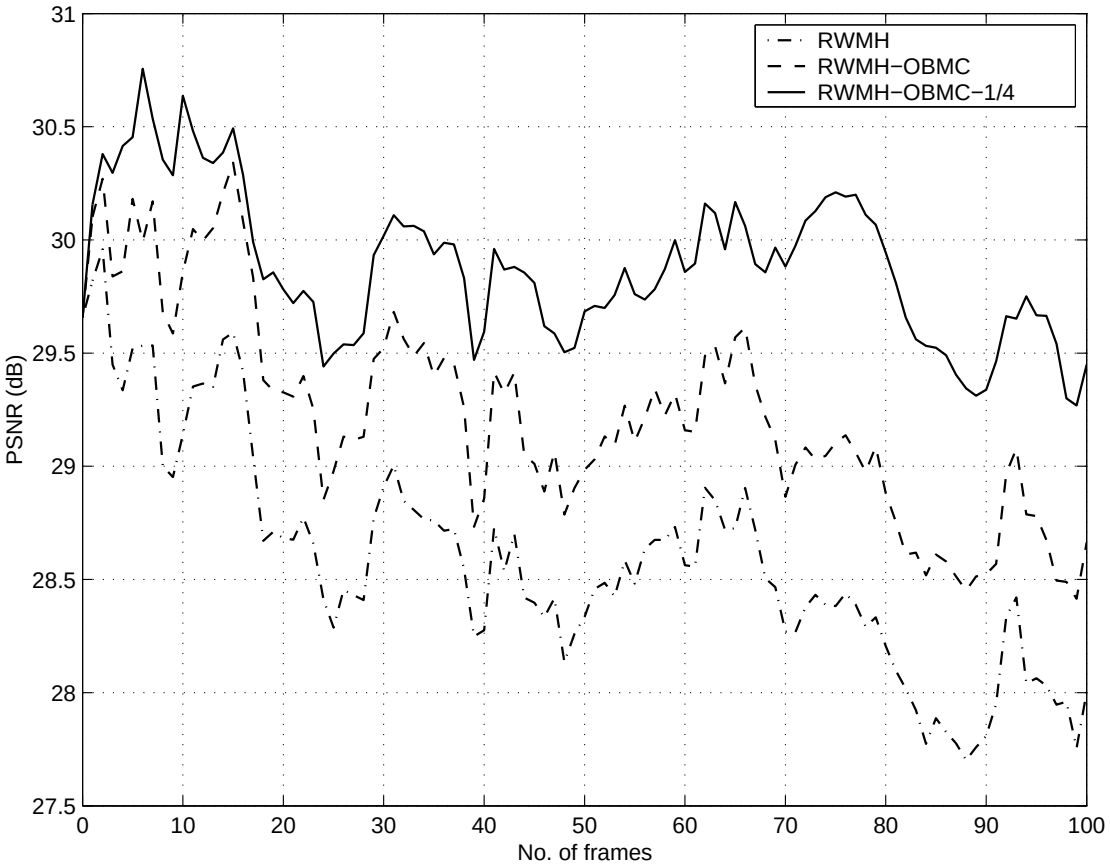


Figure 6.6: Comparison of RWMH to RWMH-OBMC—frame-by-frame PSNR for "Football" at 0.5 bpp (1.3 Mbps).

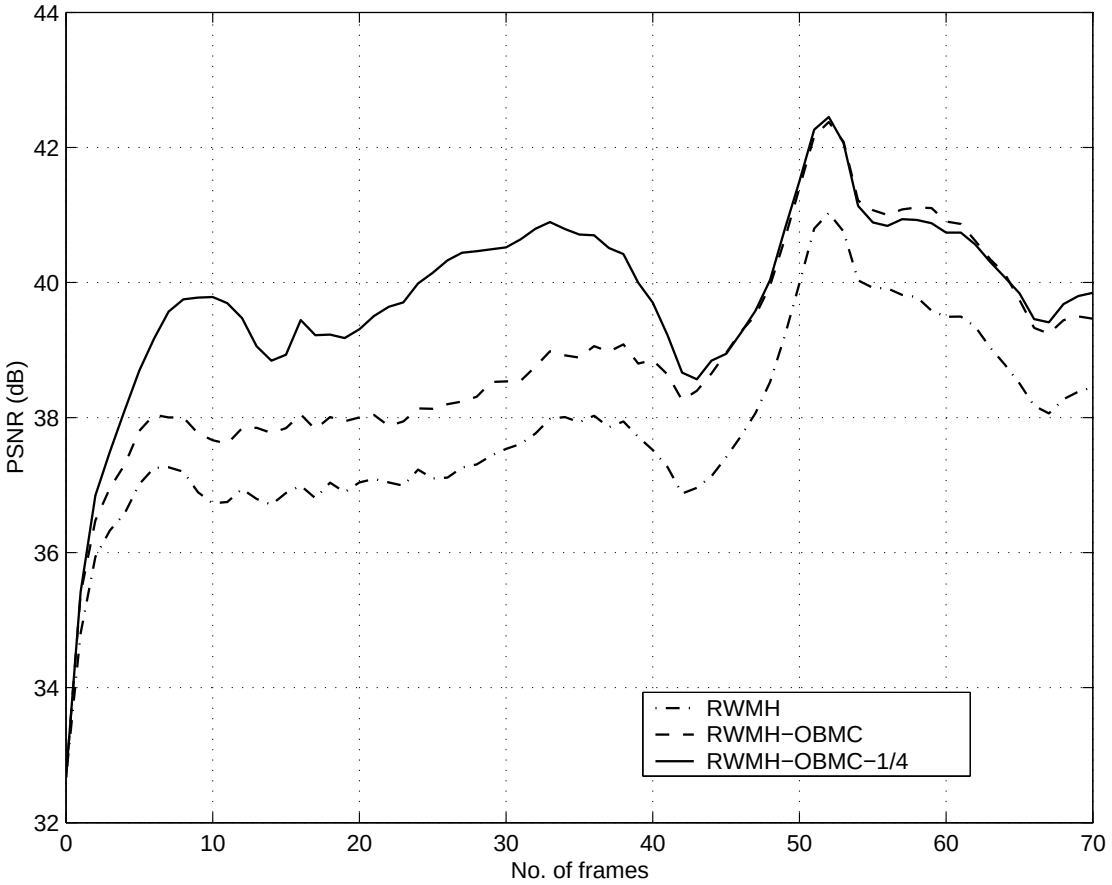


Figure 6.7: Comparison of RWMH to RWMH-OBMC—frame-by-frame PSNR for "Susie" at 0.25 bpp (634 kbps).

(a)

(b)

(c)

Figure 6.8: Original and reconstructed images for frame 76 of “Football” (cropped to show detail). (a) Original, (b) RWMH in integer-pixel, (c) RWMH-OBMC-1/4.

Table 6.4: Distortion averaged over all frames of the sequence.

	PSNR (dB)			
	Football†	Susie	Mother & Daughter	Coastguard
RWMH	28.6	37.8	41.2	29.5
RWMH-LT	29.1	38.7	42.8	30.1

Rate is 0.25 bpp except †, which is 0.5 bpp.

6.3 The RWMH-LT System

The results of the previous section indicate that RWMH operates complementary to other forms of multihypothesis. Specificially, we demonstrated gains for multihypothesis techniques employing spatial diversity. In this section, we show that RWMH also functions complementary to multihypothesis techniques employing temporal diversity, specifically long-term memory MC (LTMMC). To this end, we examine performance of the RWMH-LT system proposed in Sec. 5.5.

The average PSNRs of four sequences are shown in Table 6.4. Frame-by-frame PSNR profiles for two sequences “Football” and “Susie” are shown in Figs. 6.9 and 6.10. Since we are not considering spatial diversity with these results, ME/MC is performed with integer-pixel accuracy in both systems. We see that adding temporal diversity yields at least 0.5-dB gain over all sequences regardless as to whether the sequences have high or low motion activity. In the “Mother & Daughter” sequence, there is 1.6-dB gain. Consequently, we conclude that adding temporal-diversity multihypothesis to our RWMH system improves performance just as spatial-diversity does. This gain comes in spite of the fact that the motion-vector overhead of RWMH-LT is three times that of the RWMH system.

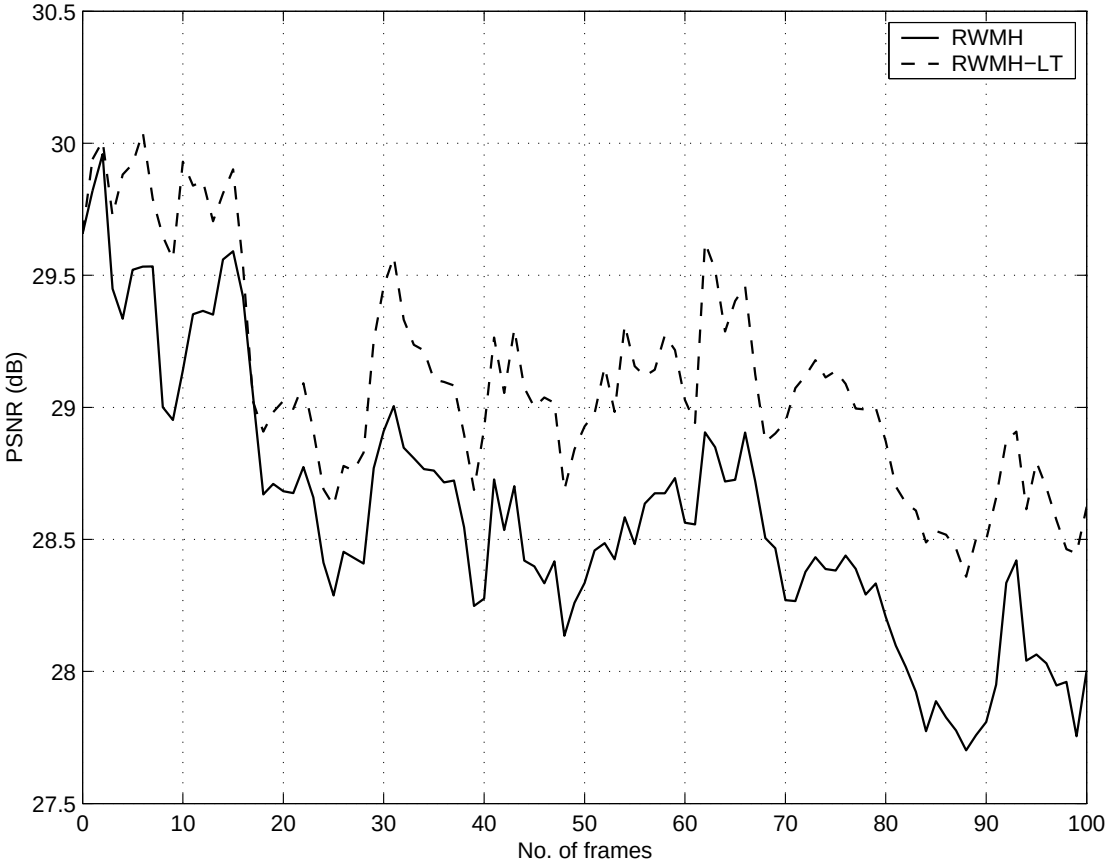


Figure 6.9: Comparison of RWMH to RWMH-LT—frame-by-frame PSNR for "Football" at 0.5 bpp (1.3 Mbps).

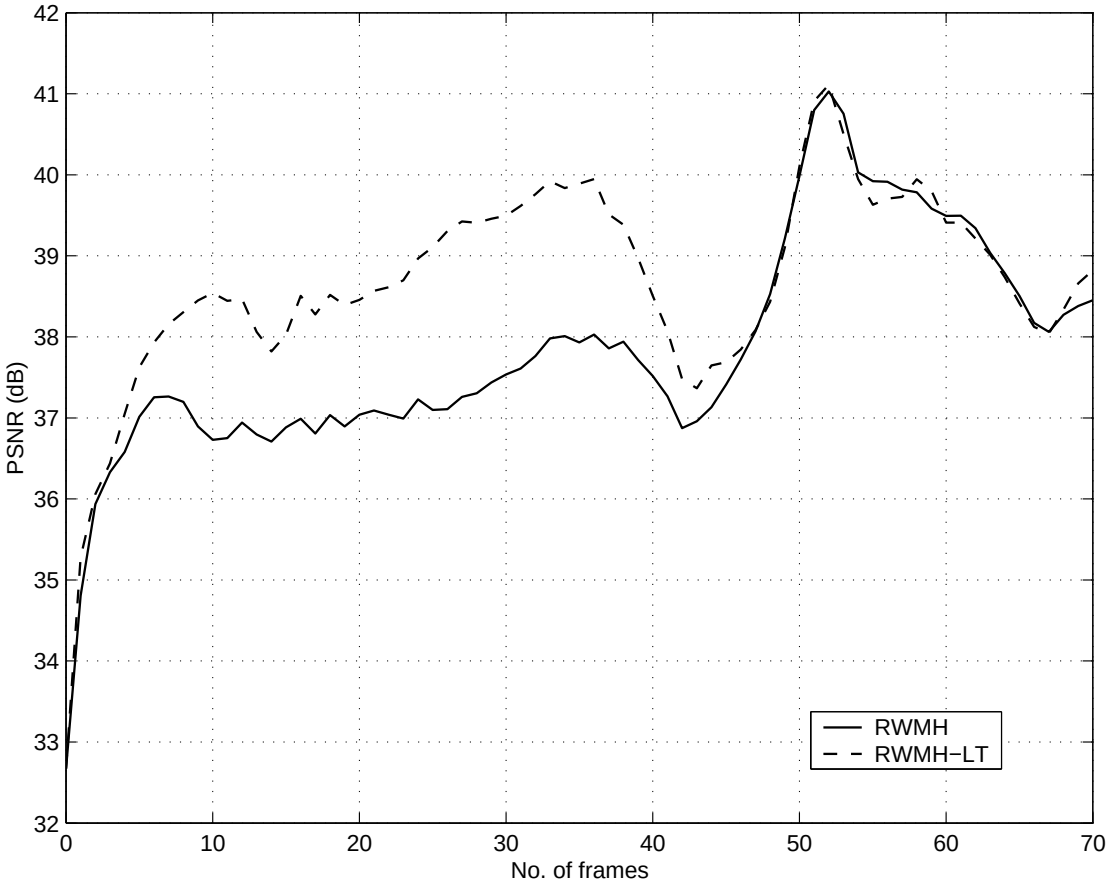


Figure 6.10: Comparison of RWMH to RWMH-LT—frame-by-frame PSNR for "Susie" at 0.25 bpp (634 kbps).

Table 6.5: Distortion averaged over all frames of the sequence.

	PSNR (dB)			
	Football†	Susie	Mother & Daughter	Coastguard
RWTM	28.3	37.8	41.7	28.7
RWTMMH	28.6	38.3	41.9	29.3

Rate is 0.25 bpp except †, which is 0.5 bpp.

6.4 The RWTMMH System

In this final section of this chapter, we demonstrate that the replacing of the traditional block-based geometry employed in our RWMH systems with the triangle-mesh structure developed for our RWTM system produces performance gain. Specifically, we compare the performance of the original RWMH system (for simplicity of discussion we do not include the spatial- and temporal-diversity refinements previously investigated) to the RWTMMH system proposed in Sec. 5.6.

The average PSNRs are shown in Table 6.5. Frame-by-frame PSNR profiles for two sequences are shown in Figs. 6.11 and 6.12. There is at least 0.2dB gain for RWTMMH over RWTM.

The results of this chapter have indicated that the RDWT can play a role in video-coding systems beyond just the introduction of shift invariance for ME/MC. Specifically, it can facilitate ME/MC geometries more general than traditional block structures as well as provide the basis for phase-diversity multihypothesis, all the while functioning complementary to a number of advanced video-coding techniques. In the next chapter, we make some concluding remarks concerning the work we have presented.

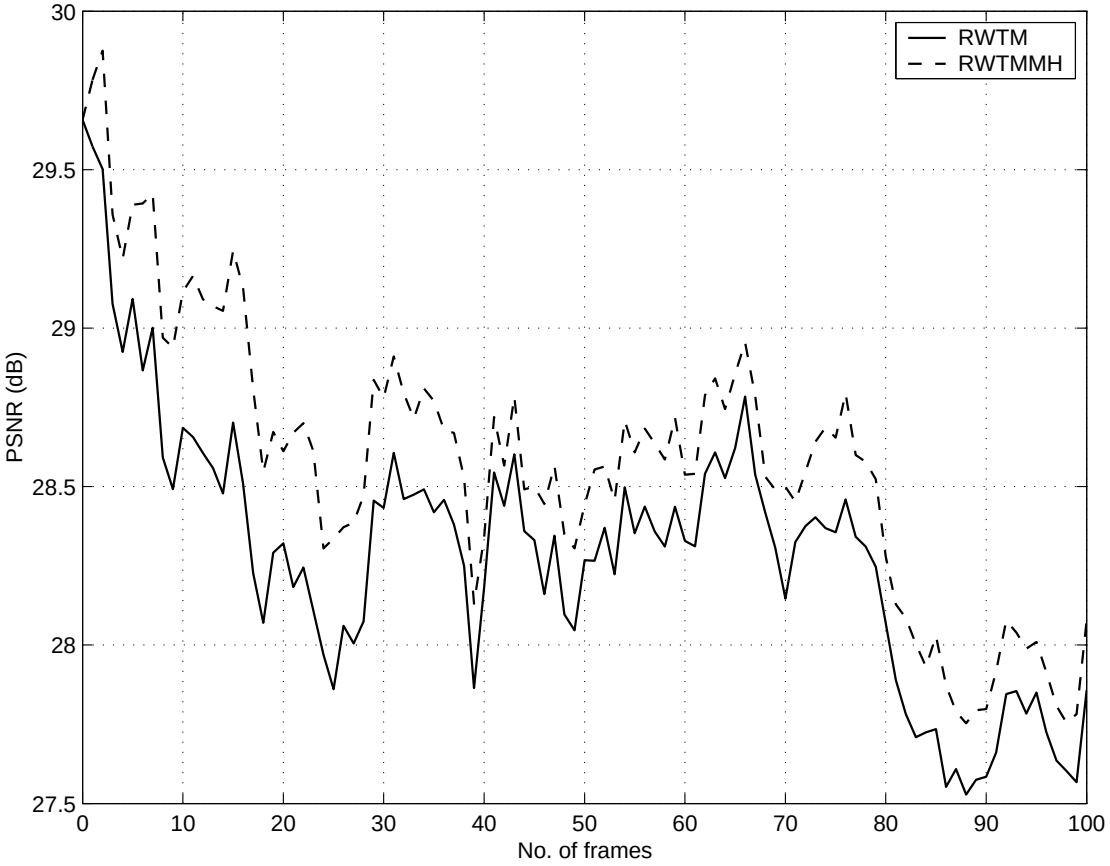


Figure 6.11: Comparison of RWTM and RWTMMH—frame-by-frame PSNR for "Football" at 0.5 bpp (1.3 Mbps).

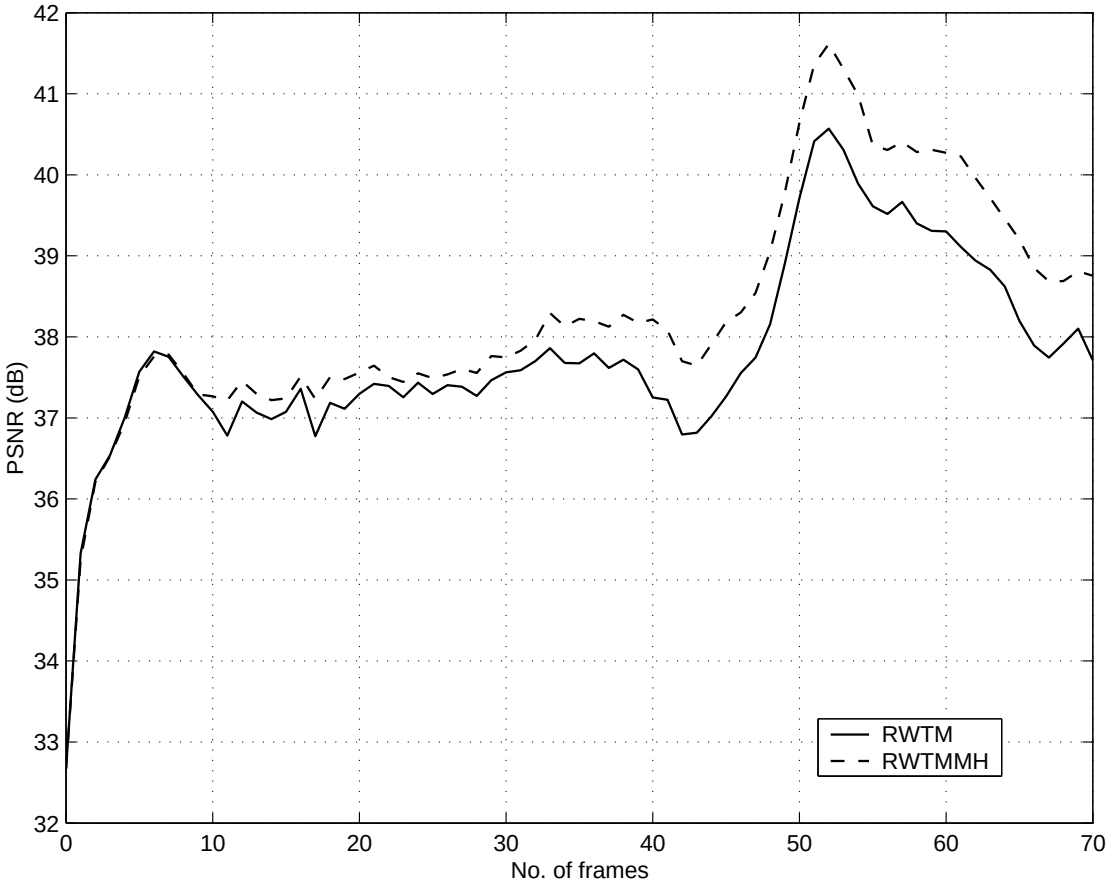


Figure 6.12: Comparison of RWTM and RWTMMH—frame-by-frame PSNR for "Susie" at 0.25 bpp (634 kbps).

CHAPTER VII
CONCLUSION

To summarize the work accomplished in this dissertation, we have built several systems which are each based on the idea of ME/MC in the domain of a redundant wavelet transform. As was demonstrated in [15] and in a number of prior investigations [16–18, 21, 23, 24], in the RDWT domain, the shift variance of the usual critically sampled DWT no longer poses a problem for the estimation of object motion. However, as we have demonstrated in this disserattion, the redundancy of RDWT can be exploited for ends other than just its mere shift invariance. Specifically, the RDWT can facilitate the deployment of an irregular triangle mesh instead of block-based ME/MC to eliminate of blocking artifacts as was done in our RWTM system introduced in Chap. IV. Additionally, it is possible to use the RDWT redundancy to enable multihypothesis prediction with phase-diversity to increase prediction accuracy as was done in the RWMH system introduced in Chap. V. In addition to phase-diversity, we can also implement spatial-diversity (e.g., OBMC and subpixel accuracy), and temporal-diversity (e.g., LTMMC), to our RWMH system to build a highly multihypothesis system such that each form of multihypothesis complements the others for significantly improved performance as was demonstrated in the results of Chap. V. Finally, we also were able to combine RWTM with RWMH to get a phase-diversity system with improved performance.

Modern video-compression systems are built upon a large collection of diverse techniques, all of which improve system performance in some fashion to various degrees. For example, it has been recognized that the significant performance improvement observed of the current H.263 Version 2 (H.263+) [6] standard results from

no one single coding element; rather, it is the accumulation of a large and diverse set of coding techniques that yield performance superior to prior systems [47]. The RDWT-based techniques considered in this dissertation are no different in this respect—each provides a significant, albeit incremental, gain in performance.

However there exists limitations to the combining of all these techniques. That is, not all techniques produce gains for all sequences over all frames. Rather, some techniques work well for, say, slowly moving scenes, while others work better for fast motion. Consequently, modern video-coding standards are typically composed of numerous coding modes such that individual coding techniques can be switched on or off as needed. The techniques we have proposed here should also be subject to such mode control—we may not, for example, use the phase-diversity of the RWMH technique on every frame of a sequence, but rather use it only when performance warrants. Coding standards (H.263 Version 2 [6]) already make such mode-control decisions for spatial-diversity and temporal-diversity multihypothesis approaches like sub-pixel accuracy and B-frames. Although beyond the scope of this dissertation, mode-control strategies for phase-diversity multihypothesis will be needed for any truly practical implementation of RWMH.

REFERENCES

[1] J. R. Jain and A. K. Jain, "Displacement measurement and its application in interframe image coding," *IEEE Transactions on Communications*, vol. 29, no. 12, pp. 1799–1808, December 1981.

[2] ITU-T, *Video Coding for for Audiovisual Services at* $p \times 64$ *kbit/s*, March 1993, ITU-T Recommendation H.261.

[3] ISO/IEC 11172-2, *Information Technology—Coding of Moving Pictures and Associated Audio for Digital Storage Media at up to about 1.5 Mbits/s*, 1993, MPEG-1 Video Coding Standard.

[4] ISO/IEC 13818-2, *Information Technology—Generic Coding of Moving Pictures and Associated Audio Information: Video*, 1995, MPEG-2 Video Coding Standard.

[5] ITU-T, *Video Coding for Low Bitrate Communication*, November 1995, ITU-T Recommendation H.263, Version 1.

[6] ITU-T, *Video Coding for Low Bitrate Communication*, January 1998, ITU-T Recommendation H.263, Version 2.

[7] ISO/IEC 14496-2, *Information Technology—Coding of Audio-Visual Objects—Part 2: Visual*, 1999, MPEG-4 Coding Standard.

[8] A. Said and W. A. Pearlman, "A new, fast, and efficient image codec based on set partitioning in hierarchical trees," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 6, no. 3, pp. 243–250, June 1996.

[9] F. Dufaux, F. Moscheni, and M. Shütz, "Motion compensated wavelet transform coding," in *Proceedings of the International Picture Coding Symposium*, Sacramento, CA, 1994.

[10] G. Van der Auwera, A. Munteanu, G. Lafruit, and J. Cornelis, "Video coding based on motion estimation in the wavelet detail images," in *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing*, Seattle, WA, May 1998, vol. 5, pp. 2801–2804.

[11] S. A. Martucci, I. Sodagar, T. Chiang, and Y.-Q. Zhang, "A zerotree wavelet video coder," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 7, no. 1, pp. 109–118, February 1997.

[12] Y.-Q. Zhang and S. Zafar, "Motion-compensated wavelet transform coding for color video compression," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 2, no. 3, pp. 285–296, September 1992.

[13] T. Naveen and J. W. Woods, "Motion compensated multiresolution transmission of high definition video," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 4, no. 1, pp. 29–41, February 1994.

[14] M. Antonini, M. Barlaud, P. Mathieu, and I. Daubechies, "Image coding using wavelet transform," *IEEE Transactions on Image Processing*, vol. 1, no. 2, pp. 205–220, April 1992.

[15] H.-W. Park and H.-S. Kim, "Motion estimation using low-band-shift method for wavelet-based moving-picture coding," *IEEE Transactions on Image Processing*, vol. 9, no. 4, pp. 577–587, April 2000.

[16] H. S. Kim and H. W. Park, "Wavelet-based moving-picture coding using shift-invariant motion estimation in wavelet domain," *Signal Processing: Image Communication*, vol. 16, no. 7, pp. 669–679, April 2001.

[17] X. Li, L. Kerofsky, and S. Lei, "All-phase motion compensated prediction in the wavelet domain for high performance video coding," in *Proceedings of the International Conference on Image Processing*, Thessaloniki, Greece, October 2001, vol. 2, pp. 538–541.

[18] X. Li and L. Kerofsky, "High-performance resolution-scalable video coding via all-phase motion-compensated prediction of wavelet coefficients," in *Visual Communications and Image Processing*, C.-C. J. Kuo, Ed. Proc. SPIE 4671, January 2002, pp. 1080–1090.

[19] X. Li and S. Lei, "Efficient motion field representation in the wavelet domain for video compression," in *Proceedings of the International Conference on Image Processing*, Rochester, NY, September 2002, vol. 3, pp. 257–260.

[20] G. Van der Auwera, A. Munteanu, P. Schelkens, and J. Cornelius, "Scalable wavelet video-coding with in-band prediction—The bottom-up overcomplete discrete wavelet transform," in *Proceedings of the International Conference on Image Processing*, Rochester, NY, September 2002, vol. 3, pp. 725–728.

[21] Y. Andreopoulos, A. Munteanu, G. Van der Auwera, P. Schelkens, and J. Cornelius, "Scalable wavelet video-coding with in-band prediction—Implementation and experimental results," in *Proceedings of the International Conference on Image Processing*, Rochester, NY, 2002, vol. 3, pp. 729–732.

[22] Y. Andreopoulos, A. Munteanu, G. Van der Auwera, P. Schelkens, and J. Cornelius, "A new method for complete-to-overcomplete discrete wavelet transforms," in *Proceedings of the International Conference on Digital Signal Processing*, Santorini, Greece, July 2002, pp. 501–504.

[23] Y. Andreopoulos, A. Munteanu, G. Van der Auwera, J. Barbarien, P. Schelkens, and J. Cornelius, "Wavelet-based fine granularity scalable video coding with in-band prediction," ISO/IEC JTC1/SC29/WG11, MPEG2002/M7906, Jeju Island, South Korea, March 2002.

[24] Y. Andreopoulos, A. Munteanu, G. Van der Auwera, P. Schelkens, and J. Cornelius, "Wavelet-based fully-scalable video coding with in-band prediction," in *Proceedings of the* 3rd *IEEE Benelux Signal Processing Symposium*, Leuven, Belgium, March 2002, pp. 217–220.

[25] S. Cui, Y. Wang, and J. E. Fowler, "Mesh-based motion estimation and compensation in the wavelet domain using a redundant transform," in *Proceedings of the International Conference on Image Processing*, Rochester, NY, September 2002, vol. 1, pp. 693–696.

[26] S. Cui, Y. Wang, and J. E. Fowler, "Motion estimation and compensation in the redundant-wavelet domain using triangle meshes," *IEEE Transactions on Circuits and Systems for Video Technology*, October 2002, submitted.

[27] N. Sebe, C. Lamba, and M. S. Lew, "An overcomplete discrete wavelet transform for video compression," in *Proceedings of the IEEE International Conference on Multimedia and Expo*, Lausanne, Switzerland, August 2002, vol. 2, pp. 541–644.

[28] S. Cui, Y. Wang, and J. E. Fowler, "Multihypothesis motion compensation in the redundant wavelet domain," in *Proceedings of the International Conference on Image Processing*, Barcelona, Spain, 2003, to appear.

[29] S. Cui, Y. Wang, and J. E. Fowler, "Motion compensation via redundant-wavelet multihypothesis," *IEEE Transactions on Circuits and Systems for Video Technology*, February 2003, submitted.

[30] P. Dutilleux, "An implementation of the "algorithme à trous" to compute the wavelet transform," in *Wavelets: Time-Frequency Methods and Phase Space*, J.-M. Combes, A. Grossman, and P. Tchamichian, Eds., pp. 298–304. Springer-Verlag, Berlin, Germany, 1989, Proceedings of the International Conference, Marseille, France, December 14–18, 1987.

[31] M. Holschneider, R. Kronland-Martinet, J. Morlet, and P. Tchamitchian, "A real-time algorithm for signal analysis with the help of the wavelet transform," in *Wavelets: Time-Frequency Methods and Phase Space*, J.-M. Combes,

A. Grossman, and P. Tchamichian, Eds., pp. 286–297. Springer-Verlag, Berlin, Germany, 1989, Proceedings of the International Conference, Marseille, France, December 14–18, 1987.

[32] M. J. Shensa, "The discrete wavelet transform: Wedding the à trous and Mallat algorithms," *IEEE Transactions on Signal Processing*, vol. 40, no. 10, pp. 2464–2482, October 1992.

[33] S. Mallat and S. Zhong, "Characterization of signals from multiscale edges," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 14, no. 7, pp. 710–732, July 1992.

[34] S. Mallat, *A Wavelet Tour of Signal Processing*, Academic Press, San Diego, CA, 1998.

[35] M. Eckert, D. Ruiz, J. I. Ronda, and N. Garcia, "Evaluation of DWT and DCT for irregular mesh-based motion compensation in predictive video coding," in *Visual Communications and Image Processing*, K. N. Ngan, T. Sikora, and M.-T. Sun, Eds. Proc. SPIE 4067, June 2000, pp. 447–456.

[36] Y. Xu, J. B. Weaver, D. Healy, Jr., and J. Lu, "Wavelet transform domain filters: A spatially selective noise filtration technique," *IEEE Transactions on Image Processing*, vol. 3, no. 6, pp. 747–758, November 1994.

[37] M. D. Berg, M. V. Kreveld, M. Overmars, and O. Schwarzkopf, *Computational Geometry: Algorithms and Applications*, Springer-Verlag, Berlin, 1997.

[38] Y. Altunbasak, A. M. Tekalp, and G. Bozdagi, "Two-dimensional object-based coding using a content-based mesh and affine motion parameterization," in *Proceedings of the International Conference on Image Processing*, Washington, DC, October 1995, vol. 2, pp. 394–397.

[39] G. J. Sullivan, "Multi-hypothesis motion compensation for low bit-rate video coding," in *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing*, Minneapolis, MN, April 1993, vol. 5, pp. 437–440.

[40] B. Girod, "Motion-compensating prediction with fractional-pel accuracy," *IEEE Transactions on Communications*, vol. 41, no. 4, pp. 604–612, April 1993.

[41] S. Nogaki and M. Ohta, "An overlapped block motion compensation for high quality motion picture coding," in *Proceedings of the IEEE International Symposium on Circuits and Systems*, San Diego, CA, May 1992, vol. 1, pp. 184–187.

[42] M. T. Orchard and G. J. Sullivan, "Overlapped block motion compensation: An estimation-theoretic approach," *IEEE Transactions on Image Processing*, vol. 3, no. 5, pp. 693–699, September 1994.

[43] T. Wiegand, X. Zhang, and B. Girod, "Long-term memory motion-compensated prediction," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 9, no. 1, pp. 70–84, February 1999.

[44] M. Flierl, T. Wiegand, and B. Girod, "Rate-constrained multihypothesis prediction for motion compensated video compression," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 12, no. 11, pp. 957–969, November 2002.

[45] M. Bierling, "Displacement estimation by hierarchical block matching," in *Visual Communications and Image Processing*, T. R. Hsing, Ed., Cambridge, MA, November 1988, Proc. SPIE 1001, pp. 942–951.

[46] J. E. Fowler, "QccPack: An open-source software library for quantization, compression, and coding," in *Applications of Digital Image Processing XXIII*, A. G. Tescher, Ed., San Diego, CA, August 2000, Proc. SPIE 4115, pp. 294–301.

[47] H. Schwarz and T. Wiegand, "The emerging JVT/H.26L video coding standard," in *Proceedings of the International Broadcasting Convention*, Amsterdam, Netherlands, September 2002.