Efficient and Robust Layered Video Coding

by

Michael David Gallant

B. A. Sc. (Electrical Engineering) University of Ottawa, 1995

A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF

THE REQUIREMENTS FOR THE DEGREE OF

Doctor of Philosophy

in

THE FACULTY OF GRADUATE STUDIES

Department of Electrical and Computer Engineering

We accept this thesis as conforming
to the required standard

The University of British Columbia

February 2001

In presenting this thesis in partial fulfilment of the requirements for an advanced degree at the University of British Columbia, I agree that the Library shall make it freely available for reference and study. I further agree that permission for extensive copying of this thesis for scholarly purposes may be granted by the head of my department or by his or her representatives. It is understood that copying or publication of this thesis for financial gain shall not be allowed without my written permission.

Department of Electrical & Computer Engineering

The University of British Columbia
Vancouver, Canada

Date 23/02/01

Abstract

Layered coding and transport has become an attractive method for enabling video communications over the current non-uniform and sub-optimal network infrastructure. In this dissertation, we present video encoding algorithms for efficient and robust layered video encoding and transport in error-free and error-prone networks.

In the first part of this dissertation, error-free layered video encoding is considered. We evaluate the effectiveness of key technical features of a layered approach to video encoding. We then determine an upper bound on the rate-distortion performance of a layered approach to video encoding. Finally, a general formulation for efficient error-free layered video encoding is presented, based on the concept of operational rate-distortion optimization. This algorithm is demonstrated to achieve significant improvement in rate-distortion performance.

In the second part, we address complexity issues of this algorithm. Our goal is to find good tradeoffs between rate-distortion performance and computational complexity. We first motivate the need to make simplifications to an

operational rate-distortion optimization framework. We then propose a model to control the operating mode of the layered video encoder. This model permits the encoder to compute *a priori* the rate-distortion optimized parameters such that a target bit rate can be achieved.

The third part considers layered video encoding and transport in lossy packet-switched networks. A complete coding and transport framework is developed, including a packetization scheme, decoder error concealment method, and prioritization mechanism. We then introduce the general formulation for an efficient and robust layered video encoding algorithm for error-prone environments. This algorithm is also based on the concept of operational rate-distortion optimization and can be viewed as a generalization of the algorithm introduced for error-free environments. The algorithm incorporates a statistical distortion measure that considers the channel conditions, error recovery capability of the channel codec and error concealment capability of the source decoder to optimize the video encoding mode selection. Then, for a given layered bitstream and given channel conditions, optimal channel protection code rates are determined. This framework is shown to produce substantial improvement in reconstructed video quality for a wide range of packet loss rates. Moreover, it is demonstrated to yield graceful degradation of reconstructed video quality with increasing packet loss rate.

Contents

List of Tables

List of Figures

List of Abbreviations

ACK	Acknowledgment
ARQ	Automatic Repeat Request
ATM	Asynchronous Transfer Mode
BER	Bit Error Rate
CBP	Coded Block Pattern
CBR	Constant Bit-Rate
CDF	Cumulative Distribution Function
CIF	Common Intermediate Format
CRC	Cyclic Redundancy Check
DCT	Discrete Cosine Transform
DPCM	Differential Pulse Code Modulation
DSL	Digital Subscriber Line
EC	Error Concealment
EEP	Equal Error Protection
EREC	Error Resilient Entropy Code
FEC	Forward Error Correction
FLC	Fixed Length Code

FPS	Frames per Second
GOB	Group of Blocks
HVS	Human Visual System
IDCT	Inverse Discrete Cosine Transform
IP	Internet Protocol
ISDN	Integrated Services Digital Network
ISO	International Standards Organization
ITU	International Telecommunication Union
JPEG	Joint Photographic Experts Group
KLT	Karhunen-Loeve Transform
LMS	Least Mean Squares
MAP	Maximum a Posteriori
MB	Macroblock
MC	Motion Compensation
MD	Multiple Description
MDC	Multiple Description Coding
MDS	Maximum Distance Separable
MPEG	Moving Picture Expert Group
MSE	Mean Squared Error
MTU	Maximum Transfer Unit
MV	Motion Vector
PLR	Packet Loss Rate
PSNR	Peak Signal-to-Noise Ratio

PSTN	Public Switched Telephone Network
QCIF	Quarter Common Intermediate Format
QP	Quantizer Parameter
QoS	Quality of Service
RD	Rate-Distortion
RFC	Request for Comments
RS	Reed-Solomon
RTCP	RTP Control Protocol
RTP	Real-Time Transport Protocol
RVLC	Reversible Variable Length Code
SAD	Sum of Absolute Difference
SNR	Signal-to-Noise Ratio
SSE	Sum of Squared Error
TCM	Trellis-Coded Modulation
TCP	Transmission Control Protocol
TEC	Temporal Error Concealment
TMN	Test Model Near-Term
UDP	User Datagram Protocol
UEP	Unequal Error Protection
VBR	Variable Bit-Rate
VLC	Variable Length Code
VRC	Video Redundancy Coding

Acknowledgements

This dissertation would not have been possible without the collaboration and support of many people. I would like to take this opportunity to acknowledge these people and express my gratitude.

First I want to thank my advisor, Dr. Faouzi Kossentini for inspiring my research activities. His guidance and encouragement throughout the course of my studies and his commitment to the overall quality of our research and publications, in particular this dissertation, are greatly appreciated. It was a privilege to work under his mentorship.

I would also like to express my gratitude to Dr. Son Vuong, the chair of my dissertation committee, as well as the committee members, Dr. Hussein Alnuweiri and Dr. Rabab Ward, the university examiners, Dr. Mabo Ito and Dr. Jim Little, and the external examiner, Dr. Ming-Ting Sun. Their time and efforts are greatly appreciated and their constructive comments helped improve the quality of this dissertation.

For sharing their friendship and technical expertise, I would like to thank Dr. Alen Docef and Dr. Stephan Wenger. I would also like to thank Dr. Victor Leung for our discussions on error control coding.

To my graduate school friends and colleagues, Sandor Abrecht, Michael Adams, Guy Côté, Simon Dimaio, Berna Erol, Keyvan Hashtrudi-Zaad, Ismaeil Ismaeil, Anthony Joch, Parvin Mousavi, Khanh Nguyen-Phi, Shahram Shirani, and Dave Tompkins, I am grateful and wish them all the best.

This work would not have been possible without the financial support of the Natural Sciences and Engineering Research Council of Canada, the British Columbia Advanced Systems Institute, the Association of Universities and Colleges of Canada, the University of British Columbia and Roger Communications Inc. More recently, the support and understanding of Marc Morin and PixStream Incorporated during the final stages of preparation of this dissertation have been very much appreciated.

Finally, and most importantly, I would like to express my thanks and love to my family. My parents, Michael and Elizabeth, deeply instilled the value of education in all their children. This work is due in no small part to their dedication, sacrifice and support. I am especially thankful to Sheri, who shared this adventure with me and whose constant love, understanding and patience has proven to be my main source of strength and motivation.

MICHAEL DAVID GALLANT

The University of British Columbia

February 2001

Chapter 1

Introduction

1.1 Introduction

The coding and transport of real-time media over the emerging integrated communication infrastructure has become an extremely active research area. Unfortunately, this infrastructure is both non-uniform and sub-optimal, comprised of a patchwork of transmission media characterized by widely varying bandwidth capabilities both for different links and for the same link at different time instances. Important scenarios, such as multi-point and multicast sessions, require communication between many parties connected through these vastly different links. Moreover, individual receivers usually have different capabilities.

Video is arguably the most demanding of real-time media in terms of coding and transport. If we consider a raw video sequence, at CIF resolution (which is only about 1/4 the television-size resolutions we are accustomed to

viewing) of 352 $\times$ 288 pixels, with an equal sampling ratio for each of the three luminance and chrominance components, eight bits per pixel and thirty frames per second, the required bandwidth would be approximately 75 Mbps. Obviously good compression is critical for communication to be viable and efficient.

The most successful video compression algorithms employ predictive coding in the form of motion compensation [1, 2]. This reduces temporal redundancies between successive images. However, when this motion information is lost to the decoder, a reconstruction error can occur. These errors can propagate temporally and spatially if the affected region is subsequently used for prediction during motion compensation. Furthermore, differential encoding is also employed within an image to reduce statistical redundancies. Loss of such information can cause additional spatial degradation throughout the affected image by producing incorrectly predicted parameters. Because of motion compensation, these errors also can propagate temporally and spatially. It is therefore critical that the communication system also be robust.

From Shannon's separation theorem [3], the task of efficient and robust communication system design can be greatly simplified. Typically, the source coder can be designed to minimize the distortion due to quantization errors while the channel coder can be designed to minimize the distortion due to transmission errors. However, Shannon's theorem is based on the assumption of infinite complexity in the source coder and infinite processing delay at the channel coder. These assumptions are not realistic for any practical coding

system. In fact, minimizing complexity and delay are usually specific design goals. As such, a joint design of the source and channel coders can yield better overall system performance.

Consequently, recent video coding standards, in particular H.263+ and MPEG-4, have included methods that facilitate joint source and channel coder design [4, 5]. One of the methods that is supported is layered coding. Layered coding produces a hierarchy of bitstreams, where the first or *base* layer is coded independently and subsequent layers are coded dependently. Each layer of the hierarchy can increase the frequency, spatial and temporal resolution over that of the previous layer. Layered coding permits graceful degradation of reconstruction quality under varying bandwidth and loss rates. Furthermore, layered coding has inherent error-resilience benefits, particularly when the base layer bitstream can be transported with higher priority, guaranteeing a basic quality of service, and the enhancement layer bitstreams can be transported with lower priorities, refining the quality of service. This approach is commonly referred to as layered coding with transport prioritization [6].

In this dissertation, we present lossy video encoding algorithms for robust and efficient layered video coding and transport in error-free and error-prone environments. We evaluate the effectiveness of key technical features of layered video encoding. We present a general formulation of a layered video encoding algorithm for error-free environments, based on the concept of operational rate-distortion optimization. We address complexity issues of this algorithm and propose a model to control the operating mode of the layered

video encoder while reducing complexity. We then consider layered video coding and transport in lossy packet-switched networks. A complete coding and transport framework is developed. We introduce the general formulation for a layered video encoding algorithm for error-prone environments. This algorithm is also based on the concept of operational rate-distortion optimization. It incorporates the effects of transmission errors via a probabilistic distortion measure. Then, for a given layered bitstream and channel conditions, optimal channel protection strengths are determined. These algorithms are shown to produce substantial improvement in reconstructed video quality for both error-free and error-prone layered video communications.

1.2 Outline of the Thesis

In this thesis we first provide the necessary background in Chapter 2. We review the most popular low bit rate video coding algorithms. We discuss in detail one particular approach, H.263 [1], as it is employed throughout this thesis as the framework for testing of the proposed algorithms. We then review layered video coding. We discuss operational rate-distortion optimization techniques, which can serve to optimize the performance of practical coding systems if judiciously applied. Finally, we discuss relevant techniques for robust video communications.

The efficiency of compression schemes arises from a sophisticated level of interaction among the many system parameters. The selection of one tuple from the set of permissible parameters constitutes a discrete optimization

problem, which can be solved using principles of operational rate-distortion optimization [7]. For error-free coding and transport, the task for the source coder is to choose, for each coding unit, the most efficient coded representation in a rate-distortion sense. In Chapter 3 we first evaluate the effectiveness of the parameters for layered video encoding. We then determine, experimentally, an upper bound on the rate-distortion performance for layered video encoding. Finally, a general formulation for a layered video encoding algorithm is presented, based on the principles of operational rate-distortion optimization. This algorithm is demonstrated to achieve significant improvement in rate-distortion performance.

In Chapter 4, we address complexity issues of this algorithm. Our goal is to find good tradeoffs between rate-distortion performance and computational complexity. We first motivate the need to make simplifications to the operational rate-distortion optimization framework. Several simplifications are then proposed and evaluated. We modify the algorithm to select the locally optimal solution for each coding unit, instead of solving for a globally optimal solution. To select the parameter that controls the encoder's rate-distortion tradeoffs, we propose a model to control the operating mode of the layered video encoder. This model permits the encoder to compute *a priori* the rate-distortion optimized parameters such that a target bit rate can be achieved.

In Chapter 5, we consider layered video encoding and transport in lossy packet-switched networks. A complete layered encoding and transport framework is developed, including a packetization scheme, decoder error conceal-

ment method, and prioritization mechanism. We then introduce the general formulation for a layered video encoding algorithm for error-prone environments. This algorithm is also based on the concept of operational rate-distortion optimization and can be viewed as a generalization of the algorithm introduced for error-free environments. The algorithm incorporates a statistical distortion measure that considers the channel conditions, error recovery capability of the channel codec and error concealment capability of the source decoder to optimize the video encoding mode selection. Then, for a given layered bitstream and given channel conditions, optimal channel protection code rates are determined. This framework is shown to achieve substantial improvement in reconstructed video quality for a wide range of packet loss rates. Moreover, it is demonstrated to yield graceful degradation of reconstructed video quality with increasing packet loss rate.

Chapter 2

Background

In this chapter, we review the most popular low bit rate video encoding algorithms. We discuss one in detail, H.263 [1], as it is employed throughout this thesis as the framework for testing of the proposed layered video coding algorithms. We then review layered video coding. We discuss operational rate-distortion optimization techniques within the context of video coding [7]. Finally, we discuss the most popular techniques for robust video communications [6].

2.1 Video Coding

For good compression, the source model must efficiently capture the main characteristics of the data source with a reasonable level of complexity. Rather than employing a single complex model, the traditional approach to achieving this goal is to employ a number of simpler models [8]. For video

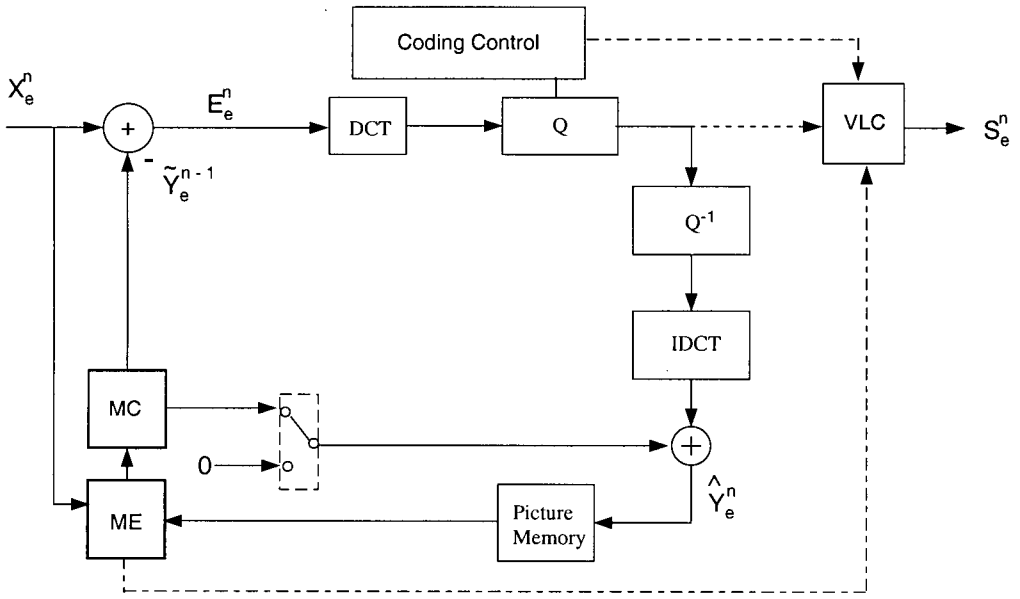


Figure 2.1: Block diagram for single-layer hybrid motion compensated, discrete-cosine transform video encoder.

coding, the dominant models combine a motion model with a transform coding model.

A wide range of motion models have been investigated. The most common model is a block-based translational motion model. Variations in block size can improve the performance of such a model. Another variation is multi-hypothesis prediction [9], wherein several prediction signals are superimposed. Examples of multi-hypothesis prediction include sub-pixel accurate prediction [10, 11], bi-directionally predicted frames [12], and overlapped block motion compensation [13]. Also, affine models, which use higher order representations of the motion field, allow for the representation of rotation, change of scale and shear, in addition to translation [14].

The transform coding model employs a linear transform that decomposes the data into frequency coefficients that can then be quantized. The transform coder compacts the signal energy and decorrelates the signal. Popular transform coders include discrete cosine transform (DCT) based coders [15] and subband based coders [16].

The source model employed in this thesis consists of a block based translational motion model, with blocks of 16×16 (referred to as macroblocks) and 8×8 (referred to as blocks) pixels, and a DCT-based transform model, with a block of 8×8 pixels. This is often termed a hybrid motion compensated DCT framework (MC-DCT). To date, this is the model employed by the most popular and successful video coding algorithms.

A generalized block diagram of a typical MC-DCT based video encoder is shown in Figure 2.1. The main components of this diagram are discussed next.

2.1.1 Prediction Types

The basic statistical property upon which video coding techniques rely is inter-pixel correlation, including the assumption of simple correlated translational motion between consecutive images. Specifically, it is assumed that the magnitude of a particular image pixel can be predicted from nearby pixels within the same image (spatial redundancy) using *intra mode* techniques or from pixels of a nearby image (temporal redundancy) using *inter mode* techniques. In some circumstances, e.g. during scene changes, the temporal correlation be-

tween pixels in nearby images is small or even vanishes and the video scene then resembles a collection of uncorrelated still images. In this case *intra mode* techniques are appropriate to exploit spatial correlation. However, if the correlation between pixels in nearby images is high, i.e. in cases where two consecutive images have similar or identical content, *inter mode* techniques (also referred to as differential pulse code modulation (DPCM) or motion compensation) are appropriate to exploit temporal correlation. In hybrid MC-DCT video coding schemes, a signal-adaptive combination of temporal prediction followed by spatial prediction is used. Thus, we can identify three basic types of coding for a given image region:

- *inter mode*: Motion compensated prediction from the previous image is used. The macroblock prediction type, the macroblock address and, if required, the motion vector, the DCT coefficients and quantization step size are transmitted. Note that the motion model employed in this thesis allows one or four motion vectors to be transmitted per macroblock.

- *skipped mode*: Prediction from the previous image with a zero motion vector. No information about the macroblock is coded or transmitted to the receiver. This is basically a special case of the *inter mode.*

- *intra mode*: No prediction is made from the previous image. Only the macroblock type, the macroblock address and the DCT coefficients and, if necessary, quantization step size are transmitted to the receiver.

mation is represented by displacement vectors or motion vectors. Due to the block-based motion representation, many algorithms employ block-matching techniques, where the motion vector is obtained by minimizing a cost function measuring the mismatch between a candidate and the current macroblock. Although any cost function can be used, the most widely-used choice is the sum of absolute difference (SAD) defined as

$$SAD = \sum_{i=1}^{N} \sum_{j=1}^{N} \mid B_{i,j} - B_{i-u,j-v} \mid, \quad N = 16 \tag{2.1}$$

Here $B_{i,j}$ represents a macroblock from the current image, and $B_{i-u,j-v}$ represents a candidate macroblock from a reference image at the spatial location (i, j) displaced by the vector (u, v). Note that the motion model used in this thesis also permits the use of four motion vectors per macroblock, in which case for (2.1) $N = 8$ and B represents a block as opposed to a macroblock.

To find the best matching macroblock producing the minimum mismatch error we need to calculate the SAD at several locations within a search window, shown in Figure 2.2. The simplest, but the most compute-intensive search method, known as the full search or exhaustive search, evaluates the SAD at every possible pixel location in the search area. To lower the computational complexity, several fast-search algorithms with a reduced number of search points have been proposed [17].

The picture memory in Figure 2.1 performs the storage of one or more previously reconstructed images. The ME block performs motion estimation, for the image to be encoded based on the previous reconstructed images that have been stored in the picture memory. The MC block builds a motion

compensated prediction of the current image using the estimates determined from the motion estimation stage.

2.1.3 Transformation

The purpose of transform coding is to decorrelate the image region content and compact energy into as few coefficients as possible, while preserving the energy of the block. For this purpose the optimal transform is the Karhunen-Loeve transform (KLT) [18]. However, the problem with the KLT is that it is signal dependent, as it depends on the autocovariance matrix. Furthermore, it is computationally complex, i.e. there exist no fast algorithms to compute the KLT. Thus, if it was used in practical communications applications, the transform would first have to be re-computed for the non-stationary data. Then, the new transform would have to transmitted to the receiver. Due to these complications, various fast approximations to the KLT have been proposed. The most successful of these is the DCT [15], originally developed [19] to approximate the KLT for a first-order Gauss-Markov process with a large positive correlation coefficient ρ ($\rho \rightarrow 1$). A Gauss-Markov process is described by the recursion

$$x(t) = \rho x(t-1) + n(t), \tag{2.2}$$

where $\rho \, \epsilon \, (-1, 1)$ and $r(t)$ is an independent identically distributed sequence of Gaussian random variables. While image and video data, as well as prediction error data, are not necessarily first-order Gauss-Markov, the DCT is still a good approximation to the KLT, and is widely employed in image and video

compression. There exist many fast algorithms for the DCT [15, 20, 21].

In the case of image/video coding, the DCT is typically applied to a two dimensional block of pixel data. For the MC-DCT framework employed in this thesis, we employ 8x8 blocks of pixels. The linear, separable and unitary forward two dimensional 8x8 DCT is defined as

$$C_{m,n} = \alpha(m)\beta(n)\sum_{i=1}^{N}\sum_{j=1}^{N} B_{i,j} \cos(\frac{\pi(2i+1)m}{2N})\cos(\frac{\pi(2j+1)n}{2N}),$$

for

$$0 \leq m,n \leq N-1, \quad N = 8,$$

where

$$\alpha(0) = \beta(0) = \sqrt{\frac{1}{N}}, \quad \alpha(m) = \beta(n) = \sqrt{\frac{2}{N}}, 1 \leq m,n \leq N-1.$$

Here, $B_{i,j}$ denotes the pixel block and $C_{m,n}$ denotes the transform coefficients. Note, that the transformation is reversible. The original 8x8 block of pixels can be reconstructed using a linear and separable inverse DCT:

$$B_{i,j} = \sum_{m=1}^{N}\sum_{n=1}^{N} C_{m,n}\alpha(m)\cos(\frac{\pi(2m+1)i}{2N})\beta(n)\cos(\frac{\pi(2n+1)j}{2N}),$$

for

$$0 \leq i,j \leq N-1, \quad N = 8$$

Energy compaction is manifested in the concentration of the most significant DCT coefficients around the low frequencies, or upper left corner. The significance of the coefficients decays with increased distance from the DC component, or upper-leftmost coefficient.

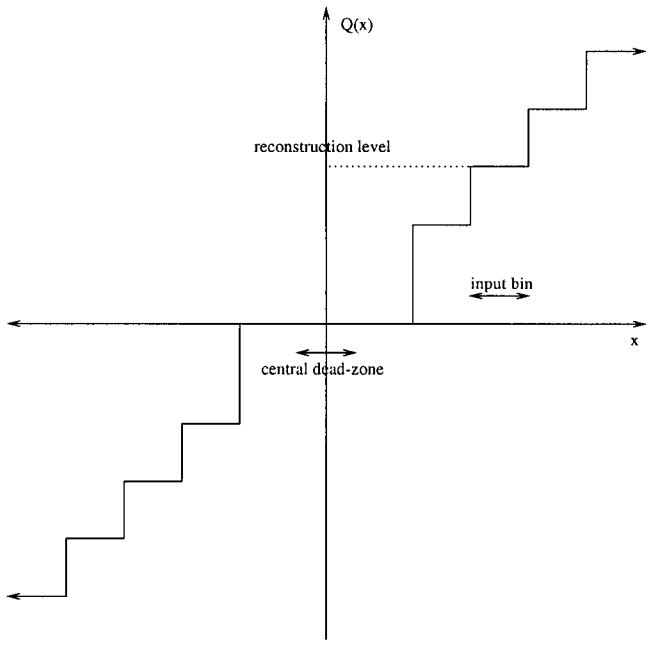


Figure 2.3: Scalar quantizer with central dead-zone.

The DCT and inverse DCT (IDCT) in Figure 2.1 perform the transformation and inverse transformation of *intra mode* or *inter mode* prediction error macroblocks.

2.1.4 Scalar Quantization

The human viewer is more sensitive to reconstruction errors related to low spatial frequencies than to high spatial frequencies [22]. Slow linear changes in intensity or color (low frequency information) are important to the eye. Quick, high frequency changes (noisy pixels, random pixels, edges, intensities above a certain level, etc.) cannot be seen and may be discarded. Quantization is therefore one source of loss in video coding and transport.

For every element position in the DCT output matrix, a corresponding quantization value is calculated by the following method.

$$C^q_{m,n} = \frac{C_{m,n} - \delta}{Q_{m,n}}, \quad 0 \leq m, n \leq N-1, \quad N = 8.$$

where $C_{m,n}$ represents the 8x8 DCT matrix of DCT coefficients, δ represents the quantizer central dead-zone and $Q_{m,n}$ is the 8x8 quantization matrix. This is illustrated in Figure 2.3. The result is then rounded to the nearest integer value. The net effect is a reduced variance between quantized coefficients as compared to the DCT coefficients, as well as a reduction of the number of non-zero coefficients.

The quantizer Q and inverse quantizer Q^{-1} in Figure 2.1 perform the quantization and inverse quantization of transform and quantized transform coefficients.

2.1.5 Entropy Coding

Prior to entropy coding, the quantized DCT coefficients are arranged into a one-dimensional array by scanning them in a zig-zag order. This re-arrangement places the DC coefficient first in the array and the remaining AC coefficients are ordered roughly from low to high frequency. This scan pattern is illustrated in Figure 2.4. The rearranged array is coded into a sequence of the run-length codes (RLC). The run is defined as the distance between two non-zero coefficients in the array. The level is the non-zero value immediately following a sequence of zeros. This coding method produces a compact representation of the 8x8 DCT coefficients, as a large number of the

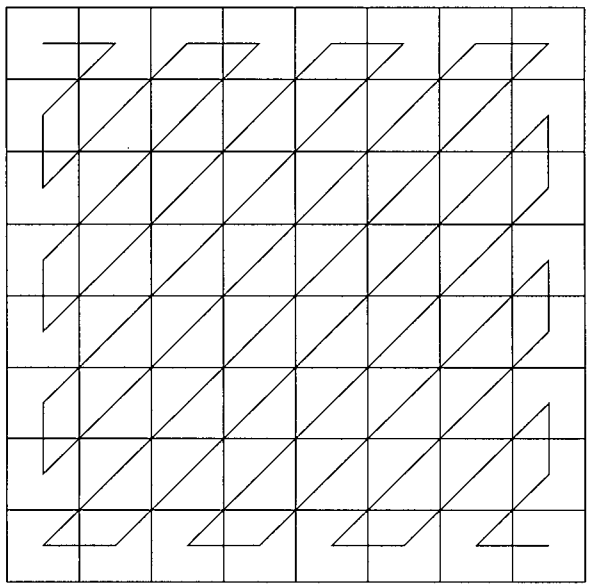


Figure 2.4: Zig-zag scan pattern to reorder DCT coefficients from low to high frequencies.

coefficients are expected to have been quantized to zero and the reordering has (ideally) resulted in the grouping of these zero values consecutively.

The run-level pairs (and other relevant information about the macroblock, such as motion vectors and prediction types) are then entropy coded. This step achieves compression by employing lossless techniques to compact the quantized coefficients based on statistical characteristics of the RLC and either Huffman or arithmetic coding. Entropy coding is performed by the variable length coding (VLC) block in Figure 2.1.

2.1.6 Buffer and Rate Control

Quantization of the source signal provides a constant bit rate. However, the use of an entropy coder following quantization results in a variable bit rate. A video buffer is essential to absorb variations in the instantaneous rate of the encoded signal. The quantization step size can then be adjusted for each macroblock within an image to achieve a given target bit rate and to avoid buffer overflow and underflow. This enables a high degree of flexibility in the bit allocation scheme.

A rate control algorithm at the encoder adjusts the quantizer step size depending on the video content and activity to ensure that the video buffers will never overflow while at the same time targeting to keep the buffers as full as possible to maximize image quality. In theory, overflow of buffers can always be avoided by using a large enough video buffer. However, besides the undesirable implementation costs of large buffers, there may be additional disadvantages for applications requiring low end-to-end delay. If the coded bit stream is smoothed using a video buffer to generate a constant bit rate output, a delay is introduced between the encoding process and the time the video can be reconstructed at the decoder. Usually a larger buffer entails a longer delay.

2.2 H.263 Video Coding

In this section, we discuss the ITU-T H.263 video coding algorithms in further detail as they are used as a framework to test the algorithms proposed in this

thesis. Although its coding structure is based on that of H.261 [23], H.263 provides better picture quality at low bit rates at the cost of some additional complexity. It also includes four optional modes, aimed at improving compression performance. H.263 version 2, or H.263+, is an extension of H.263. H.263+ provides twelve new optional modes to H.263. Note that while we maintain the distinction between H.263 and H.263+ in this section, we will use H.263 exclusively throughout the remainder of the thesis to refer to H.263 version 2, unless it is necessary to make the distinction.

2.2.1 H.263 Version 1

The block diagram in Figure 2.1 is representative of an H.263 baseline encoder. Motion compensated prediction first reduces temporal redundancies. DCT coding of the prediction error block then reduces spatial redundancies. Finally, VLC coding reduces statistical redundancies. H.263 supports five standardized image formats. The luminance component of the image is sampled at full resolution while the chrominance components, Cb and Cr, are downsampled by two in both the horizontal and vertical directions. The picture structure is shown in Figure 2.5 for the quarter common intermediate format (QCIF) resolution, 176×144 pixels. Each image in the input video sequence is divided into macroblocks, consisting of four luminance blocks of 8 pixels by 8 lines followed by one Cb block and one Cr block, each consisting of 8 pixels by 8 lines. A group of blocks (GOB) is defined as an integer number of macroblock rows, a number that is dependent on image resolution. For example, a GOB

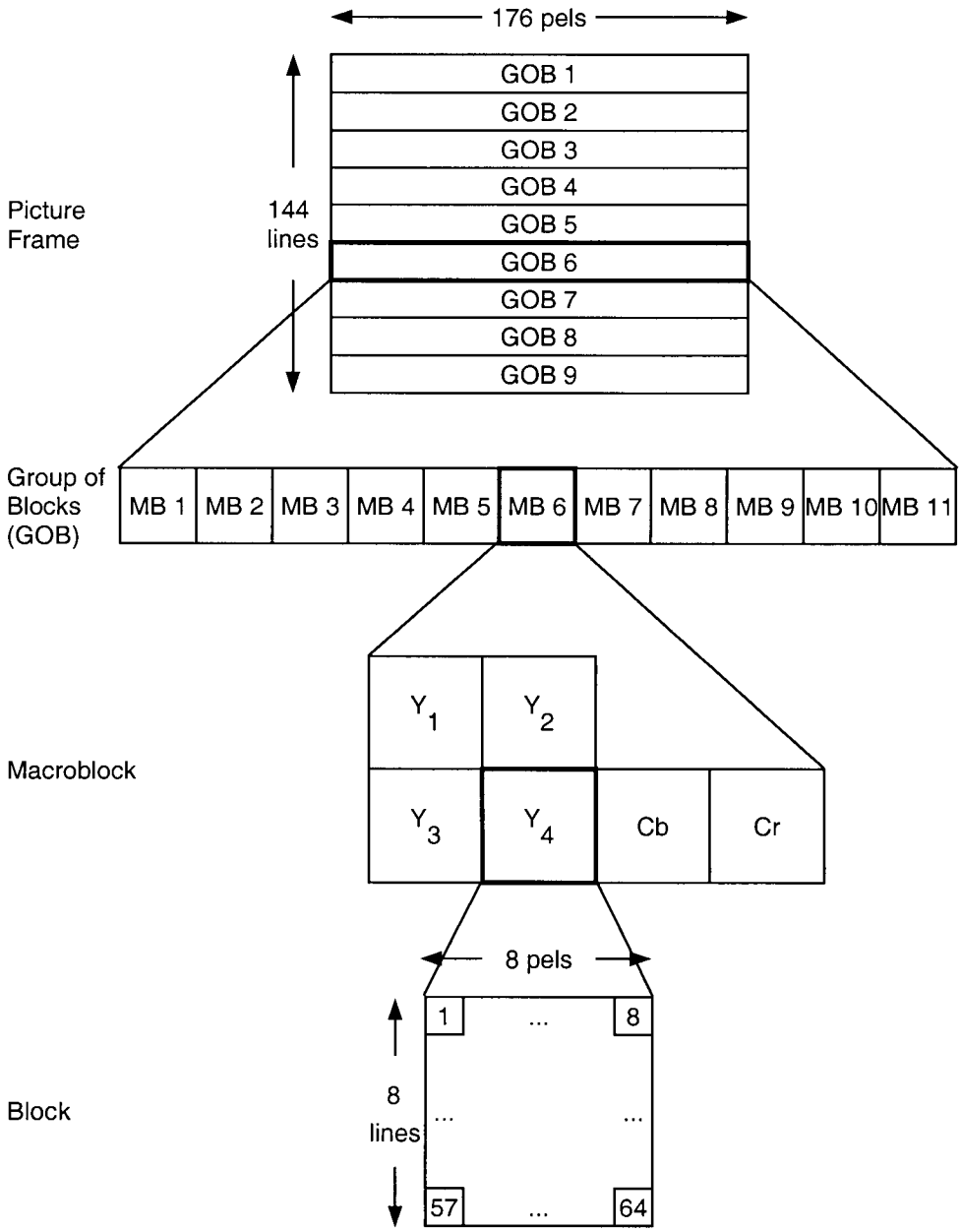


Figure 2.5: H.263 picture structure at QCIF resolution.

consists of a single macroblock row at QCIF resolution.

H.263 supports motion compensated prediction as described above. Recall that, in the *inter mode*, only the prediction error blocks need be encoded. If motion compensated prediction is not employed, the block is coded in the *intra mode*. As stated previously, how to choose an appropriate coding mode for a particular block is one of the questions that this thesis attempts to answer, for layered video coding in error-free and error-prone environments.

Optional Modes

In addition to the core encoding and decoding algorithms described above, H.263 includes four negotiable advanced coding modes as annexes to the standard: unrestricted motion vector mode (annex D), advanced prediction mode (annex F), PB-frames mode (annex G) and syntax-based arithmetic coding mode (annex E). The first two modes are used to improve motion compensated prediction. The PB-frames mode improves temporal resolution with little bit rate increase. When the syntax-based arithmetic coding mode is enabled, arithmetic coding replaces the default Huffman VLC coding. These optional modes allow developers to trade off between compression performance and complexity. We next provide a brief description of annexes D and F as they improve compression performance and are widely used. Consequently, they have been incorporated into the algorithms proposed in this thesis. A more detailed description of all modes can be found in [24] and [25].

Unrestricted Motion Vector mode (annex D) In baseline H.263, motion vectors can only reference pixels that are within the picture area. Because of this, macroblocks at the border of a picture may not be well predicted. When the unrestricted motion vector mode is used, motion vectors can take on values in the extended range of [-31.5, 31.5] pixels instead of [-16, 15.5] pixels and are allowed to point outside the picture boundaries. The longer motion vectors improve coding efficiency for larger picture formats, i.e. 4CIF or 16CIF. Moreover, by allowing motions vectors to point outside the picture, a significant gain is achieved if there is movement along picture edges. This is especially useful in the case of camera movement or background movement.

Advanced Prediction mode (annex F) This mode allows for the use of four motion vectors per macroblock, one for each of the four 8×8 luminance blocks. Furthermore, overlapped block motion compensation [13] is used for the luminance macroblocks, and motion vectors are allowed to point outside the picture as in the unrestricted motion vector mode. Use of this mode improves *inter mode* prediction and yields a significant improvement in subjective picture quality for the same bit rate by reducing blocking artifacts.

2.2.2 H.263 Version 2

The objective of H.263+ is to broaden the range of applications and to improve compression efficiency over H.263 version 1. H.263+ is backwards compatible with H.263. Not only is this critical due to the large number of video applications currently using the H.263 standard, but it is also required by ITU-T

rules.

H.263+ offers many improvements over H.263. It allows the use of a wide range of custom source formats, as opposed to H.263, wherein only five video source formats defining picture size, picture shape and clock frequency can be used. This added flexibility opens H.263+ to a broader range of video scenes and applications, such as wide format pictures, re-sizeable computer windows and higher refresh rates. Moreover, picture size, aspect ratio and clock frequency can be specified as part of the H.263+ bit stream. Another major improvement of H.263+ over H.263 is scalability, which is discussed in detail in Section 2.3. Furthermore, there are modes designed to improve error resilience and compression efficiency over H.263. This rich set of features makes H.263+ a natural choice as the framework within which to test the algorithms proposed in this thesis.

Optional Modes

Next, we describe several of the twelve new optional coding modes of H.263+[1], as they are incorporated into the algorithms proposed in this thesis.

Unrestricted Motion Vector mode (annex D) The definition of the unrestricted motion vector mode in H.263+ is different from that of H.263. When this mode is employed within an H.263+ framework, new reversible VLCs (RVLCs) [26] are used for encoding the difference motion vectors. These codes are single valued, as opposed to the earlier H.263 VLCs which were

[1] We defer our discussion of the H.263+ optional mode for layered coding to Section 2.3.

Picture width	Horizontal motion vector range	Picture height	Vertical motion vector range
4, ..., 352	[-32,31.5]	4, ..., 288	[-32, 31.5]
356, ..., 704	[-64,63.5]	292, ..., 576	[-64, 63.5]
708, ..., 1408	[-128,127.5]	292, ..., 576	[-64, 63.5]
1412, ..., 2048	[-256,255.5]	580, ..., 1152	[-128, 127.5]

Table 2.1: Motion vector range in H.263+ unrestricted motion vector range mode.

double valued. The double valued codes were not popular due to limitations in their extendibility and also to their high implementation cost. Reversible VLCs are easy to implement, as a simple state machine can be used to generate and decode them.

More importantly, reversible VLCs can be used to increase resilience to channel errors. The idea behind RVLCs is that decoding can be performed by processing the received motion vector part of the bit stream in the forward and reverse directions. If an error is detected while decoding in the forward direction, motion vector data is not completely lost as the decoder can proceed in the reverse direction; this improves error resilience of the bit stream. Furthermore, the motion vector range is extended to up to ± 256 pixels, depending on the picture size, as depicted in Table 2.1. This is very useful given the wide range of new picture formats available in H.263+.

Advanced Intra Coding mode (annex I) This mode improves compression performance when a macroblock is coded in *intra mode*. In this mode, DCT coefficient prediction from neighboring blocks, a modified inverse quan-

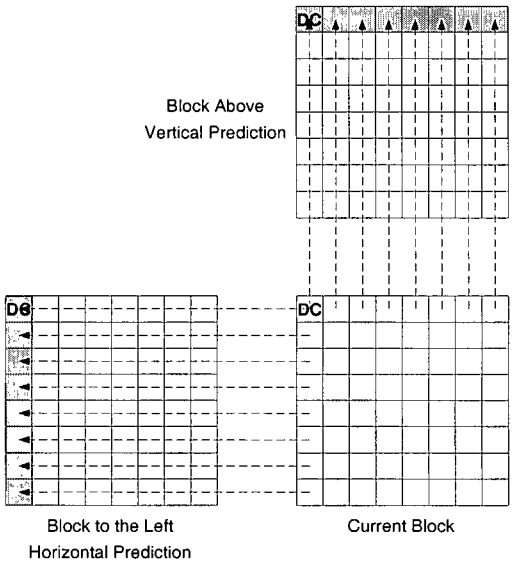


Figure 2.6: Neighboring blocks used for prediction in H.263+ advanced intra coding mode.

tization of DCT coefficients and a separate VLC table for DCT coefficients are employed. Block prediction is performed using data from the same luminance or chrominance components (Y, Cr or Cb). As illustrated in Figure 2.6, one of three different prediction options can be signaled: DC only, vertical DC & AC, or horizontal DC & AC. The option that yields the best prediction is applied to all blocks of the subject macroblock. The difference coefficients, obtained by subtracting the predicted DCT coefficients from the original ones, are then quantized and scanned differently depending on the selected prediction option. Three scanning patterns are used: the basic zig-zag scan for DC only prediction, the alternate-vertical scan (as in MPEG-2) for horizontally predicted blocks or the alternate-horizontal scan for vertically predicted blocks. The

main part of the standard employs the same VLC table for coding all quantized coefficients. However, this table is designed for *inter mode* macroblocks and is not very effective for coding *intra mode* macroblocks. In *intra mode* macroblocks, larger coefficients with smaller runs of zeros are more common. Thus, advanced intra coding mode employs a new VLC table for encoding the quantized coefficients, a table that is optimized to global statistics of *intra mode* macroblocks.

Deblocking Filter mode (annex J) This mode introduces a deblocking filter inside the coding loop. Unlike in post-filtering, predicted pictures are computed based on filtered versions of the previous ones. A filter is applied to the edge boundaries of the four luminance and two chrominance 8×8 blocks. The filter is applied to a window of four edge pixels in the horizontal direction and it is then similarly applied in the vertical direction. The weight of the filter's coefficients depend on the quantizer step size for a given macroblock, where stronger coefficients are used for a coarser quantizer. This mode also allows the use of four motion vectors per macroblock, as specified in advanced prediction mode of H.263, and also allows motion vectors to point outside picture boundaries, as in unrestricted motion vector mode. The above techniques, as well as filtering, result in better prediction and a reduction in blocking artifacts. The computationally expensive overlapping motion compensation operation of advanced prediction mode is not used here in order to keep the additional complexity of this mode minimal.

Alternative Inter VLC mode (annex S) The VLC table designed for encoding quantized *intra mode* DCT coefficients in advanced intra coding mode can be used for encoding quantized *inter mode* DCT coefficients when this mode is enabled. Large quantized coefficients and small runs of zeros, typically present in *intra mode* blocks, become more frequent in *inter mode* when small quantizer step sizes are used. When bit savings are obtained, and the use of the *intra mode* VLC table can be detected at the decoder, the encoder will use the *intra mode* VLC table.

Modified Quantization mode (annex T) Modified quantization mode includes three features. First, it allows rate control methods more flexibility by permitting the quantizer step size to be changed to any value at the macroblock layer. Second, it enhances chrominance quality by specifying a finer chrominance quantizer step size. Third, it improves picture quality by extending the range of representable quantized DCT coefficients, improving reconstruction quality for small quantizer step sizes.

2.3 Layered Video Coding

Layered video encoding is essential due to the growing interest in carrying video over the current non-uniform and sub-optimal network infrastructure. Layered video encoding was first proposed in [27]. A layered framework creates a flexible bitstream that can be manipulated at any point after it has been generated. This property is desirable in order to counter limitations that, in

the case of multi-point and multicast session, cannot be foreseen at the time of encoding.

In layered video encoding algorithms, there are two main approaches to the prediction of enhancement layer information. The first approach uses only the base layer information to form the prediction [28]. This includes techniques such as re-quantization [29], multi-stage quantization [30], progressive coding [31] and more recently fine granularity scalability [32]. Since this approach completely ignores the high quality information available in the previous enhancement layer reconstruction, it can result in repeated encoding of refinement information for persistent static image regions. Generally, this approach suffers from poor enhancement layer coding efficiency. The second approach relies only on the previous enhancement layer reconstruction to form the prediction [33, 34]. This approach completely ignores the information available in the current base layer reconstruction. As such, it performs poorly in the presence of certain types of motion, for example occlusions, which the base layer reconstruction will capture. Recently, layered video encoding algorithms having more flexible approaches to selecting the source for prediction have been proposed. In [35], a promising estimation-theoretic approach was introduced. This approach allows for switching the prediction of each transform coefficient between the corresponding reconstructed base layer coefficient or (motion compensated) reconstructed enhancement layer coefficient. Layered encoding, as supported in H.263+ [1] allows the source for prediction to be selected at the macroblock level. Prediction can be made from the corresponding

reconstructed base layer macroblock, a motion compensated macroblock from the previous enhancement layer reconstruction, or the linear interpolation of the two. For our work, we employ a fully standard-compliant H.263+ layered video encoding algorithm.

Several researchers have focused on non-DCT approaches having inherently scalable properties, such as subband-based transform models [30, 36, 37, 38]. Unfortunately, while these algorithms perform well for still image coding, they usually suffer from inferior compression efficiency due to the difficulty of effectively including a good motion model within subband schemes.

2.3.1 Types of Scalability

There are three well-known types of scalablity. These are illustrated in Figure 2.7. The first type, SNR scalability, is illustrated in Figure 2.7 (a). SNR scalability implies the creation of multi-rate bit streams. It allows for the recovery of coding error, or difference between an original picture and its reconstruction, in a reference layer by encoding this error as an enhancement layer, using a finer quantizer in the enhancement layer as compared to the reference layer. This additional information increases the SNR of the overall reproduced picture, hence the term SNR scalability.

The second type of scalability, spatial scalability, is illustrate in Figure 2.7 (b). It is essentially the same as SNR scalability except for the fact that a spatial enhancement layer attempts to recover the coding loss between an upsampled version of the decoded reconstructed reference layer picture and a

higher resolution version of the original picture.

The third type, temporal scalability provides a mechanism for enhancing perceptual quality by increasing the picture display rate. This is achieved via bi-directionally predicted frames, inserted between anchor frame pairs and predicted from either one or both of these anchor frames, as illustrated in Figure 2.7 (c). The resulting frames are never used as predictions for other frames. Therefore, they can be discarded without impacting picture quality of future frames, hence the temporal scalability feature. Note that while bi-directionally predicted frames can improve compression performance, as compared to P pictures, they add complexity and increase storage requirements. We do not consider temporal scalability in this thesis.

2.3.2 H.263+ Layered Video Coding, Scalability mode (annex O)

In addition to the numerous optional modes discussed previously, H.263+ specifies an optional mode for layered coding. This mode specifies syntax to support SNR, spatial and temporal scalability capabilities. Further details on H.263+ layered encoding can be found in [39, 40, 41].

In either SNR or spatial scalability, the enhancement layer pictures are referred to as EI- or EP-pictures, as illustrated in Figure 2.7 (a) and (b). If the enhancement layer picture is upward predicted, from a picture in the reference layer, then the enhancement layer picture is referred to as an Enhancement-I (EI) picture. A picture that can be forward predicted from a previous en-

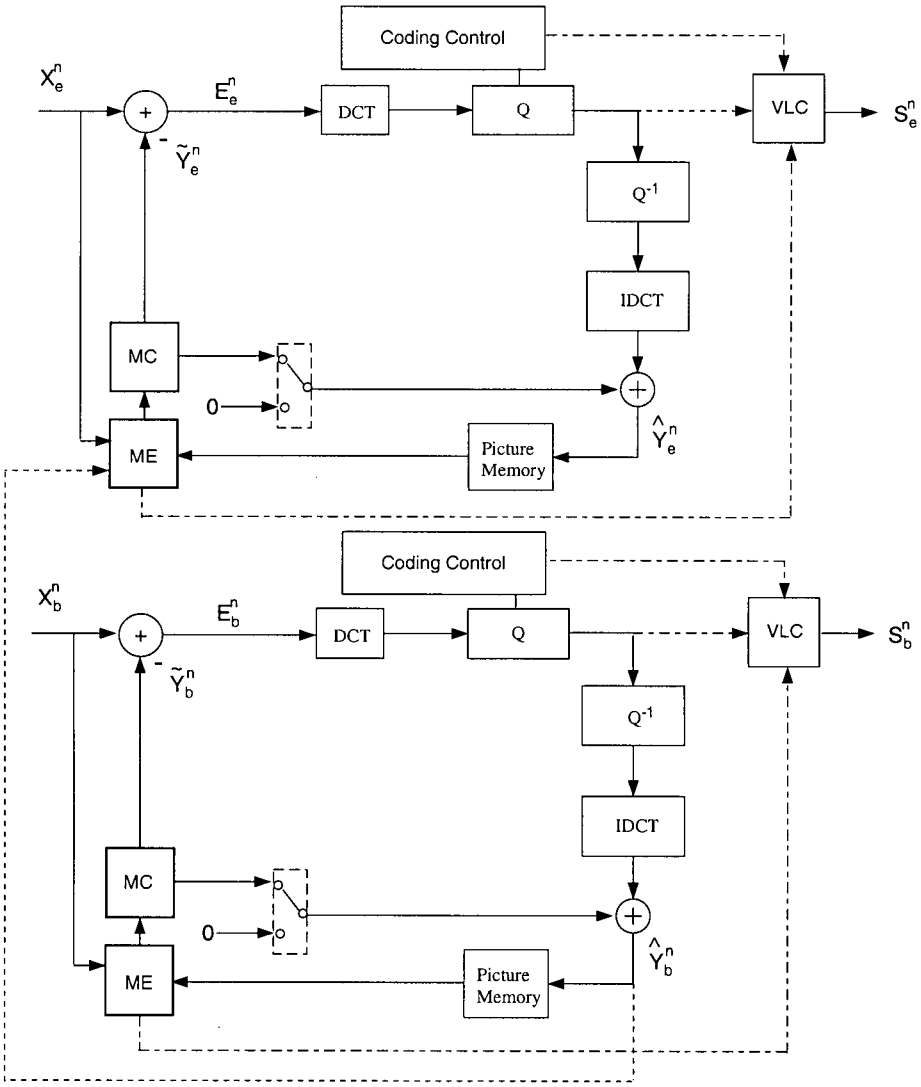


Figure 2.8: Generalized block diagram for scalable hybrid MC-DCT video encoder.

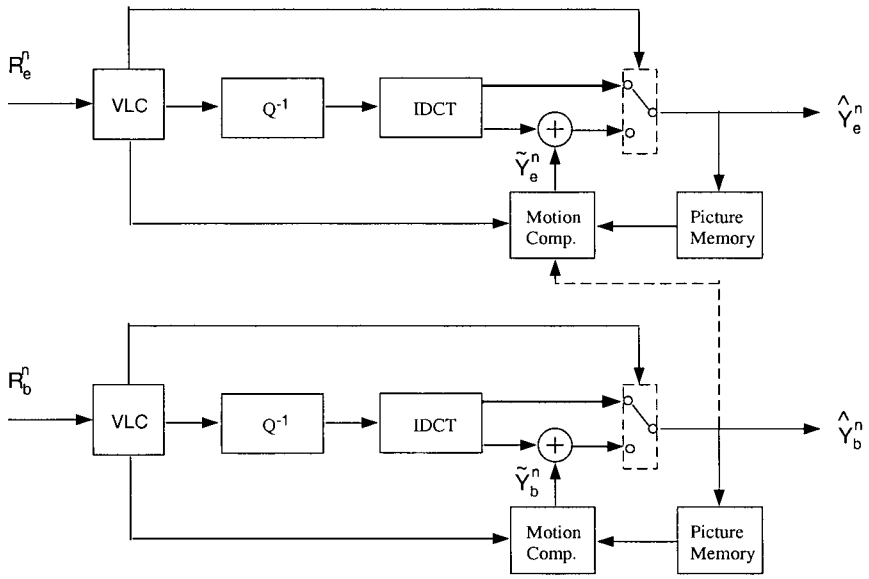


Figure 2.9: Generalized block diagram for scalable hybrid MC-DCT video decoder.

hancement layer picture or upward predicted from the reference layer picture is referred to as an Enhancement-P (EP) picture. The bilinear interpolation of the upward and forward predicted pictures is also permitted as a prediction option for EP-pictures. For both EI- and EP-pictures, upward prediction from the reference layer picture implies no motion vectors are required. In the case of forward prediction for EP-pictures, motion vectors are required. As stated above, H.263+ permits the source for prediction to be selected at the macroblock level. How to choose an appropriate coding mode for a particular block for enhancement layer prediction is one of the questions that this thesis attempts to answer, for both error-free and error-prone environments.

A block diagram of a two-layered H.263+ video encoder is shown in

Figure 2.8 and the corresponding decoder is shown in Figure 2.9. The switches in the base layer represent the choice between the *intra mode* and *inter mode.* In the enhancement layer, the motion estimation stage is also provided with the base layer reconstruction. Therefore, for *inter mode* in the enhancement layer, a choice must also be made between the motion compensated enhancement layer reconstruction and the current base layer reconstruction. The input signals to the encoder at time n are x_b^n and x_e^n for the base and enhancement layers respectively. In the case of SNR scalability, $x_e^n = x_b^n$. For an error free-channel, $r_b^n = s_b^n$ and $r_e^n = s_e^n$. Thus, the only source of error between the original signal and the decoded and reconstructed signal is $x_b^n - \hat{y}_b^n = q_b^n$ for the base layer and $x_e^n - \hat{y}_e^n = q_e^n$ for the enhancement layer, where q_b^n and q_e^n are the quantization errors for the base and enhancement layers respectively. However, in the case of packet loss, $r_b^n = r_e^n = 0$. The lost information should be concealed by the decoder. Therefore, $\hat{y}_b^n = c_b^n$ and $\hat{y}_e^n = c_e^n$, where c_b^n and c_e^n are the blocks used for concealment in the base and enhancement layers respectively. These concealment errors may propagate temporally. For example, if we consider packet loss in the enhancement layer, the error for prediction from a previously concealed region will be $x_e^n - \hat{y}_e^n = q_e^n + (\hat{y}_e^{n-1} - x_e^{n-1}) = q_e^n + (c_e^{n-1} - x_e^{n-1})$ where the second term represents the additional error due to concealment. Therefore, when considering the distortion to determine an appropriate coding mode, we should consider the effects of prediction from potentially concealed regions as well as the potential cumulative effects of concealment. Note that, in the case of layered coding, for packet loss in the

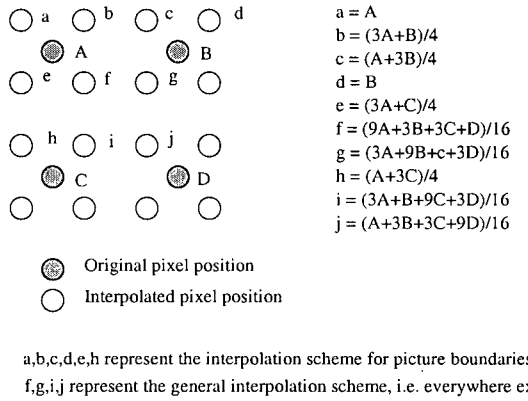


Figure 2.10: Interpolation filters for spatial scalability.

enhancement layer only, we can choose $c_e^n = \hat{y}_b^n$ and it is reasonable to expect $\hat{y}_b^n$ to be a good approximation of $\hat{y}_e^n$.

As stated above, the only difference between SNR and spatial scalability is that a spatial enhancement layer attempts to recover the coding loss between an upsampled version of the reconstructed reference layer picture and a higher resolution version of the original picture. For example, if the reference layer has a QCIF resolution, and the enhancement layer has a common intermediate format (CIF) resolution, the reference layer picture must be scaled accordingly such that the enhancement layer picture can be appropriately predicted from it. The interpolation filters used to upsample the reference layer picture are explicitly defined in the standard and are illustrated in Figure 2.10.

2.4 Rate Distortion Optimized Video Coding

Classical Rate-Distortion Theory

The underlying philosophy of Shannon's pioneering work on rate-distortion theory [3, 42], namely the fundamental tradeoff between fidelity and rate in lossy coding systems, is the essence of many modern signal processing problems [7, 43], not the least of which is that of lossy image and video coding.

Classical rate-distortion theory [44] is concerned with, for a given source distribution and distortion measure (or fidelity criteria) [45], bounding the region of achievable rate-distortion points, either

- the minimum expected distortion achievable at a particular rate or
- the minimum rate description required to achieve a particular distortion.

The impact of rate-distortion theory on practical lossy source coding was not immediate [43]. The main obstacles, aside from the separation of research communities working in each field, were twofold. First, the theoretical bounds were derived using simple statistical models that did not accurately characterize real sources. Second, there was a feeling that implementation complexity (in terms of delay, memory, or computations) would be prohibitive, given the random coding arguments used to prove the theoretical results. In the past few decades, these problems have become less important, and the field of operational rate-distortion has emerged as a fundamental framework for practical coding system design.

Operational Rate-Distortion Optimization

Operational rate-distortion optimization [7] is grounded in Shannon's philosophy of rate-distortion theory. In an operational rate-distortion framework, the encoder makes coding decisions based on the rate-distortion operating points that arise from applying particular choices of coding parameters. In sweeping through all possible combinations, for a given source and coding framework, an operational rate-distortion curve can be traced out. As the set of coding parameters is finite, the problem is essentially a discrete optimization. The operational rate-distortion bound is then the convex hull of the set of all operating points. If operating points exist, then an achievable solution also exists, although this does not guarantee that the achievable solution is optimal. It is important to recognize the importance of the underlying operational model the coder employs. This operational model dictates the set of coding parameters and admissible combinations thereof. A highly optimized model that is fundamentally poor, e.g. one which fails to efficiently capture the main characteristics of the source, can yield significantly inferior performance relative to an unoptimized model that is good. The operational rate-distortion problem can be stated more formally as follows:

> *Given a constraint R_c, a coding parameter $x \in X$, some constraint function $R(x)$, and some objective function $D(x)$ to be minimized, find*

$$\min_{x \in X} D(x), \text{ subject to} \tag{2.3}$$

$$R(x) \leq R_c. \tag{2.4}$$

The constrained problem of (2.3) and (2.4) can be converted to an equivalent unconstrained problem using the discrete Lagrangian optimization formulation [46]. The problem then becomes [47]:

For any $\lambda \geq 0$, the solution $x^*(\lambda)$ to the unconstrained problem,

$$\min_{x \epsilon X} D(x) + \lambda R(x), \tag{2.5}$$

is also the solution to the constrained problem of (2.3) with the constraint $R_c = R(x^*(\lambda))$.

For a given value of λ and a parameter choice $x \epsilon X$, the function (2.5) produces the corresponding cost, which we refer to as the Lagrangian cost, or simply the Lagrangian, from here-on. For a given λ, we can test all permissible choices of $x \epsilon X$ from which a choice $x^*(\lambda)$ that minimizes (2.5) can be selected. For $\lambda = 0$, minimizing (2.5) is equivalent to minimizing the distortion only. For $\lambda \rightarrow \infty$, minimizing (2.5) is equivalent to minimizing the rate. As we sweep λ from 0 to ∞, we obtain operating points having different rate-distortion tradeoffs, where a given value of λ represents a specific operating point on the rate-distortion curve.

Applications to Video Coding

Operational rate-distortion optimization was first applied to source coding in [47, 48] and has been widely applied since. In hybrid MC-DCT video coding, the task is to choose, for each coding unit, the most efficient coded representation in a rate-distortion sense. The coding unit is generally chosen to be a macroblock. In the case of lossy video coding, the set of coding parameters for

a coding unit includes the motion vectors, quantizer step size, and the coding mode. The selection of one combination from the set of these parameters constitutes the discrete optimization discussed above and can be solved optimally using principles of operational rate-distortion optimization.

A complicating factor in the optimization framework is the dependencies between parameters selected from one coding unit to the next and from one frame to the next. The former is due to the differential encoding of these parameters between coding units, while the latter is a result of the inherent dependencies in a predictive coding framework. In the majority of the literature, various simplifications are made such that the optimization task remains tractable. For example, a common assumption is that inter-frame dependencies can be ignored, in the sense that the selection of optimal parameters for frame n assumes that the optimal parameters for frame $n-1$ have already been determined. Clearly, the benefit of this assumption is significantly reduced complexity and encoding delay. We propose and quantify the reduction in complexity for several simplifications in Chapter 4. The modified optimization algorithm selects a locally optimal set of parameters.

The Lagrangian parameter λ can be selected in many ways. It would be beneficial to be able to select the value of λ *a priori*, such that a known rate constraint could be closely matched. Given the known monotonicity property between λ and rate, one solution is the bisection algorithm [49]. However, this usually requires several encoding iterations with different values of λ until the target rate is matched. In [50] a least mean squares (LMS) adaptation [51]

approach is employed to update λ using

$$\lambda(t) = \lambda(t-1) + (\frac{R_c}{R(t-1)} - 1), \tag{2.6}$$

where R_c is the target rate and $R(t-1)$ is the actual rate for encoding the previous frame. Another technique for setting λ using a feedback approach is presented in [52], where λ is a function of the current buffer state. In [53] λ is controlled using the recursion formula

$$\lambda(t) = \lambda(t-1)\frac{s(t-1)}{s^*}, \tag{2.7}$$

where $s(t-1)$ denotes the actual buffer fullness and s^* denotes the ideal desired buffer fullness, which is usually one-half. In [54], a relationship between the quantizer step size Q and λ was presented

$$\lambda(t) = c\left(\frac{Q}{2}\right)^2, \tag{2.8}$$

where c is a constant that depends on the coding framework. This relationship is obtained by recording the quantizer step size Q that minimizes the Lagrangian for a given fixed value of λ. For constant bit rate (CBR) applications, this framework depends on another mechanism to adapt Q appropriately such that the rate constraint is satisfied. This method clearly eliminates the need to search for the optimal operating point. In Chapter 4, such an approach is investigated for the dependent layered coding framework of this thesis. Models are developed to control the Lagrangian parameter for enhancement layer frames with reduced complexity.

Rate-distortion optimized motion estimation has been widely studied in the literature. In [55] rate-distortion optimization for variable block-size

motion estimation algorithms is proposed. Ignoring macroblock dependencies, a multi-level quadtree structure is constructed for each (largest possible) block size. The rate-distortion optimal motion vector is found for the largest possible block. Then rate-distortion optimal motion vectors are found for each sub-block of the quadtree structure. Taking into consideration the associated rate to describe the quadtree structure, Lagrangian costs for all possible blocks in the quadtree are computed. From this, the optimal block sizes and associated motion vectors are selected. In [56, 50] a rate-distortion optimal macroblock coding mode selection algorithm, formulated as a dynamic programming problem [57], is proposed for an H.263 video encoder. A trellis is constructed for each row of macroblocks, where each stage represents a macroblock and each node represents a particular choice of coding mode and quantizer level. To reduce complexity, the quantizer level is not permitted to change between nodes. The branches account for the dependency between the coding mode and motion vector rate components. In [54] the algorithm is extended to incorporate rate-distortion optimized motion estimation. Furthermore, a new approach for selecting λ, using (2.8), is presented. This framework is then employed to analyze the rate-distortion tradeoffs in a sophisticated MC-DCT video encoder based on H.263+. In [58, 59] a rate-distortion optimized motion estimation algorithm formulated as a dynamic programming problem is presented. Assuming one-dimensional differential encoding of motion vectors, a trellis is constructed where each stage represents a macroblock, each node represents a particular motion vector choice (with corresponding resid-

ual) and each branch represents the dependency introduced by the motion vector rate component. In [58] the algorithm is extended to accommodate variable block size motion compensation. In [60, 61, 62] a rate-distortion optimized bit allocation between motion and residual encoding formulated as a dynamic programming problem is presented. Assuming 1-D differential encoding of motion vectors, a multi-level trellis is constructed, where each level represents a quadtree segmentation of the previous level, each node represents a particular motion vector and quantizer choice for the block size determined by the level. The branches between nodes within a level represent the dependency introduced by the motion vector rate component while the branches between nodes in different levels represent the dependency introduced by both the motion vector rate and the segmentation overhead components. In [63] a rate-distortion optimized motion estimation and mode decision algorithm is presented. Using well-known training techniques [64], quantizer dependent parametric functions are obtained to approximate the rate for encoding a prediction error block given the obtained motion estimation distortion measure. The parameter λ is selected by preprocessing a portion of the input sequence and estimating the rate-distortion curve. This work is extended in [53] to study the impact of the dependency introduced by the motion vector rate component. Various motion vector prediction techniques are evaluated within a rate-distortion optimized H.263 encoder. The well-known median prediction is found to yield the best performance and permits a constrained search area to be employed for motion estimation. A novel fast-search pattern that ex-

ploits this constraint is presented and is demonstrated to be significantly more efficient the the exhaustive search algorithm with little or no degradation in rate-distortion performance. Moreover, a new approach for selecting λ, using (2.7), is presented.

Rate-distortion optimized quantization has also been studied. Here, the goal is to select the rate-distortion optimal quantizer output level given the input level. In [48, 65], this is achieved by biasing the decision thresholds towards lower rates. In [66], an iterative greedy algorithm prunes quantized DCT coefficients by minimizing that ratio of increase in distortion to decrease in bit rate. This technique is also employed in the H.263 reference model [67]. In current video coding standards, the quantized transform coefficients are zig-zag scanned and run-length coded, as described above. This leads to a complex rate inter-dependency between neighboring levels. In [68, 69] this complexity is addressed using a trellis-based rate-distortion optimization technique, where each stage of the trellis represents a coefficient position and each node represents a specific run and level for the coefficient in that position.

Operational Rate-Distortion Optimization for Layered Coding Frameworks

For a layered coding framework, choices made for the coding parameters for the independent base layer will have an impact not only on neighboring coding units and subsequent frames, but also on the corresponding frame in the dependent enhancement layer. In [70], the bit allocation problem is addressed for both temporally and spatially dependent coding frameworks. We do not

consider temporal dependencies in this thesis, due to the enormous complexity of such schemes. However, the spatial dependencies are of great interest, as they affect the error-recovery performance of the system.

Beginning from the solution to the optimal independent allocation case [47], where all coding units operate at a constant slope λ on their operational rate-distortion curves, the general problem can be posed as follows:

> *Without loss of generality, consider a two-layer dependency, where the rate-distortion operating points for the second layer are dependent on the choice of coding parameters made in the first layer. Given a constraint R_c, coding parameters x_1, $x_2 \in X$, some constraint functions $R_1(x_1)$ and $R_2(x_1, x_2)$, and some objective function to be minimized $D_1(x_1)$ and $D_2(x_1, x_2)$, find $D(x)$ to be minimized, find*

$$\min_{x_1\, x_2 \in X} [w_1 D_1(x_1) + w_2 D_2(x_1, x_2)], \text{ subject to} \tag{2.9}$$

$$R_1(x_1) + R_2(x_1, x_2) \leq R_c. \tag{2.10}$$

The constrained problem of (2.9) and (2.10) can be converted to an equivalent unconstrained problem using the discrete Lagrangian optimization formulation [46]. The problem then becomes [47]:

> *For any $\lambda \geq 0$, the solution $x_1^*(\lambda)$ and $x_2^*(\lambda)$ to the unconstrained problem,*

$$\min_{x_1, x_2 \in X} [J_1(x_1) + J_2(x_2)], \text{ where} \tag{2.11}$$

$$J_1(x_1) = w_1 D_1(x_1) + \lambda R_1(x_1) \text{ and} \tag{2.12}$$

$$J_2(x_1, x_2) = w_2 D_2(x_1, x_2) + \lambda R_2(x_1, x_2), \tag{2.13}$$

is also the solution to the constrained problem of (2.9) with the constraint $R_1(x_1^*) + R_2(x_1^*, x_2^*) \leq R_c$.

Again, as λ is swept from 0 to ∞, the convex hull of the rate-distortion curve for the dependent allocation problem is traced out. The search for $x_1^*(\lambda)$ and $x_2^*(\lambda)$ is done by, for the given value λ, finding the optimal solution, for all choices x_1 of the independent layer layer, $x_2^*(x_1)$, which "lives" at the absolute slope λ on the dependent layer rate-distortion curve associated with x_1 [70]. It is straightforward to extend this results to N-layer dependencies.

The dependency between the base and enhancement layers, D_1 and D_2, is critical. As such, additional constraints must be placed on the base layer. Otherwise, if only the full-resolution distortion D_2 is minimized under the total rate constraint, $R_1(x_1) + R_2(x_1, x_2) \leq R_c$, the resulting base layer quality may be unacceptable. Therefore, an additional constraint on the base layer bit rate is imposed,

$$R_1(x_1) \leq R_{c_1}. \tag{2.14}$$

In a sense, this is sacrificing a small amount of the enhancement layer quality to ensure acceptable base layer quality. To achieve this, at optimality, each layer must operate at its own constant slope, λ_1 and λ_2. The base layer should then just satisfy the added constraint (2.14), operating at its constant slope. All remaining bits should then be allocated to the enhancement layer, again operating at its own constant slope. This guarantees that, for the particular allocation to the base layer, no better distortion performance can be achieved.

Similarly, for the particular allocation to the enhancement layer, given the allocation to and optimality of the base layer, no better distortion performance can be achieved. This also has the added benefit of greatly reducing the complexity of the optimization task, as it removes the need to consider spatial dependencies, i.e. the minimization should be performed independently for each layer [70].

2.5 Error Resilient Video Coding and Transport

Error resilient, or robust, video communications is essential due to the growing interest in carrying video over the current non-uniform and sub-optimal network infrastructure. In the case of packet-switched networks, network congestion and buffer overflow inevitably lead to packets being delayed and discarded. Approaches to recover from packet loss can be broadly categorized as closed-loop, for example retransmission protocols, and open-loop, for example forward-error correction (FEC) techniques. However, in some scenarios a closed-loop approach may not be possible, for example in some multi-point or multicast sessions. Therefore we consider only an open-loop approach.

The simplest and most popular open-loop methods to recover from packet loss rely on the decoder alone to perform error concealment through post-processing [6]. These methods can be broadly classified into spatial and temporal domain approaches [71]. Unfortunately, under anything more than

very light losses, such methods are not sufficient to provide acceptable quality video. Under medium to heavy losses, the encoder and decoder quickly lose synchronization, leading to rapid and devastating spatio-temporal error propagation.

One solution is to include some form of pro-active error recovery in the system. This can be in the form of adding controlled source coding redundancy, channel coding redundancy or some combination of the two. However, until the affected regions are updated without motion information, i.e. through intra-coding, the encoder and decoder will remain unsynchronized. Because coding in *intra mode* is expensive, in terms of the number of bits required. Various approaches have been proposed for selecting the appropriate amount of *intra mode* coding [72, 73, 74]. Residual loss effects can then be concealed by the source decoder.

In this section we first discuss the general issues of packet video communications. We then describe the effects of packet loss. We highlight the main techniques for robust video communications, paying particular attention to techniques closely related to those proposed in this thesis.

2.5.1 Packet Video Communications

It is well-known that packet-switching increases utilization of a physical channel, by permitting multiplexing of many different connections. However, this inevitably leads to delays and packets may even be dropped under heavy congestion. For best-effort packet networks such as the Internet, there exist no

quality-of-service (QoS) mechanisms to guarantee delivery of packets with a given fidelity. For traditional data communications applications, higher level protocols like TCP/IP are necessary to guarantee end-to-end delivery. However, for real-time or delay sensitive media, such as audio and video, the end-to-end latency incurred from TCP's retransmission delays are unacceptable. Therefore, UDP/IP is the protocol of choice. UDP provides no guarantee of end-to-end delivery. Thus, the video communications system must be robust to delay and packet loss.

For handling data having a real-time constraint, in addition to UDP, the Internet draft real-time transport protocol (RTP) [75] is employed. RTP requires that packets contain real-time information such as time-stamp, sequence number and payload data type. Typically, for each different media type, a separate payload specification is required, such as [76, 77] for H.263 and H.263+ data respectively. It is important to recognize that RTP does not provide any mechanism to guarantee QoS or real-time delivery. However the sequence numbers allow for easy detection of packet loss. This comes at the expense of additional channel rate. For example, the packetization overhead for IP/UDP/RTP headers is approximately forty bytes per packet.

Note that we can actually categorize two types of transmission errors, random bit errors and erasures. For the Internet, the bit error rate is effectively zero. Furthermore, in the case of random bit errors in VLCs, the bits following the bit in error may not be decodeable, effectively resulting in an erasure. Therefore we consider only erasures, i.e. packet loss, in this thesis.

or eliminate the extent of error propagation, otherwise the visual quality can degrade significantly and rapidly.

2.5.3 Error Resilient Video Communication Techniques

In this section, we discuss relevant error resilient video coding and transport techniques in more detail. We classify the techniques according to whether the encoder or decoder play the primary role [6]. For the first class, forward error concealment techniques, the source and/or channel encoders play the primary role. For the second class, post-processing techniques, the decoders play the primary role. In this dissertation, we do not consider interactive error concealment techniques, that rely on cooperation between the encoder and decoder, as they may not suitable for certain scenarios, for example some multi-point or multicast sessions.

Forward Error Concealment

Forward error concealment techniques rely on the source and/or channel coder to play the primary role to simplify the error concealment task at the decoder. Typically this is accomplished by introducing a controlled amount of redundancy to the system, via the source and/or channel coder. We now review popular error resilient video communication techniques that are related to those proposed in this thesis.

Layered Coding With Transport Prioritization Techniques To date this has been the most popular and effective scheme for providing a robust

video communications system [6]. As described in Section 2.3, the video information is partitioned into two or more layers. It is clear that layered coding must be combined with some sort of transport prioritization to combat channel errors such that the base layer, containing the highest priority data, is delivered with higher reliability. Prioritization can be achieved in several ways. First, the network itself may support transport prioritization as is the case for ATM. In a wireless environment, transport prioritization can be achieved by using different levels of power to transmit the individual layer streams. Finally, transport prioritization can be achieved by adding different amounts of FEC to the individual layer streams, i.e. unequal error protection.

Layered coding with transport prioritization was first introduced for video in [27], where the base layer data, based on hybrid DPCM-DCT coding, is transmitted as high priority using a guaranteed QoS over ATM networks. Enhancement layer data based on DPCM is transmitted as low priority. In [29], the coding efficiency of the overall system is improved by replacing the base layer coder with a standard H.261 [23] coder. In [78], a layered network architecture model is discussed to support packet video communications. Using a non-motion adaptive 3-D subband coder, baseband data is transmitted with high priority while non-baseband data is transmitted as low priority. Prioritization is simulated by applying different packet loss rates to the high and low priority packets. In [79] a multi-resolution joint source/channel coder, based on 3-D spatio-temporal pyramid decomposition and embedded trellis-coded modulation (TCM) [80] is proposed. The resulting spatio-temporally

subsampled image sequence constitutes high priority data while residual images, following spatial and temporal interpolation, produce additional layers which are considered low priority. The TCM scheme allows for two-level embedding, hence two priority levels. In [81], a pyramid coder that employs a standard H.261 [23] coder in the base layer and a separate motion compensation loop in the enhancement layer is presented. The delivery of base layer data is assumed to be guaranteed thus it is high priority, while enhancement layer data is considered low priority. In [82] the performance of MPEG-2 [83] SNR scalability, spatial scalability and data partitioning for ATM networks is discussed. Prioritization is simulated assuming that the delivery of base layer data is guaranteed while enhancement layer data is subject to random cell loss. This leads to a discussion of data partitioning.

Data Partitioning Data partitioning is a form of layered coding that usually does not encode new information. It re-orders and/or separates elements of the data stream such that elements of similar importance or priority are grouped together. This is beneficial as an appropriate priority can then be assigned to such a group based on the importance of the contained elements, relative to the importance of the elements in the other groups.

In [84], data partitioning of DCT coefficients in an MC-DCT coder is studied for ATM networks. The partitioning is performed both on a fixed rate threshold and a fixed energy threshold. The low frequency coefficients that fall within the threshold are included as high priority data while the remaining coefficients are considered low priority data. In [85] a quadtree DPCM-DCT

progressive transmission scheme is proposed. Based on a threshold, it is determined whether or not an image region is further decomposed. The resulting low frequency coefficients are included as high priority data while the remaining high frequency coefficients are considered low priority data. Prioritization is simulated using different packet loss rates for high and low priority packets. In [86] fixed position coefficient segmentation is proposed for MPEG-1 [87] video on ATM networks to generate four partitions. Each partition is considered to have a different priority. Prioritization is simulated by applying different packet loss rates to the different partitions. In [88] an adaptive algorithm for partitioning DCT coefficients from a DPCM-DCT coder is proposed for transmitting high quality video on ATM networks. The algorithm considers the amount of energy contained in a subset of low frequency coefficients. The low frequency coefficients are included as high priority data using guaranteed QoS.

Recently, video coding standards have recognized the benefits of data partitioning in addition to layered coding. Data partitioning is particularly interesting for wireless environments, where transmission errors occur but the bit rate requirements of SNR or spatial scalability may not be acceptable. In [82] MPEG-2 data partitioning is analyzed. H.263 Version 3 will likely include a mode to support data partitioning [89], and preliminary results were presented in [90]. Also, MPEG-4 [2] includes a data partitioning mode as part of its error resilience tools [91].

Forward Error Correction Techniques FEC is a well-known technique in data communications for error detection and correction [92]. It involves the transmission of redundant data with the original data so that, if some of the original data is lost, it can be recovered from the redundant data. The amount of redundant information is typically small, as FEC introduces overhead in the form of increased channel rate, so that the FEC remains efficient and does not reduce too severely the amount of channel rate usable by the source coder. Thus, the amount of FEC applied must be carefully selected, such that the benefits of its application, i.e. the amount of information receiver, prevails over the added rate it introduces, i.e. the amount of channel rate lost to the source coder. How to select an appropriate amount of FEC for a layered coding and transport framework in error-prone environments is another question we attempt to answer in this thesis.

In [93, 88] an ATM cell loss and recovery mechanism is presented for low loss rates. Cells are arranged in a two-dimensional matrix. Error detection cells are generated from the rows of the matrix while error correction cells are generated from the columns of the matrix by applying simple XOR codes. In [94] another method for cell loss recovery in ATM networks is proposed. Reed-Solomon FEC cells are generated for a block of data cells. The proposed method is analyzed for different levels of network congestion, different code rates and different numbers of sources generating additional FEC data.

Temporal Error Resilience Temporal error resilience techniques can be employed to limit the effects of temporal error propagation. For example, it is

possible to use an earlier picture than the last decoded one for temporal prediction. This reference picture can be chosen to minimize error propagation. This can be done with or without a feedback channel, using reference picture selection mode, annex N of H.263 [1]. As stated above, we do not consider interactive error concealment techniques, as they may not be suitable for certain multi-point or multicast sessions. Thus, we discuss only the sub-mode of annex N that does not require a feedback channel. This sub-mode is commonly referred to as video redundancy coding. In addition to video redundancy coding, we discuss one other approach to increase temporal error resilience. This approach works by introducing a controlled amount of *intra-mode* encoding to increase temporal error resilience.

Video redundancy coding improves temporal error resilience using multiple prediction options without the use of a feedback channel [95]. The principle of video redundancy coding is to divide the sequence of pictures into two or more threads, with each thread coded independently. The frame rate within one thread is much lower than the overall frame rate, which leads to a substantial coding efficiency penalty. At regular intervals, all threads converge into what is referred to as a sync frame. From this sync frame, a new thread series is started. Note that the sync frame is encoded within each thread, i.e. there is more than one representation of the picture scene at the same temporal instant. When this mode is employed and multiple adjacent pictures in the bitstream having the same temporal reference are received by the decoder, the decoder regards this as an indication that multiple representations

of the same picture scene content have been sent and it ignores all but the first representation. Thus, if one of these threads is damaged because of a packet loss, the remaining threads stay intact and can be used to predict the next sync frame. Experimental results [95] show that video redundancy coding with three threads and three pictures per thread provides good video quality for a picture loss rate of 20%.

Another simple and popular technique to avoid error propagation in the temporal direction is to increase the frequency of *intra-mode* encoding. Because coding in the *intra-mode* is expensive, in terms of the number of bits required, various approaches have been proposed for selecting the appropriate amount of *intra-mode* encoding. One simple approach is to encode macroblocks with the *intra-mode* in a random pattern [96, 97]. Another method was proposed in [72], where only blocks with high activity are coded in the *intra-mode*. In [98, 74] feedback information is used to select the encoding mode. This approach incorporates knowledge of the motion compensation error propagation and the error concealment method employed. In [73], rate-distortion theory is employed to determine when to encode with the *intra-mode* based on both the source coding distortion and the expected concealment distortion. In all cases, residual loss effects can then be concealed by the source decoder.

Error Concealment by Post-Processing

For these techniques, the decoder plays the primary role in error concealment. These methods typically rely on estimation and interpolation for performing the concealment and can be broadly classified into spatial and temporal do-

main approaches [6, 71]. Most of these techniques seek to exploit an assumed spatial and temporal smoothness occurring in natural images and image sequences. Spatial domain approaches attempt to estimate missing pixels from neighboring spatial information. Temporal domain approaches employ motion compensation to reconstruct missing pixels from information in previously reconstructed frames. In this thesis, we consider only temporal domain approaches. Next, we review well-known temporal domain approaches to post-processing error concealment that are related to those proposed in this thesis.

Motion Compensated Temporal Prediction Techniques A simple concealment method would replace lost blocks with the spatially corresponding blocks in the previously decoded frame. This is satisfactory for low activity blocks, however it can produce objectionable artifacts under high motion and non-motion changes. Concealment can be improved by replacing lost blocks with motion compensated blocks. If motion vector information is unavailable, it must be estimated as described below. Note that motion compensated temporal prediction can still produce objectionable artifacts if for example, the block being concealed was coded in *intra mode* due to high motion or non-motion changes.

Recovery of Coding Modes and Motion Vectors In the case where coding mode and motion vector information is lost they must first be estimated. Using the assumption of spatial and temporal smoothness, they can be estimated from spatially and/or temporally neighboring blocks. Several estimates

have been proposed, for example using the average, median and maximum a posteriori (MAP) estimates, or a side matching criterion [71]. In [99], it was found that using the median estimate for motion compensation yielded better subjective quality than the averaging technique. This is also the technique employed in the H.263 Test Model [67]. Therefore, in this thesis we employ the median estimate. In this approach, the motion vector for the missing block is set to the median value of the motion vectors from the blocks to the left, above and above right of the missing block. If no motion vectors are available in these positions, the estimated motion vector is set to $(0, 0)$. Note that in the case of layered coding, motion information can also be estimated from the corresponding base layer reconstruction. Enhancement layer temporal domain error concealment is another topic we address in this thesis.

2.6 Conclusion

In this chapter, we have reviewed low bit rate video encoding algorithms, H.263 algorithms in particular. We discussed layered video encoding. We reviewed operational rate-distortion optimization techniques within the context of video coding. Finally, we highlighted the most popular techniques for robust video communications.

Chapter 3

Efficient Layered Video Coding in Error-Free Environments

In this chapter, our main goal is to develop algorithms for efficient layered video encoding in error-free environments. We first evaluate the effectiveness of the key system parameters for layered video encoding. This not only provides valuable insight into the relative importance of the various technical features, it also motivates our rate-distortion optimization algorithm. We then determine an upper bound on the rate-distortion performance for layered video encoding. This is an important contribution of our work. Next, the general formulation for our layered video encoding algorithm for error-free environments is presented. This algorithm is the main contribution of the chapter. It is based on the principles of operational rate-distortion optimization and is demonstrated to achieve significant improvement in rate-distortion performance. Furthermore, we distinguish between overhead and data elements in the enhancement

layer bit stream. While the rate-distortion optimization algorithm is shown to improve the coding efficiency of data elements, it does not directly address the coding efficiency of overhead elements. Therefore, we present a detailed analysis of coding overhead inherent in the layered bit stream. We conclude the chapter with a study of the effects of the allocation of the total video bit rate between the base and enhancement layers.

3.1 Motivation

In Chapter 2, the key technical features of MC-DCT video encoding in general, and H.263+ in particular, that will be employed throughout this thesis were presented. To appreciate the effectiveness of these technical features, we illustrate the source coding performance as they are added incrementally to a video encoder whose operational mode is rate-distortion optimized. Following the approach in [54], we illustrate the encoding options as follows:

- *intra mode* only: Each macroblock is coded independently, similar to JPEG [100].
- *intra* and *skipped mode*: A macroblock can be coded in the *intra mode* or replaced by the macroblock at the same spatial location in the previously decoded frame.
- *intra*, *skipped*, and *inter mode* with (0,0) motion vector: A macroblock can be coded in the *intra mode*, *skipped mode*, or as a combination of the predicted macroblock in the previously decoded frame, displaced by

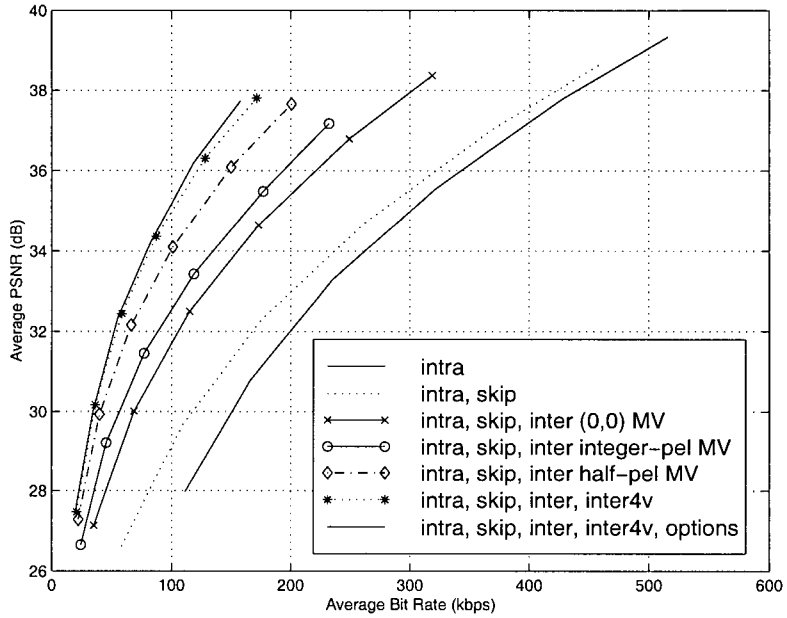
the (0,0) motion vector, along with DCT coding of the prediction error block.

- *intra*, *skipped* and *inter mode* with integer-pel accuracy motion vector: A macroblock can be coded in the *intra mode*, *skipped mode*, or as a combination of the predicted macroblock in the previously decoded frame, displaced by an integer-pel accuracy motion vector, along with DCT coding of the prediction error block.

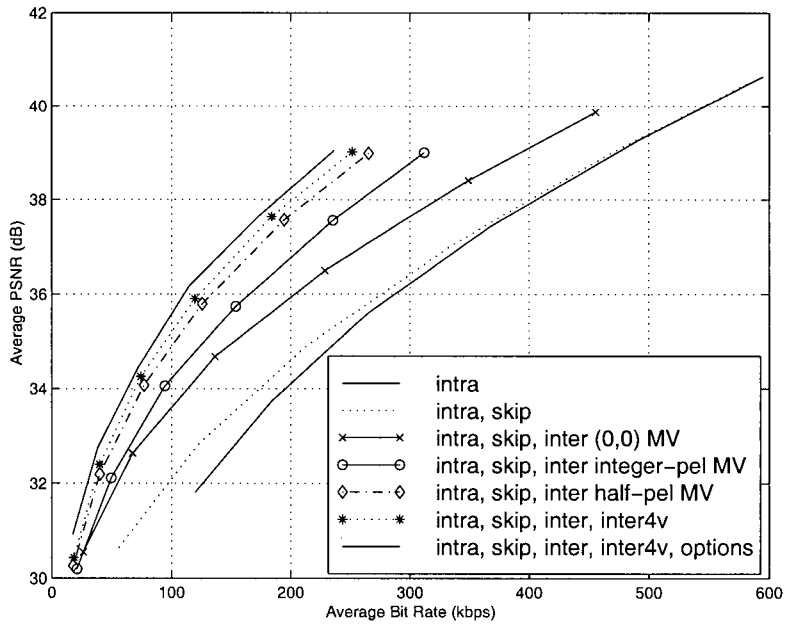
- *intra*, *skipped* and *inter mode* with half-pel accuracy motion vector: A macroblock can be coded in the *intra mode*, *skipped mode*, or as a combination of the predicted macroblock in the previously decoded frame, displaced by an half-pel accuracy motion vector, along with DCT coding of the prediction error block.

- *intra*, *skipped*, *inter* and *inter4v mode* with half-pel accuracy motion vectors: A macroblock can be coded in the *intra mode*, *skipped mode*, or as a combination of the predicted macroblock in the previously decoded frame, displaced by one or four half-pel accuracy motion vector, along with DCT coding of the prediction error block.

- *intra*, *skipped*, *inter* and *inter4v mode* with half-pel accuracy motion vectors and all additional H.263 optional modes: This is the same as the previous coder. In addition H.263 Annexes D, F, I, J, S and T [1] as described in Section 2.2.2 are enabled.



(a)



(b)

Figure 3.1: PSNR versus total bit rate, (a) FOREMAN and (b) COASTGUARD, QCIF, 10 fps, for the incremental addition of key technical features.

Sequence	Resolutions	Motion Activity	Spatial Detail
MOTHER AND DAUGHTER	QCIF and CIF	low	low
AKIYO	QCIF and CIF	low	low
HALL MONITOR	QCIF and CIF	low	low
CONTAINER SHIP	QCIF and CIF	low	low
FOREMAN	QCIF and CIF	medium	low
NEWS	QCIF and CIF	medium	low
SILENT VOICE	QCIF and CIF	low	medium
COASTGUARD	QCIF and CIF	low	medium

Table 3.1: A list and associated characteristics of well accepted video sequences used for testing within the low bit rate video communications research community.

Results are shown in Figure 3.1 for two video sequences, FOREMAN and COASTGUARD. In all cases where motion vectors are permitted, an exhaustive search algorithm is employed. All sequences consist of 300 frames, of which every third frame is coded, resulting in 100 coded frames. These sequences are representative of a set of sequences commonly used and well accepted within the low bit rate video communications research community. A list of these sequences and their characteristics is provide in Table 3.1 and the first frame of each sequence is shown in Figure 3.2.

Throughout this thesis, average peak signal-to-noise ratio (PSNR) is used as a distortion measure. For each color component, the PSNR is calculated as

$$PSNR = 10 \log \frac{255^2}{\frac{1}{M}\sum_{n=1}^{M}(o_n - r_n)^2}, \tag{3.1}$$

where M is the number of samples and o_i and r_i are the amplitudes of the original and reconstructed pictures respectively. The denominator is simply

the mean squared error (MSE). The average PSNR for a frame is computed as the weighted sum (4:1:1) of the PSNR for the luminance and two chrominance components. The average PSNR for the sequence is calculated as the average of the individual frame PSNRs. Alternatively, we could compute the PSNR for the frame with M representing the total number of pixels including the luminance and both chrominance components. Similarly, we could compute the PSNR for the sequence with M representing to total number of pixels including the luminance and both chrominance components of all frames. Although the MSE does not always correlate well to subjective quality, it is the most widely accepted objective quality measure in the image and video coding research communities. Recently, there has been significant activity through the image and video coding standardization efforts to determine and recommend an objective measure for subjective quality. It is interesting to note that the results of the initial phase of this testing indicate that PSNR performance is statistically equivalent to, or better than, the performance of other more sophisticated methods that were proposed [101].

From Figure 3.1 it is clear that the rich set of available coding parameters substantially improves coding efficiency. We observe as much as a factor of four increase in coding efficiency between the encoder employing only the *intra mode* and the encoder employing the full set of permissible encoding options.

Figure 3.3 illustrates how a less-sophisticated selection of coding parameters from the same available set fails to encode the same data as efficiently. In

this figure, we compare one encoder whose operational mode is rate-distortion optimized to another less-sophisticated encoder. The operational mode of the less sophisticated encoder is based on thresholds [67]. For mode decision it employs the minimum integer-pel SAD from motion estimation, where the (0,0) integer-pel motion vector is reduced by 100 to bias the decision towards the *skipped mode.* This SAD is used to determine whether or not to encode the macroblock in the *intra mode* as follows:

$$W < min\{SAD_{integer-pel,16x16}\} - 500 \tag{3.2}$$

When this holds, the *intra mode* is selected as the encoding mode for the macroblock. When this does not hold, the *inter mode* and the *inter4v mode* are then tested. For the *inter mode*, half-pel accuracy motion estimation is performed around the given integer-pel 16x16 motion vector. For the *inter4v mode*, the motion vectors are found by performing half-pel accuracy motion estimation, obviously on 8x8 blocks, also around the given integer-pel 16x16 motion vector. The *inter4v mode* is selected if the sum of the minimum half-pel SADS for the component 8x8 blocks is less than the minimum half-pel SAD for the 16x16 macroblock as follows:

$$\sum_{block=0}^{3} min\{SAD^{block}_{half-pel,8x8}\} < min\{SAD_{half-pel,16x16}\} - 200 \tag{3.3}$$

Finally, if the *inter mode* is selected and the motion vectors and quantized DCT coefficients are all zero, the macroblock is encoded in *skipped mode.*

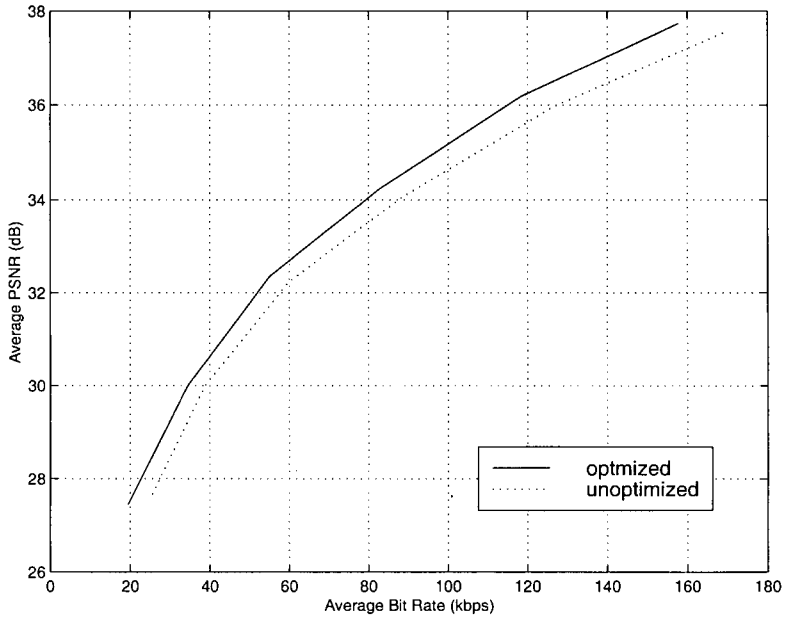
From Figure 3.3, we can conclude that rate-distortion optimization improves coding efficiency, in this case yielding around a 10% reduction in bit

rate, or 0.5 dB increase in PSNR. Obtaining such gains within a layered encoding framework is an important goal of our work.

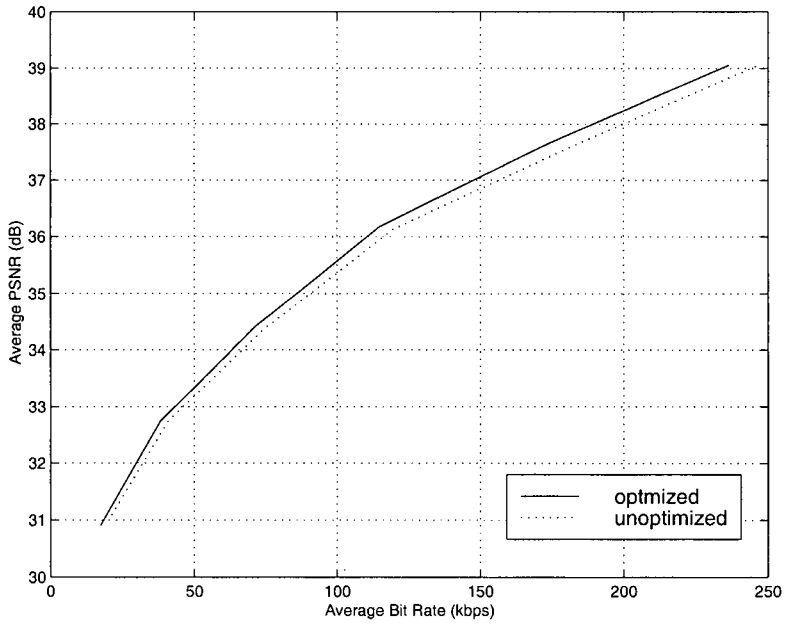
Not surprisingly, in the case of a layered encoding framework, the need for a rich set of coding parameters is further pronounced. For example, while the primary objective of enhancement layer data is to refine base layer data, repeatedly encoding the same error signal for base layer blocks is not optimal in a rate-distortion sense. In fact, this leads to over-coding in the enhancement layer for persistent static regions. Moreover, enhancement layer frames exhibit a similarly high degree of temporal correlation as their base layer counterparts. Therefore, a significant improvement in coding efficiency can be realized by incorporating temporal prediction in the enhancement layer, as the previous enhancement layer data offers better quality of reconstruction.

Recall from Section 2.3.2 that the source for prediction in the enhancement layer can be selected at the macroblock layer. This flexibility is well suited to the application of rate-distortion optimization techniques.

We now study the key technical features of layered video encoding, using H.263+. We illustrate the coding performance as features are added incrementally to a video encoder who operational mode is rate-distortion optimized. In all cases, the same encoder, that supports the full set of permissible coding modes, is employed in the base layer, thus base layer streams are identical. For simplicity, we restrict the evaluation to two layers. We illustrate encoding options as follows:



(a)



(b)

Figure 3.3: PSNR versus total bit rate, (a) FOREMAN and (b) COASTGUARD, QCIF, 10 fps, for the optimized and unoptimized encoders.

- *upward mode* only: Each macroblock is coded as a combination of the predicted macroblock from the corresponding reference layer frame along with DCT coding of the prediction error block.

- *upward* and *skipped mode*: A macroblock can be coded in upward mode or replaced by the macroblock at the same spatial location in the previously decoded enhancement layer frame.

- *upward*, *skipped* and *inter mode* with (0,0) motion vector: A macroblock can be coded in upward mode, skip mode, or as a combination of the predicted macroblock in the previously decoded enhancement layer frame, displaced by the (0,0) motion vector, along with DCT coding of the prediction error block.

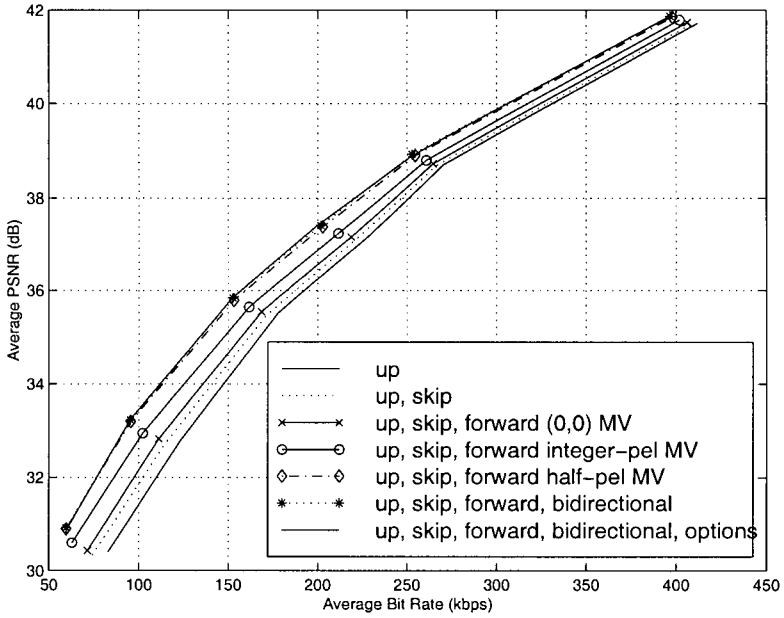
- *upward*, *skipped* and *inter mode* with integer-pel accuracy motion vector: A macroblock can be coded in upward mode, skip mode, or as a combination of the predicted macroblock in the previously decoded enhancement layer frame, displaced by an integer-pel accuracy motion vector, along with DCT coding of the prediction error block.

- *upward*, *skipped* and *inter mode* with half-pel accuracy motion vector: A macroblock can be coded in upward mode, skip mode, or as a combination of the predicted macroblock in the previously decoded enhancement layer frame, displaced by a half-pel accuracy motion vector, along with DCT coding of the prediction error block.

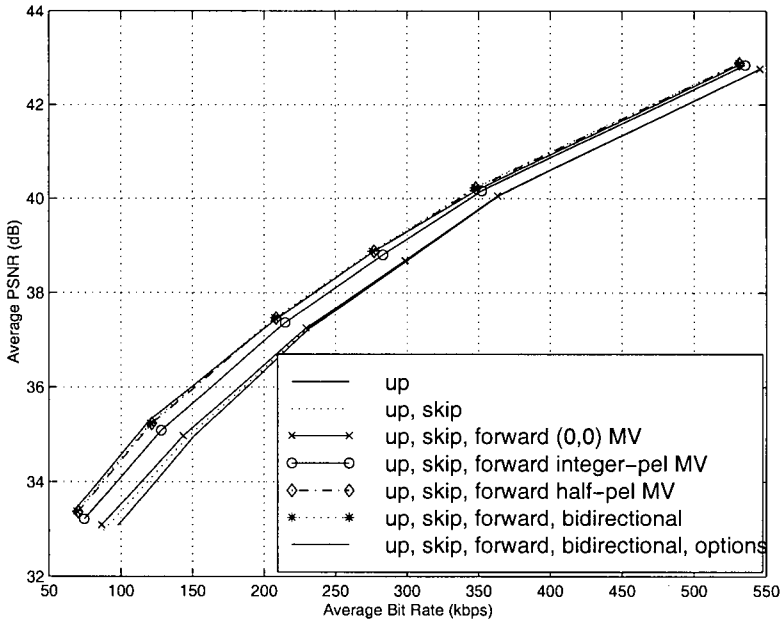
- *upward*, *skipped*, *inter* and *bi-directional mode* with half-pel accuracy motion vectors: A macroblock can be coded in upward mode, skip mode, inter mode, or as a combination of the average of the forward predicted macroblock from the previously decoded enhancement layer frame and the upward predicted macroblock from the corresponding reference layer frame, along with DCT coding of the prediction error block.

- *upward*, *skipped*, *inter* and *bi-directional mode* with half-pel accuracy motion vectors and all additional H.263 optional modes: This is the same as the previous coder. In addition H.263 Annexes D, F, I, J, S and T [1] as described in Section 2.2.2 are enabled in the enhancement layer coder.

Results for SNR scalability are illustrated in Figure 3.4 for two video sequences, FOREMAN and COASTGUARD. In all cases, the enhancement layer quantizer level is half that of the base layer. Results for spatial scalability are illustrated in Figure 3.5, for the same two video sequences. In all cases, the enhancement layer quantizer is identical to that of the base layer. Also, the spatial resolution of the enhancement layer is exactly twice that of the base layer. In all cases where motion vectors are permitted, an exhaustive search algorithm is employed. From the figures, it is clear that the rich set of available coding options significantly improves the efficiency of layered video encoding. We observe, for both SNR and spatial scalability, up to a factor of two increase in coding efficiency between the encoder that employs only the *upward mode* and the encoder that employs the full set of encoding options. It

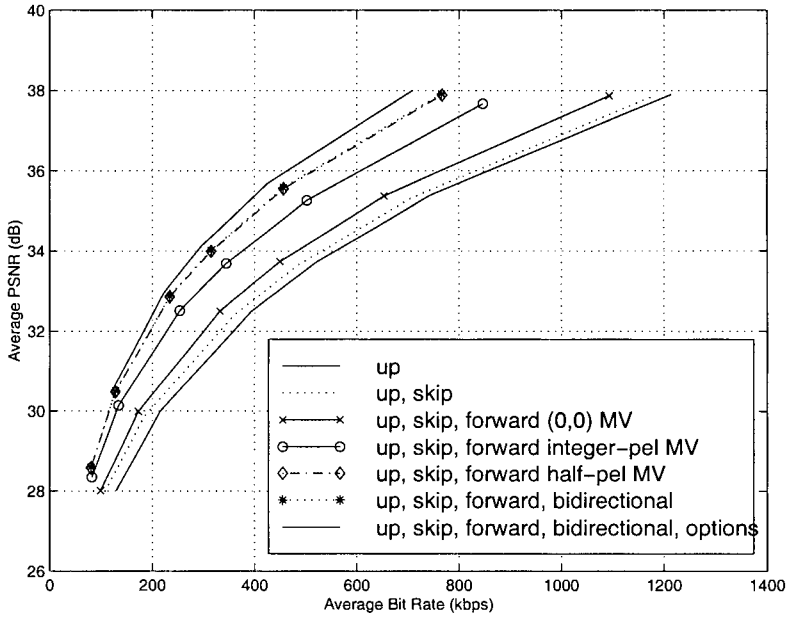


(a)

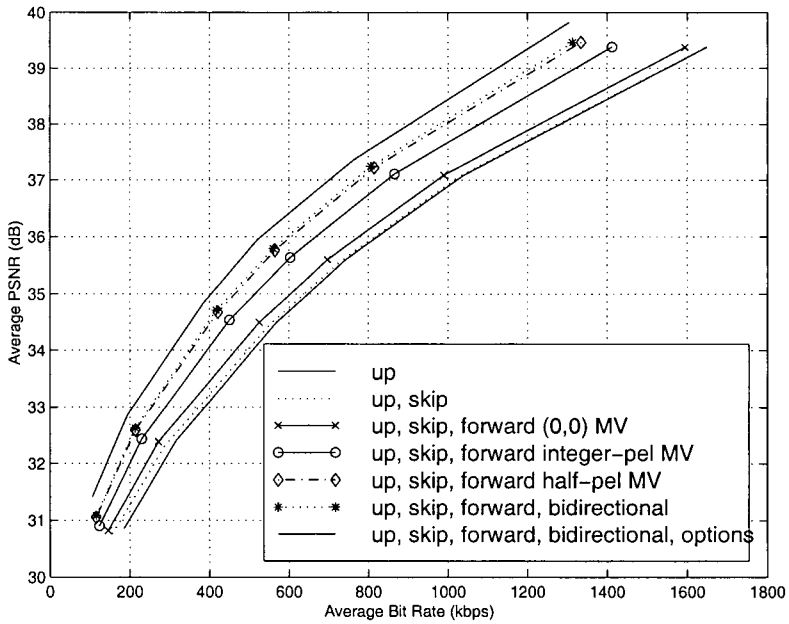


(b)

Figure 3.4: PSNR versus total bit rate, (a) FOREMAN and (b) COASTGUARD, base layer QCIF, 10 fps, enhancement layer QCIF, 10 fps, for the incremental addition of technical features into a layered coder, SNR scalability.



(a)

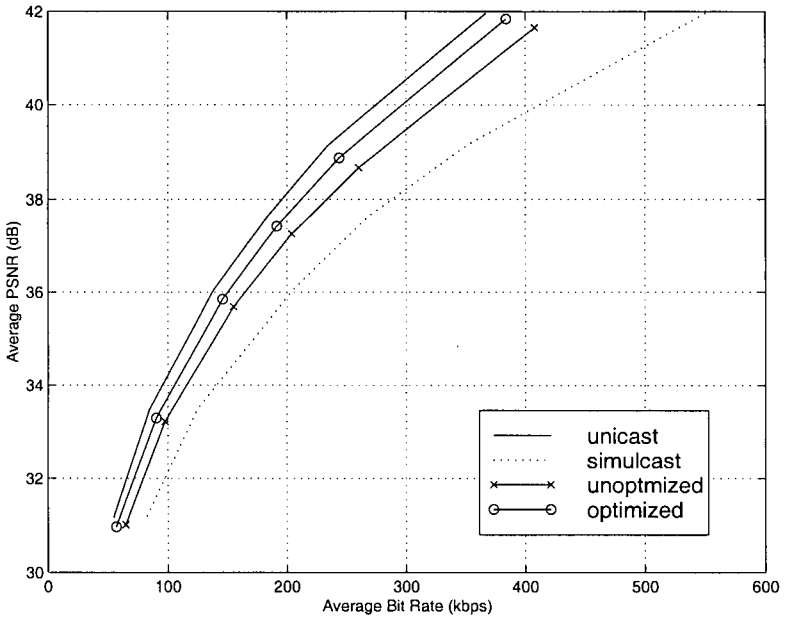


(b)

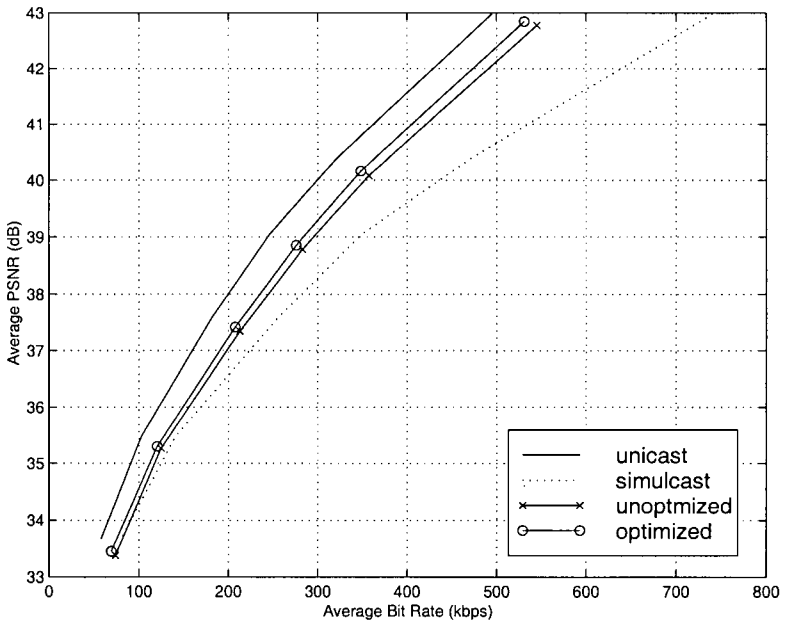
Figure 3.5: PSNR versus total bit rate, (a) FOREMAN and (b) COASTGUARD, base layer QCIF, 10 fps, enhancement layer CIF, 10 fps, for the incremental addition of technical features into a layered coder, spatial scalability.

is worth noting that algorithms that employ only the base layer reconstruction for prediction [29, 30, 31, 32] achieve coding efficiency similar to that of the encoder employing only the *upward mode* for prediction. From Figure 3.4 we see that for SNR scalability at high bit rates, little compression efficiency is sacrificed. However, as we have observed above, for SNR scalability (at low bit rates) and for spatial scalability, our rate-distortion optimized layered encoding algorithm provides up to a factor of two increase in coding efficiency compared to such algorithms.

Figure 3.6 illustrates how a less optimized selection of coding parameters, from the same available set, fails to encode the combined base and SNR enhancement layer as efficiently. Similar results are illustrated for spatial scalability in Figure 3.7. The operational mode of the less-sophisticated layered encoder is also based on thresholds [67]. The base layer is encoded using the procedure described above. The enhancement layer mode decision employs the minimum integer-pel SAD from enhancement layer motion estimation, where the (0,0) integer-pel motion vector is again reduced by 100 to bias the decision towards the *skipped mode*. This SAD is used as above, to determine whether or not to encode the macroblock using the *intra mode*. If the *inter mode* is selected, half-pel accuracy motion estimation is performed around the given integer-pel 16x16 motion vector. Then, the *upward mode* and the *bidirectional mode* are considered. The order of preference for these modes is *upward mode*, *inter mode*, and *bidirectional mode*. The SADs for prediction for these additional modes are computed. A motion vector of (0,0) is implicit

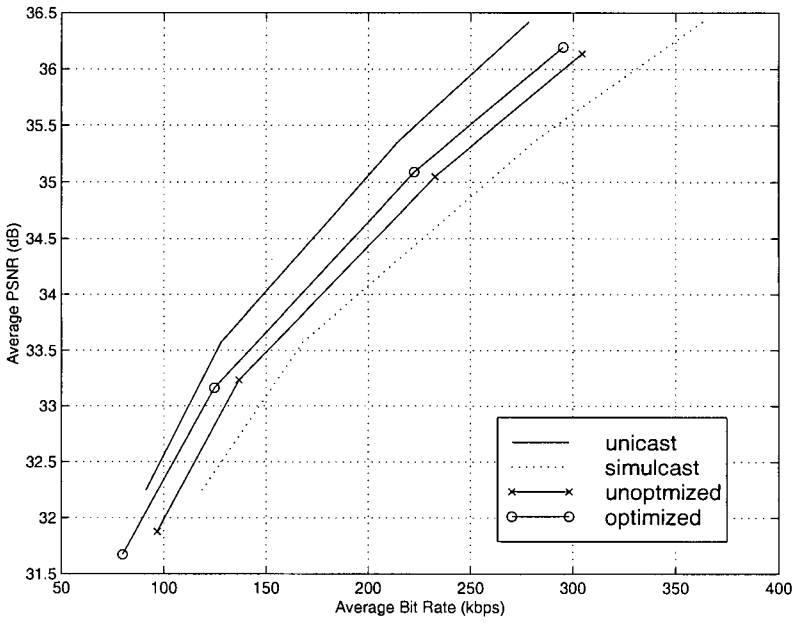


(a)

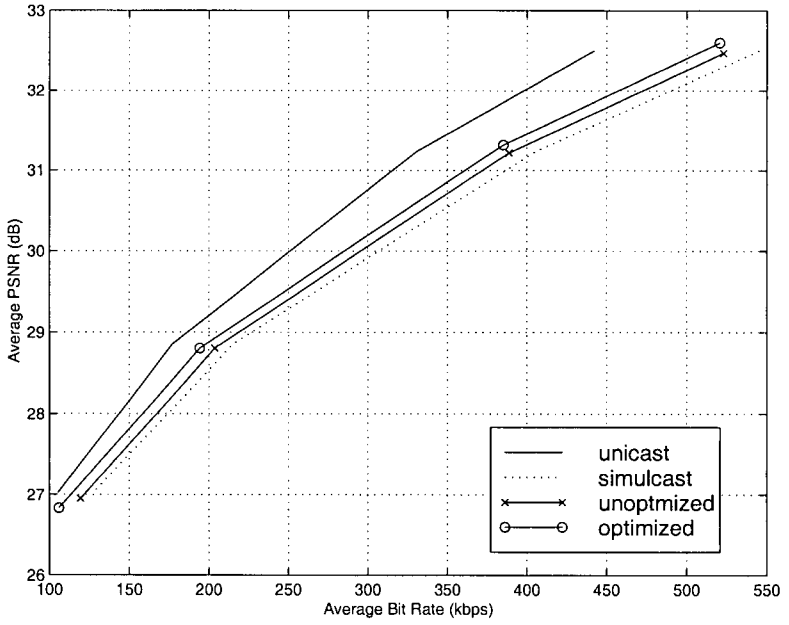


(b)

Figure 3.6: PSNR versus total bit rate, (a) FOREMAN and (b) COASTGUARD, base layer QCIF, 10 fps, enhancement layer QCIF, 10 fps, for unicast, simulcast, optimized and unoptimized SNR scalable coder.



(a)



(b)

Figure 3.7: PSNR versus total bit rate, (a) FOREMAN and (b) COASTGUARD, base layer QCIF, 10 fps, enhancement layer QCIF, 10 fps, for unicast, simulcast, optimized and unoptimized spatial scalable coder.

in the *upward mode*. The bidirectional mode employs a bilinear interpolation of the *upward mode* prediction and the *inter mode* prediction (using the given half-pel motion vector). To reflect the order of preference, the *upward mode* SAD is reduced by 50, the *inter mode* SAD is unchanged, and the *bidirectional mode* SAD is increased by 100. The mode the yields the minimum SAD is selected as the encoding mode for the macroblock. Again, if the *inter mode* is selected and the motion vectors and quantized DCT coefficients are all zero, the macroblock is encoded in *skipped mode*. From Figure 3.6, we can conclude that rate-distortion optimization significantly improves coding efficiency in a layered encoding framework.

In these figures, we have also illustrated the performance of unicast and simulcast encoding. For simulcast encoding, each representation (corresponding to each layer in the layered encoding framework) is encoded independently. As expected, layered encoding provides increased bandwidth efficiency relative to simulcast encoding. This is due to the reuse of reference layer information in the enhancement layer. Note that in some situations, layered encoding can result in decreased bandwidth efficiency relative to simulcast encoding [40]. This can occur when either too little or too much of the aggregate bit rate is devoted to the base layer. In the case of the former, the quality of the base layer is usually too low for the information to be useful in the enhancement layer. In the case of the latter, there is insufficient bit rate remaining for the enhancement layer to produce a good quality representation.

For unicast encoding, only the highest resolution representation (corresponding to the top-most enhancement layer in the layered framework) is encoded. From Figures 3.6 and 3.7 it is evident that there is a clear decrease in bandwidth efficiency for layered encoding. This decreased efficiency is due to the less efficient encoding of both the overhead and data elements in the enhancement layer data stream [28, 41]. Thus, the upper bound on the rate-distortion performance for layered encoding is the rate-distortion performance for non-layered encoding of the topmost enhancement layer. This makes sense given that the layered encoding framework we employ does not produce a fully embedded representation.

3.2 Algorithm

We now present the general formulation for our operational rate-distortion optimized encoding algorithm. Our goal is to select the best macroblock encoding parameters, in a rate-distortion sense, including the motion vector, quantization step size, and encoding mode. Thus, for given block b in layer l of frame k, we select the parameters that minimize the Lagrangian as follows: [46].

$$J(b,l,k) = D(b,l,k) + \lambda(l,k)R(b,l,k). \tag{3.4}$$

We choose the Lagrangian rate-distortion functional as it provides an elegant framework for determining the optimal choice of motion vectors and prediction modes by weighting a distortion term against a resulting rate term

Coding Mode	COD	Quantizer	Motion Vector	DCT	Side Information
skipped	1	n/a	n/a	n/a	none
inter	0	n/a	MVD	residual	CBP
interq	0	DQUANT	MVD	residual	CBP
inter4v	0	n/a	MVD4	residual	CBP
inter4vq	0	DQUANT	MVD4	residual	CBP
intra	0	n/a	n/a	intra	CBP
intraq	0	DQUANT	n/a	intra	CBP

Table 3.2: Parameters for permissible coding modes for H.263 P-picture macroblocks.

for a particular choice of coding parameters. Here, D is defined as some distortion measure, typically the sum of absolute error (SAE) or sum of squared error (SSE). R is defined as some rate measure, typically the resulting rate to encode the macroblock for a particular choice of coding parameters.

Table 3.2 outlines the set of coding parameters for 3.4 for H.263 P-picture macroblocks [1]. Similarly, Table 3.3 outlines the set of coding parameters for 3.4 for H.263 EP-picture macroblocks [1].

If a macroblock is not coded, i.e. coded in the *skipped mode*, the COD parameter is set to 1, no further information is required, and the macroblock is replaced by the macroblock at the same spatial location in the previously decoded picture. This mode works well for image regions where there is little or no change relative to the previously decoded picture. In the *inter* and *interq mode*, one motion vector is transmitted (MVD), along with the intra coded prediction error (residual) blocks. The difference is that in *interq mode*, the value of the quantizer is also changed at the macroblock level (DQUANT).

This requires two additional signaling bits and is useful to compensate for prediction inaccuracies. In the *inter4v* and *inter4vq mode*, four motion vectors can be transmitted (MVD4) along with the prediction error blocks. This mode is useful for image regions with high motion activity. For image regions exhibiting non-motion activity, such as camera noise, occlusion, camera zoom and illumination changes or complex non-translational motion such as rotation, coding the macroblock content directly in the *intra mode* (intra), i.e. without prediction, can be more productive, thus the *intra* and *intraq mode* are beneficial. For all modes, except the *skipped mode*, side information must be provided to indicate which of the blocks contain non-zero DCT coded content, i.e. the coded block pattern (CBP).

From Table 3.2 we see that seven Lagrangian values must be computed per macroblock to determine the encoding mode that yields the lowest Lagrangian cost. In fact, this is further complicated as the *inter* and *inter4v mode* involve a joint optimization between each candidate motion vector and the resulting DCT coded prediction error block. In Chapter 4 we will show that the complexity of this task is prohibitive. To reduce this complexity, we decouple the motion estimation and mode decision process. For an algorithm that considers joint optimization of motion and prediction error block encoding, refer to [58, 59]. In our algorithm, we first select the motion vector that yields a minimum motion Lagrangian cost

$$J_{motion}(b,l,k) = D_{motion}(b,l,k) + \lambda_{motion}(l,k)R_{motion}(b,l,k). \tag{3.5}$$

Then, using the obtained motion vector, the optimal coding mode is selected

Coding Mode	COD	Quantizer	Motion Vector	DCT	Side Information
skipped	1	n/a	n/a	n/a	none
inter-upward	0	n/a	none	residual	CBP+MBTYPE
interq-upward	0	DQUANT	none	residual	CBP+MBTYPE
inter-forward	0	n/a	MVD	residual	CBP+MBTYPE
interq-forward	0	DQUANT	MVD	residual	CBP+MBTYPE
inter-bidir	0	n/a	MVD	residual	CBP+MBTYPE
interq-bidir	0	DQUANT	MVD	residual	CBP+MBTYPE
intra	0	n/a	n/a	intra	CBP+MBTYPE
intraq	0	DQUANT	n/a	intra	CBP+MBTYPE

Table 3.3: Parameters for permissible coding modes for H.263 EP-picture macroblocks.

by minimizing the Lagrangian

$$J_{mode}(b,l,k) = D_{mode}(b,l,k) + \lambda_{mode}(l,k)R_{mode}(b,l,k). \tag{3.6}$$

As part of the mode selection, the permissible quantizer values are considered. In this sense, the *inter mode* is essentially a sub-mode of the *interq mode*, for which the change in quantizer value relative to the previous macroblock is set to zero.

From Table 3.3, we see that nine Lagrangian values must be computed per-macroblock to determine the encoding mode that yields the lowest Lagrangian cost for enhancement layer pictures. In addition to the dependence between each candidate motion vector and the DCT coding of the prediction error block, another complicating factor is the dependence between encoding decisions made in each layer. The rate-distortion performance of a given enhancement layer depends on that of its reference layer. To reduce complexity, we decouple the optimization process to be performed individually for each

layer. This simplification still yields a locally optimal solution, as discussed in Chapter 2.

Returning to Figures 3.3, 3.6 and 3.7, we assess the performance gains that can be obtained by the rate-distortion optimized coding algorithm. From Figure 3.3 we see that the performance improvement achievable by rate-distortion optimization for a single layer is approximately 0.5 dB, or a 10 % reduction in bit rate. From Figure 3.6 we see that the performance of the layered, SNR scalable, coder is 0.5 - 1.0 dB lower than for unicast but up to 2.0 dB higher than for simulcast. Moreover, comparing the performance of the optimized and unoptimized layered coders, we see that the performance improvement is again up to 0.5 dB. From Figure 3.7 we see that the performance of the layered, spatially scalable, coder is 0.5 - 1.0 dB lower than for unicast. However, the performance is up to 0.75 dB higher than simulcast. Moreover, comparing the performance of the optimized and unoptimized layered coders, we see that the improvement is again up to 0.5 dB.

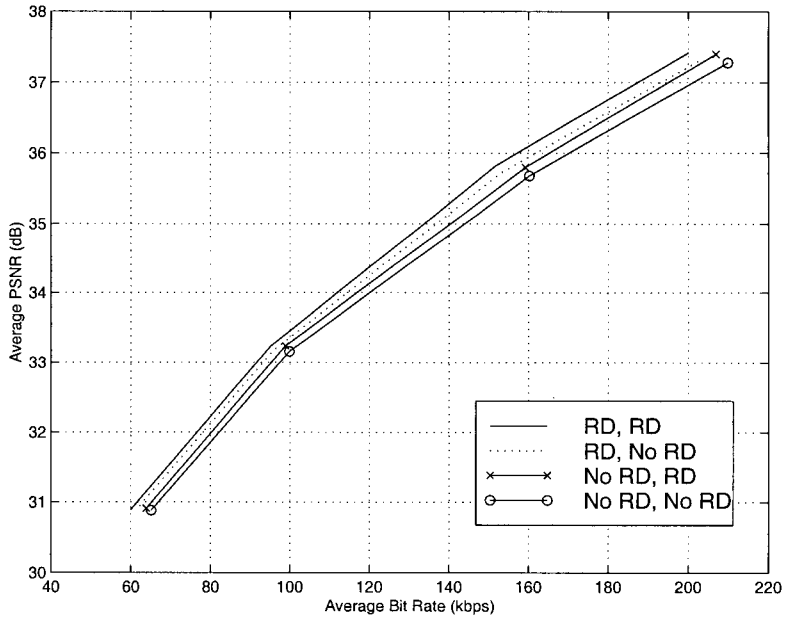
We can also observe that the rate-distortion performance gains are more pronounced for higher activity sequences. As the FOREMAN sequence contains high motion, camera motion, and occlusions, a significant proportion of P-picture macroblocks are coded in the *intra mode* in the unicast encoder, which encodes the sequence at CIF resolution. In the layered encoder, most of the *intra mode* coding is performed in the base layer. Therefore, blocks that are encoded in the *intra mode* by the unicast encoder can be, in the enhancement layer pictures of the layered encoder, predicted from the corresponding base

layer reconstruction. Still, as expected, none of the layered encoders can quite achieve the performance of the unicast encoders as, in addition to the inherent inefficiencies in the layering framework, rate-distortion optimization in the unicast encoder significantly reduces the number of macroblocks that are coded as intra.

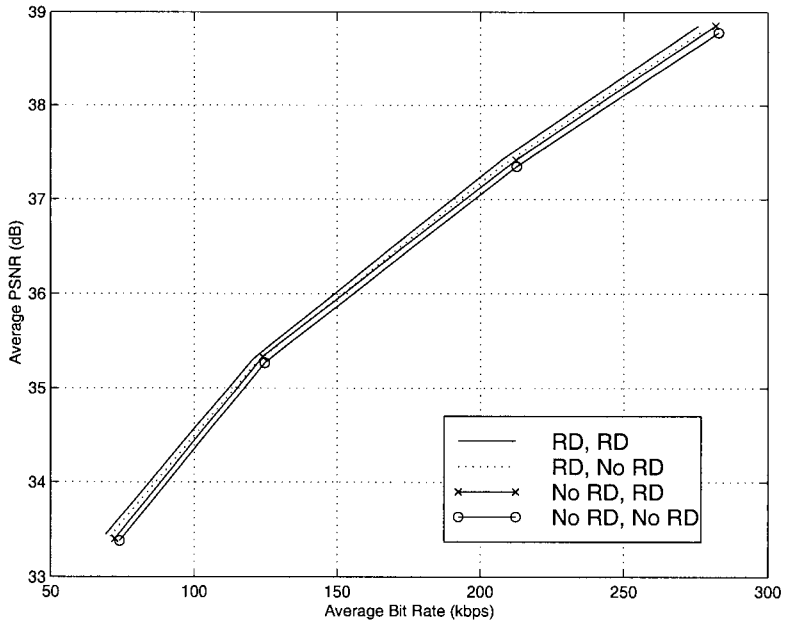
In Figures 3.8 and 3.9, we illustrate the rate-distortion performance of four layered encoders, for SNR and spatial scalability, respectively, that employ encoding algorithms in the base and enhancement layers as follows:

- Base layer and enhancement layer not optimized
- Base layer optimized, enhancement layer not optimized
- Base layer not optimized, enhancement layer optimized
- Base layer and enhancement layer optimized

Of interest for SNR scalability is the observation that, in Figure 3.8, rate-distortion optimization in the base layer alone provides more gains, in terms of rate-distortion performance, than rate-distortion optimization in the enhancement layer alone. This is due to the fact that rate-distortion optimization in the base layer significantly reduces the amount of macroblocks encoded in the *intra mode*, which are the most expensive in terms of bits. On the other hand, in the enhancement layer, although the *intra mode* is a possible encoding mode, it is rarely used. This basically eliminates the potential for rate-distortion optimization in the enhancement layer to produce as significant savings as realized in the base layer.

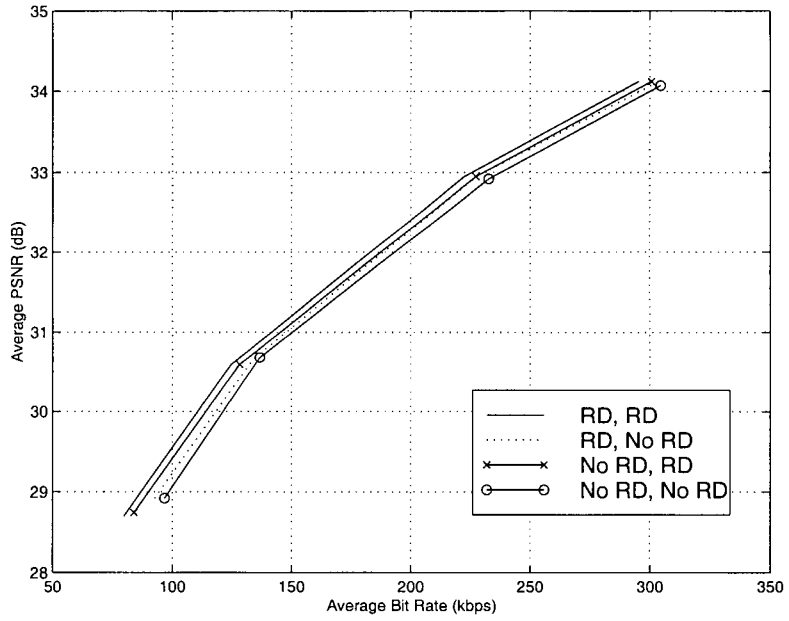


(a)

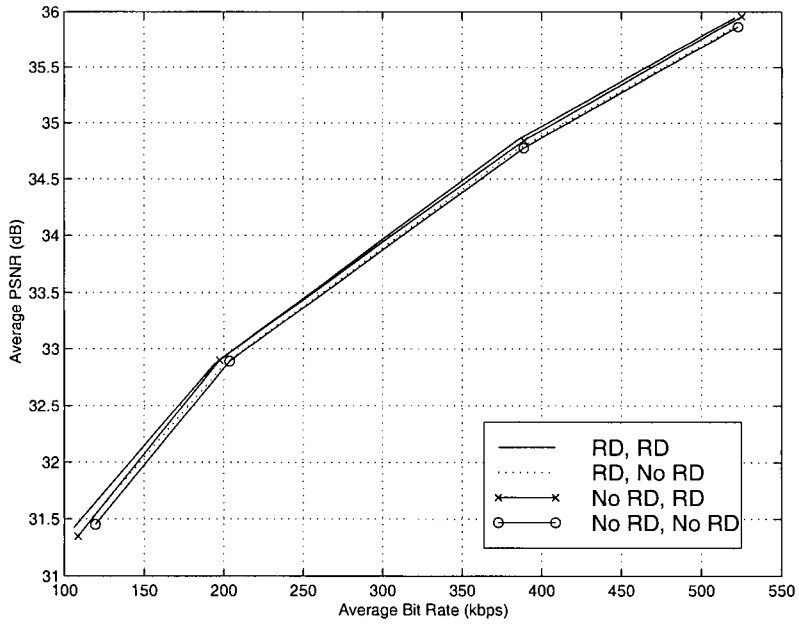


(b)

Figure 3.8: PSNR versus total bit rate, (a) FOREMAN and (b) COASTGUARD, QCIF, 10 fps, for different combinations of optimization applied to the base and enhancement layers, SNR scalability.



(a)



(b)

Figure 3.9: PSNR versus total bit rate, (a) FOREMAN and (b) COASTGUARD, QCIF and CIF, 10 fps, for different combinations of optimization applied to the base and enhancement layers, spatial scalability.

In the case of spatial scalability, illustrated in Figure 3.9, we observe that rate-distortion optimization in the base layer alone provides similar gains, in terms of rate-distortion performance, as rate-distortion optimization in the enhancement layer alone. This is due to the fact that, while rate-distortion optimization in the base layer significantly reduces the amount of macroblocks encoded in the *intra mode*, rate-distortion optimization in the enhancement layer operates on pictures having higher spatial resolution. This results in high coding efficiency for both the base and enhancement layers.

We can also observe that the overall improvement in rate-distortion performance is not simply the sum of the improvements in the individual layers. Rather, the rate-distortion improvements achieved in the base layer limit somewhat the gains achievable by optimization in the enhancement layer.

3.3 Overhead Elements

Our goal has been to improve coding efficiency in a layered encoding framework. We have shown that our rate-distortion optimization algorithm can provide up to 10% reduction in bit rate for the same picture quality. We have also stated that there are inherent inefficiencies in the layering framework that limit the potential for further gains. These inefficiencies are due to the following:

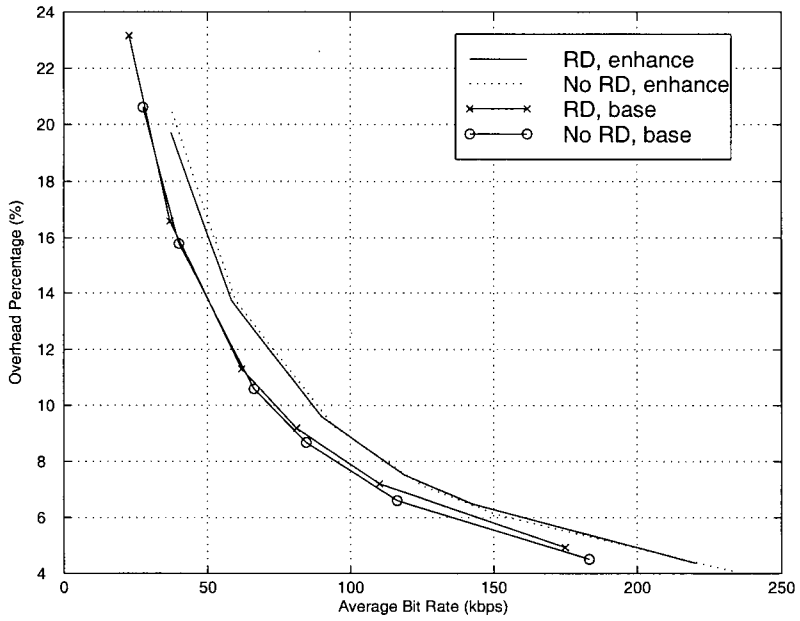
- Inefficient signaling of overhead elements (control information).

- Differing statistics of the enhancement layer (band-pass) error signal, for which the source model is not as well suited.

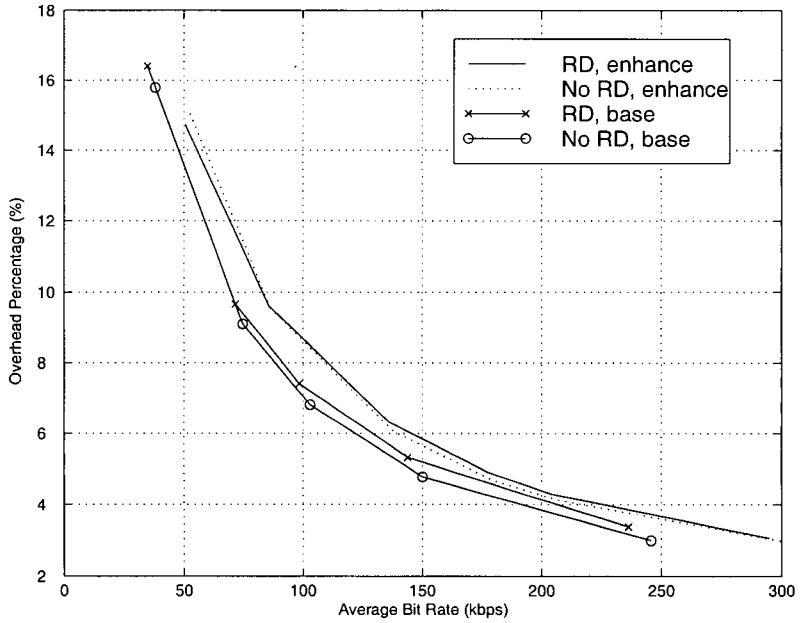
While we do not propose to improve the coding efficiency of these overhead elements, we analyze these inefficiencies in detail.

Expressing a layered structure introduces additional complexity to the syntax of the data stream. When we discuss overhead elements that decrease coding efficiency, we refer to fields such as picture headers and macroblock headers, including the COD field (skipped or not), the macroblock type field (MBTYPE), the coded block pattern field (CBP), and differential quantizer value (DQUANT). While these overhead fields are critical and convey important information to a decoder, they do not directly result in the reconstruction of non-zero pixels that can increase picture quality. Thus, the encoding of overhead elements should be as efficient as possible. However, many of these fields tend to require a fixed number of bits, independent of the target bit rate, effectively resulting in reduced overhead coding efficiency at lower bit rates.

This effect is illustrated in Figures 3.10 and 3.11, for two sequences, for SNR and spatial scalability, respectively. Moreover, we illustrate this effect for both the optimized and unoptimized encoders. A separate curve is plotted for each layer of each encoder. Most noticeable is the increase in overhead percentage at lower bit rates, as described above. This occurs in the base and enhancement layers of the optimized and unoptimized encoders. Also, we see that overhead coding is consistently much less efficient for the enhancement layer data streams, especially at low bit rates. Such a high overhead percentage

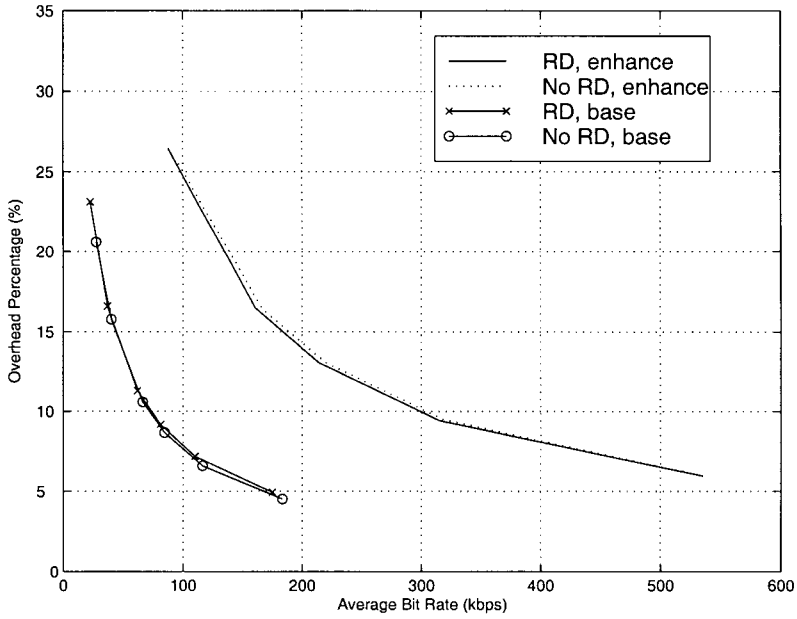


(a)

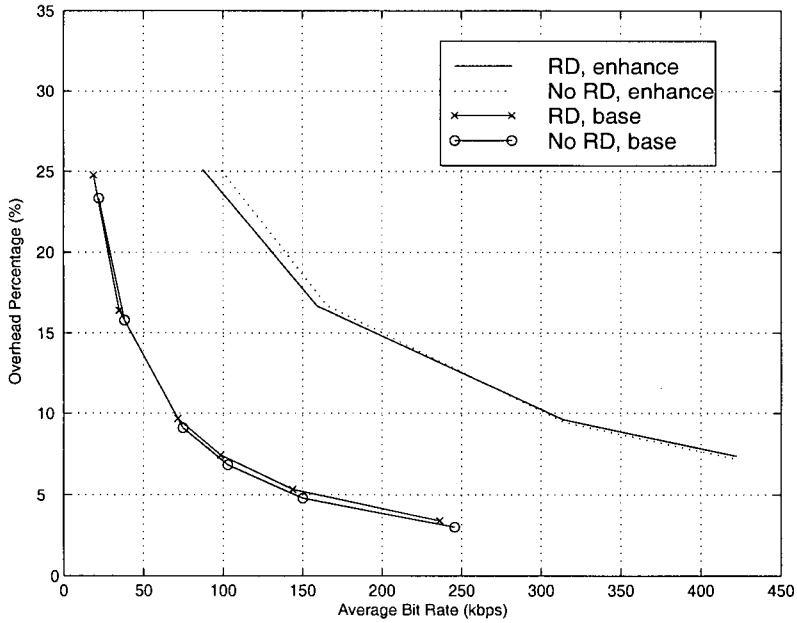


(b)

Figure 3.10: Overhead percentage versus total bit rate, (a) FOREMAN and (b) COASTGUARD, QCIF, 10 fps, for the base and enhancement layer data streams, both optimized and unoptimized, SNR scalability.



(a)



(b)

Figure 3.11: Overhead percentage versus total bit rate, (a) FOREMAN and (b) COASTGUARD, QCIF and CIF, 10 fps, for the base and enhancement layer data streams, both optimized and unoptimized, spatial scalability.

prevents bits from being allocated to the coding of data elements. For example, a single layer, 100 kbps bit stream for the sequence FOREMAN would include 8% overhead, or 8 kbps. For a two-layered, SNR scalable bit stream of the same sequence, with a 25 kbps - 75 kbps bit rate allocation between the base and enhancement layers, the resulting overhead would be 14% (.2 $\times$ 25 + .12 $\times$ 75), or 14 kbps. For spatial scalability, the situation is worse, with a resulting overhead of 25 % (.2 $\times$ 25 + .27 $\times$ 75), or 25 kbps.

In [41], the syntax of H.263 scalability [1] is modified to increase the coding efficiency of overhead elements. First, where it is appropriate, they re-group the various overhead fields into combined tables. They also eliminate several MBTYPE code-words permitted by the syntax. While this produces a less flexible syntax, it produces more efficient MBTYPE code-words. Finally, for the new groupings, they create multiple tables. The table to be used is specified at the picture layer. The new tables are designed to exploit instances where one MBTYPE is predominant within a picture. They report that the modifications produce a consistent increase in rate-distortion performance of 0.5 dB for SNR scalability at low bit rates.

3.4 Rate Allocation Tradeoffs

We have established that providing a layered representation of video, rather than independently simulcasting multiple representations, provides bandwidth savings. For a layered representation, typical end-user connections will generally dictate the bit rate allocation among the individual layers. Still, it is

Connection Type	Connection Bit Rate
Modem	14.4 - 56 kbps
ISDN	56 - 112 kbps
DSL/Cable	256 - 512 kbps
LAN	$\geq$ 1 Mbps

Table 3.4: Types of end-user Internet connections and associated bit rates.

worth investigating how video quality is affected for different partitions of the total video bit rate between the base and enhancement layers. In Table 3.4 we outline possible end-user connections and the associated connection rate.

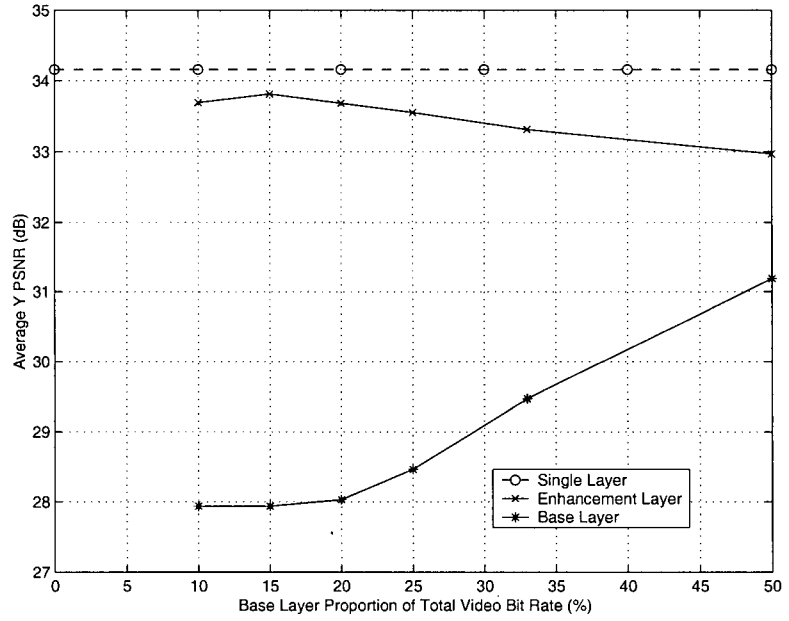
We now present results for the effects of the bit rate partition, for SNR and spatial scalability. For theses results, we select nominal total video bit rates well within the digital subscriber line (DSL) and Cable modem range outlined in Table 3.4. For SNR scalability, results are illustrated in Figure 3.12. For these simulations, the base and enhancement layer have CIF resolution and the total video bit rate is 256 kbps. All figures include curves for the average PSNR for the base and enhancement layer respectively. In all cases, the PSNR for a non-layered representation is also included. From Figure 3.12, we see that the base layer quality improves substantially as it is allocated an increasing proportion of the total video bit rate. Meanwhile, the enhancement layer quality only degrades slowly. Therefore, for SNR scalability, it is reasonable to allocate 25% or more of the total bit rate to the base layer when the total video bit rate is within the DSL/Cable range. As a larger proportion of the video bit rate is allocated to the base layer, however, the difference in quality between the base and enhancement layer becomes less noticeable. For such an

allocation, purely in terms of reconstructed video quality, this could make the need to receive enhancement layers questionable from the perspective of the end-user. Clearly, there are still inherent error resilience benefits.

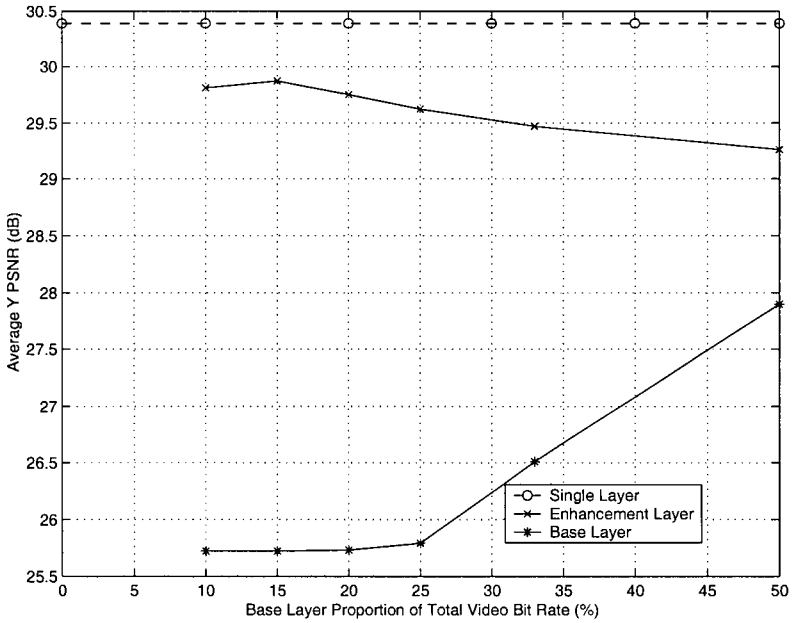
For spatial scalability, results are illustrated in 3.13. For these simulations, the base layer has QCIF resolution and the enhancement layer has CIF resolution, and the total video bit rate is 396 kbps. Again, all figures include curves for the average PSNR for the base and enhancement layer respectively. In all cases, the PSNR for a non-layered representation is also included. From Figure 3.13, we see that the base layer quality increases even more dramatically than for SNR scalability as it is allocated an increasing proportion of the total video bit rate. However, it is important to remember that this the base layer has QCIF resolution while the enhancement layer has CIF resolution. Therefore, for spatial scalability, it appears reasonable to allocate as little as 10-20 % of the total video bit rate to the base layer when the total video bit rate is in the DSL/Cable range. Allocating much more than 25% of the total video bit rate to the base layer results in the base layer quality far surpassing the enhancement layer quality. For such an allocation, it could be considered wasteful to so increase the quality of a QCIF representation at the expense of the CIF resolution enhancement layer.

3.5 Conclusions

In this chapter we studied the effectiveness of the key technical features of a layered video encoding algorithm. One valuable contribution was the deter-

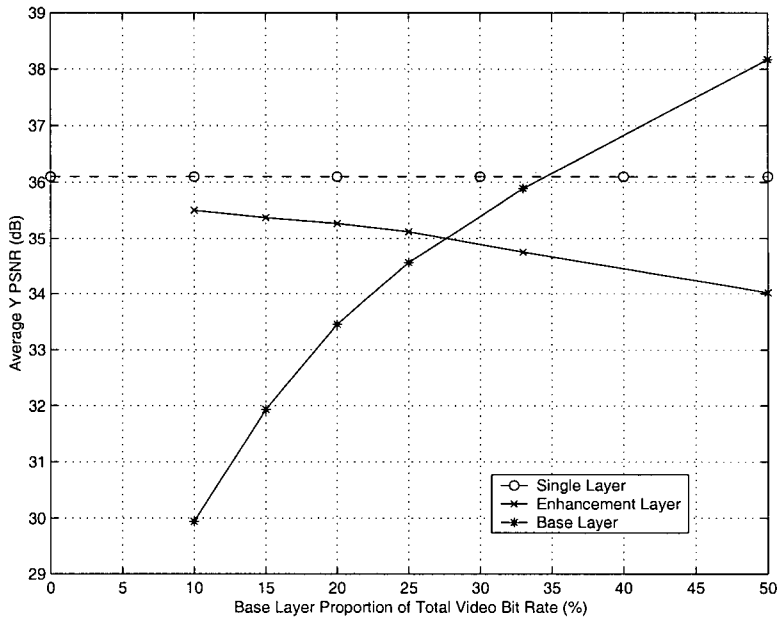


(a)

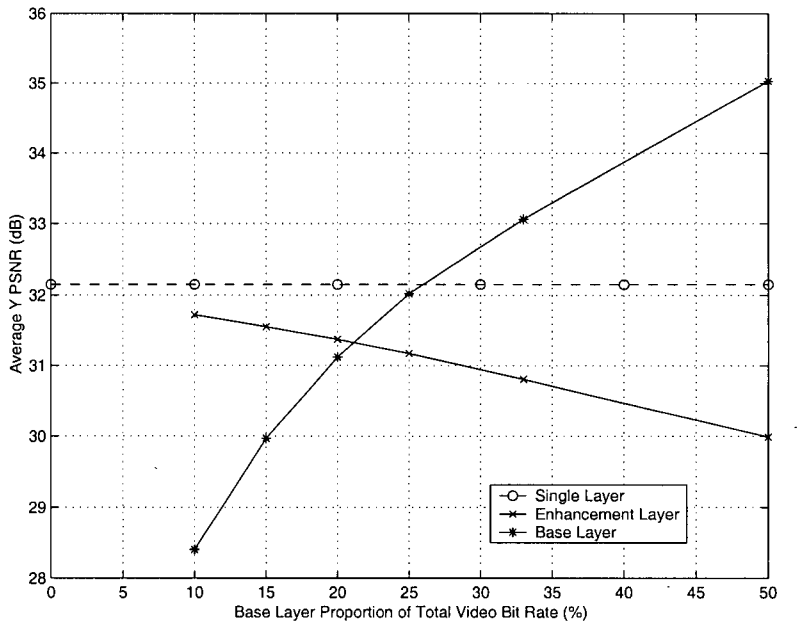


(b)

Figure 3.12: PSNR versus base layer percentage of total video bit rate (256 kbps) for the base and enhancement layers (a) FOREMAN and (b) COAST-GUARD. For SNR scalability, the base layer and enhancement layer resolution is CIF.



(a)



(b)

Figure 3.13: PSNR versus base layer percentage of total video bit rate (396 kbps) for the base and enhancement layers (a) FOREMAN and (b) COAST-GUARD. For Spatial scalability, the base layer resolution is QCIF and the enhancement layer resolution is CIF.

mination of the upper bound on rate-distortion performance for layered video encoding in error-free environments. We found this upper bound was the rate-distortion performance for non-layered encoding of the topmost enhancement layer, using the same system parameters. We then introduced a rate-distortion optimized layered video encoding algorithm for error-free environments. This algorithm was the main contribution of this chapter. The algorithm was demonstrated to achieve a significant improvement in rate-distortion performance. Moreover, we made some important observations. First, we observed that the overall improvement in rate-distortion performance was not simply the sum of the improvements in the individual layers. Rather, the rate-distortion improvements achieved in the base layer limited the gains achievable in the enhancement layer. Furthermore, we showed that a significant deficiency with the layered coding framework supported by H.263+ is the inefficiency in the coding overhead elements. More efficient coding of these elements was shown to yield a consistent a substantial improvement in rate-distortion performance. Finally, we showed the effects of the allocation of the total video bit rate between the base and enhancement layers. We showed that, for a total video bit rate within the DSL/Cable range, allocating upwards of 25% of the total video bit rate to the base layer provided reasonable quality, for both the base and enhancement layer, for SNR scalability. We also showed that allocating up to 20% of the total video bit rate to the base layer provided reasonable quality, for both the base and enhancement layer, for spatial scalability.

Chapter 4

Complexity Issues

In this chapter, we address complexity issues of the algorithm introduced in Chapter 3. One main goal is to find good tradeoffs between rate-distortion performance and computational complexity. We first motivate the need to make simplifications to the operational rate-distortion optimization framework. Several simplifications are then proposed. We re-formulate the minimization to select locally optimal solutions rather than a globally optimal solution. We also decouple the motion estimation from the mode decision, resulting in a two-stage optimization. We then perform a complexity analysis of the proposed algorithm. To select the parameter that controls the encoder's rate-distortion trade-offs, we propose a model to control the operating mode of the layered video encoder. This model permits the encoder to compute *a priori* the rate-distortion optimized parameters such that a target bit rate can be achieved. It is the main contribution of this chapter.

4.1 Analysis and Preliminary Simplifications

In this section, we perform a complexity analysis of the operational rate-distortion optimization algorithm and present some preliminary, albeit necessary, simplifications.

For the purpose of our analysis, we consider the set of permissible coding parameters outlined in Tables 3.2 and 3.3, excluding the *inter4v mode* and the *inter4vq mode*. We define M_i as the set of permissible motion vectors and Q_i as the set of permissible quantizers for macroblock i. In a practical system both of these sets are finite. We then define the combination of permissible parameters to be jointly optimized as $P_i = \{M_i, Q_i\}$ and one pair from this set $p_i = (m_i, q_i)$, where $m_i \in M_i$ and $q_i \in Q_i$. We therefore obtain

$$|P_i^{skip}| = 1$$

$$|P_i^{intra}| = |Q_i|$$

$$|P_i^{inter}| = |M_i \times Q_i|$$

where $|\cdot|$ denotes cardinality. Note that, by including Q_i, P_i^{intra} accounts for the *intra mode* and *intraq mode* and P_i^{inter} accounts for the *inter mode* and *interq mode* of Tables 3.2 and 3.3 respectively.

The Lagrangian we wish to minimize, over a frame containing N macroblocks, can be re-defined as

$$J = \min_{P_i \, \forall i} \sum_{i=1}^{N} \left\{ D^i + \lambda R^i \right\}. \tag{4.1}$$

Here D^i and R^i denote the distortion and resulting rate, respectively, for macroblock i. These values depend on the choice of coding parameters $p_i \in P_i$,

including motion vector, coding mode and quantizer level. Therefore, for each possible p_i, the prediction error block must be obtained, encoded and decoded to compute the corresponding Lagrangian cost. In theory, given λ, the value of Equation (4.1) can be minimized by computing exhaustively the values for all possible combinations of these parameters, for all the macroblocks. While this is already computationally prohibitive, the situation is actually worse. In most practical coding frameworks there is a rate dependency between neighboring macroblocks, as parameters such as motion vectors and quantizer levels are often differentially encoded. Thus, this would require computing

$$|P_i^{skip} + P_i^{inter} + P_i^{intra}|^N = |Q_i \times (M_i + 1) + 1|^N$$

different costs. Generally, for MC-DCT video coding, $M_i >> Q_i$. For example, for the picture resolutions we consider, a +/- 32 integer pel motion vector range is permitted for motion estimation, resulting in 65 $\times$ 65 candidate integer pel motion vectors, and 4 times as many candidate half pel motion vectors. In addition, there are 31 permitted quantizer level per macroblock each for the *interq mode* and the *intraq mode*. This means computing

$$|31 \times (84500 + 1) + 1|^N$$

different costs. We observe that, in practice, this term is dominated by M_i and the resulting complexity is astronomical. To reduce computations, we reformulate the exhaustive minimization as a cascade of local minimizations, as follows

$$J = \sum_{i=1}^{N} \min_{P_i} \left\{ D^i + \lambda R^i \right\}. \tag{4.2}$$

This simplification does yield sub-optimal results as the future implications arising from the motion vector and quantizer level rate dependencies are ignored. This is not to say that these rate dependencies are completely omitted. The algorithm still accounts for them, but only as parameters that have been determined *a priori*. Consequently, the performance loss is small [59]. Such a re-formulation requires computing

$$|P_i^{skip} + P_i^{inter} + P_i^{intra}| \times N = |Q_i \times (M_i + 1) + 1| \times N$$

different costs. This term is also dominated by M_i and is still computationally prohibitive. To further reduce computations, another simplification is to decouple the motion estimation and mode decision into two sequential stages. In the first stage, the locally optimal motion vector is determined for macroblock i. Effectively, the decrease in distortion and resulting increase in rate due to the encoding of the prediction error block are ignored. In the second stage, the locally optimal coding mode and quantizer level are determined for macroblock i. This is accomplished by computing the cost of encoding the macroblock using permissible coding modes and quantizer levels which, when applicable (i.e. for the *interq mode*), encode the prediction error block resulting from the already determined motion vector. While this is also sub-optimal, the performance loss may be negligible. This is because, while the locally optimal motion vector can be determined via traditional block-matching algorithms, based on minimizing a distortion term only, minimizing a Lagrangian cost for block-matching can maintain the performance level of jointly optimizing coding mode, motion vector and quantizer level choices [55]. At low bit rates, the

Identifier	Sequence	Resolution	Bit Rate
1	FOREMAN	QCIF	72000
2	COASTGUARD	QCIF	72000
3	FOREMAN	CIF	396000
4	COASTGUARD	CIF	396000

Table 4.1: Non-layered test scenarios for profiling the encoding runs.

Identifier	Sequence	Scalability	Resolution	Base Bit Rate	Enhancement Bit Rate
5	FOREMAN	SNR	QCIF/QCIF	24000	48000
6	COASTGUARD	SNR	QCIF/QCIF	24000	48000
7	FOREMAN	Spatial	QCIF/CIF	48000	348000
8	COASTGUARD	Spatial	QCIF/CIF	48000	348000

Table 4.2: Layered test scenarios for profiling the encoding runs.

accuracy of the motion compensation is the dominant performance factor [12].

Generally, the complexity of video encoding is highly non-deterministic. The number of computations depends on the scene content, image resolution and target bit rate. Therefore, we do not perform a theoretical complexity analysis. Rather, we instrument and analyze our software for actual encoding runs. Using an instruction level profiler, *iprof* [102], which is commonly used within the MPEG research community for measuring complexity, we can perform the necessary statistical analysis. We perform this analysis for both non-layered and layered scenarios, where appropriate. We use the test scenarios in Tables 4.1 and 4.2.

Results are presented in Table 4.3 for the scenarios of Tables 4.1 and 4.2, using the operational rate-distortion optimization algorithm that incor-

Identifier	Total Instructions (millions)
1	54648
2	62652
3	144323
4	280240
5	89267
6	92878
7	133943
8	172417

Table 4.3: Total instructions (in millions) for the test scenarios.

porates the simplifications described above. As expected, we observe that the complexity increases approximately linearly with spatial resolution. We also see that the complexity is somewhat sequence dependent. SNR scalability, while effectively encoding two frames for every one frame encoded in the single layer scenario, requires only approximately 1.5 times the complexity. Interestingly, spatial scalability, which effectively encodes one QCIF and one CIF frame for every frame encoded in the single layer CIF scenarios, requires fewer computations than the single layer CIF scenarios. Note that, for these results, a non-deterministic number of intermediate encodings are required for each frame, as the bisection search attempts to adjust λ_{mode} until the target bit rate can be closely matched. In this case, it cannot be guaranteed that each layer of each scenario requires the same number of iterations per frame. This is the source of disparity in the complexity measures.

4.2 Choice of Lagrangian Parameter

The output bit rate of the video encoder is determined by the particular choice of coding parameters. Of all the parameters, the quantizer level is typically the most important for controlling the output bit rate. It can be made more fine, to compensate for motion estimation inaccuracies. This has the effect of increasing the image quality and the output bit rate. It can be made more coarse, to effectively allocate a larger portion of the bit rate to motion vectors since, at low bit rates, motion compensation is the dominant performance factor. This has the effect of reducing the image quality and the output bit rate. This implies that there is a close relationship between the quantizer level and the desired rate-distortion tradeoffs, i.e. λ.

Ultimately, in the operational rate-distortion optimization framework, we need to employ a value of λ that allows a target output bit rate to be closely matched. However, for a given value of λ, the resulting output bit rate cannot be known *a priori*. Several approaches for finding a suitable value for λ are discussed in Section 2.4. The most obvious of these approaches, because of the monotonic relationship between λ and rate, is the bisection search algorithm [49]. However, this algorithm generally requires a non-deterministic number of "trial" encodings of a frame, using intermediate values for λ, before a suitable value is obtained. In Table 4.3, this approach was shown to add significant computations and delay.

In our system, to avoid iterating until a suitable value of λ is obtained, we attempt to model the choice of λ as a function of the base and enhance-

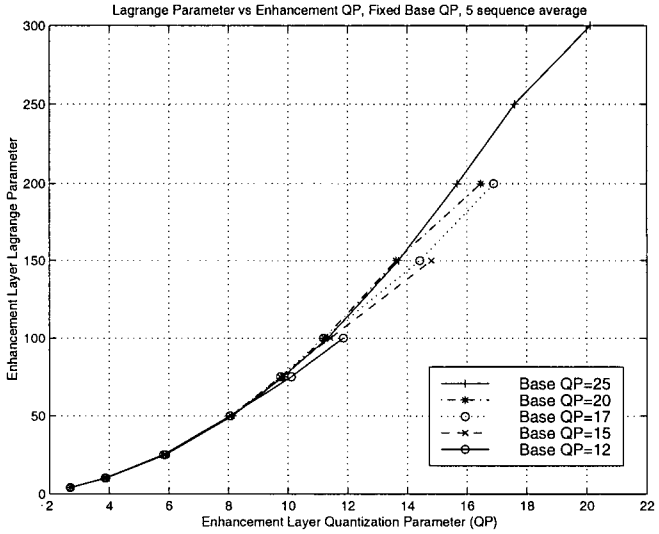


Figure 4.1: Relationship between the enhancement layer Lagrangian and quantization parameters for SNR scalability.

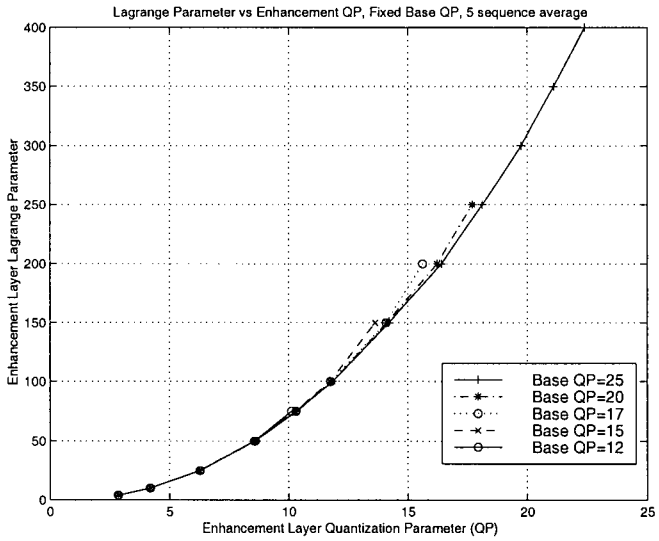


Figure 4.2: Relationship between the enhancement layer Lagrangian and quantizer levels for spatial scalability.

ment layer quantizer levels, $Q(0)$ and $Q(1)$ [54]. This approach is intuitively the most natural based on the observation regarding the close relationship between quantization level and rate-distortion tradeoffs. Moreover, this approach allows the rate-distortion optimized framework to work easily in conjunction with independent rate control techniques that control the average bit rate by adjusting the quantizer level. This approach was demonstrated to work well for single layered encoding in [54, 103, 104] using the relationship

$$\lambda(0) = 0.85 \times Q(0)^2. \tag{4.3}$$

In Figure 4.1, we plot the average SNR enhancement layer quantizer level $Q(1)$ obtained by fixing $\lambda(1)$ and allowing $Q(1)$ to vary. Results were obtained by gathering data for five different sequences, using six different values of $Q(0)$ for each sequence, and nine different values of $\lambda(1)$ for each value of $Q(0)$. For fine enhancement layer quantizers, i.e. less than 10, the relationship between the enhancement layer quantizer level and Lagrangian parameters is well approximated by the second order polynomial

$$\lambda(1) = 0.8 \times \left(\frac{Q(1)}{2}\right)^2 - 0.25 \times \left(\frac{Q(1)}{2}\right) - 1.25. \tag{4.4}$$

For coarse enhancement layer quantizers, i.e. greater than 10, the relationship between the enhancement layer quantization and Lagrangian parameters is well approximated by the linear equation

$$\lambda(1) = \alpha \times \left(\frac{Q(1)}{2}\right) - \beta, \tag{4.5}$$

where

$$\alpha = 0.8 \times \left(\frac{Q(0)}{2}\right) + 3 \tag{4.6}$$

and

$$\beta = 9 \times \left(\frac{Q(0)}{2}\right) - 66. \tag{4.7}$$

In Figure 4.2, we plot the average enhancement layer quantizer level obtained from similar experiments conducted for spatial enhancement layers. For fine enhancement layer quantizer levels, i.e. less than 10, the relationship between the enhancement layer quantizer level and Lagrangian parameters is well approximated by the second order polynomial

$$\lambda(1) = 0.8 \times \left(\frac{Q(1)}{2}\right)^2 - \left(\frac{Q(1)}{2}\right). \tag{4.8}$$

For coarse enhancement layer quantizer levels, i.e. greater than 10, the relationship between the enhancement layer quantizer level and Lagrangian parameters is well approximated by the second order polynomial

$$\lambda(1) = \alpha \times \left(\frac{Q(1)}{2}\right)^2 - \beta \times \left(\frac{Q(1)}{2}\right), \tag{4.9}$$

where α and β depend on $Q(0)$, as determined by plotting the empirical values against $Q(0)$, and are given by

$$\alpha = 0.003 \times \left(\frac{Q(0)}{2}\right)^2 - 0.2 \times \left(\frac{Q(0)}{2}\right) + 2.8 \tag{4.10}$$

and

$$\beta = 0.03 \times \left(\frac{Q(0)}{2}\right)^2 - 1.6 \times \left(\frac{Q(0)}{2}\right) + 21.4. \tag{4.11}$$

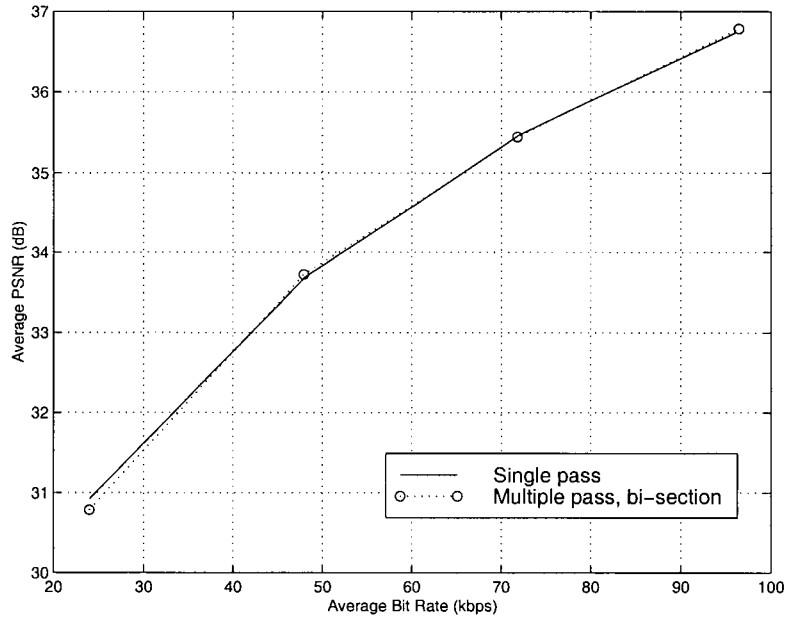
In Table 4.4, new profiling results are presented for the test scenarios. For these results, the encoder incorporates equations (4.4) - (4.11) to set the Lagrangian

Identifier	Total Instructions (millions)	Ratio
1	12957	0.24
2	14903	0.24
3	23842	0.16
4	38181	0.14
5	17417	0.19
6	20177	0.22
7	29844	0.22
8	36053	0.21

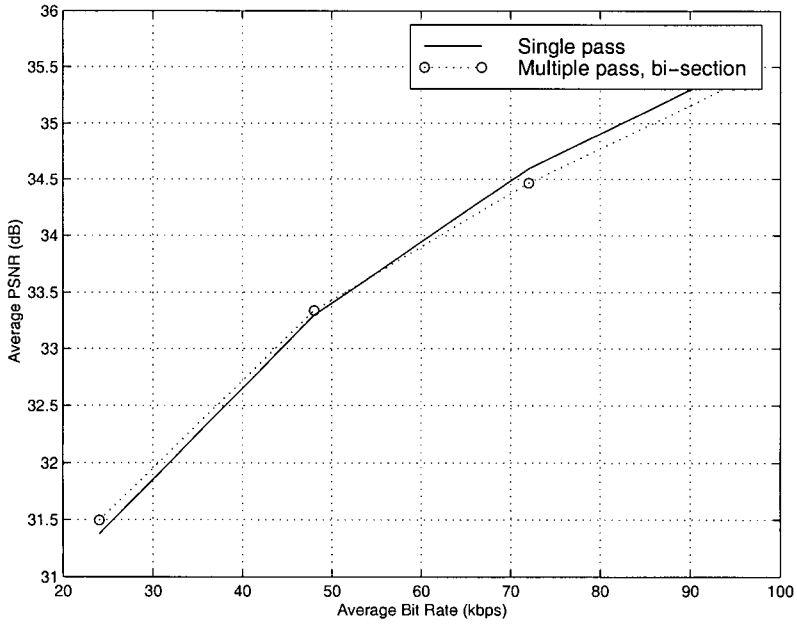
Table 4.4: Total instructions (in millions) for the test scenarios.

parameter. Total instructions are again presented for 30 frames. Furthermore, the reduction in complexity afforded by this relationship can be seen from the last column. This column shows the ratio of the instruction counts for the runs of the modified encoder relative to the original encoder. Clearly, the proposed modification reduces complexity by as factor of 4-7. This variation is due to the non-deterministic number of iterations per frame required by the original encoder, in order to closely match the target bit rate.

In Figures 4.3-4.5, we plot the rate-distortion performance of two encoders for the sequences FOREMAN and COASTGUARD. In all cases, the encoders operate with an explicit rate constraint. The different data points are the result of changing the value of the target bit rate. The first encoder employs the bisection search algorithm to closely match the target bit rate. The second encoder incorporates equations (4.4) - (4.11) and the rate-control algorithm described in the H.263 Test Model TMN11 [67, 105] to closely match the target bit rate. Specifically, the rate control algorithm selects the initial quantizer

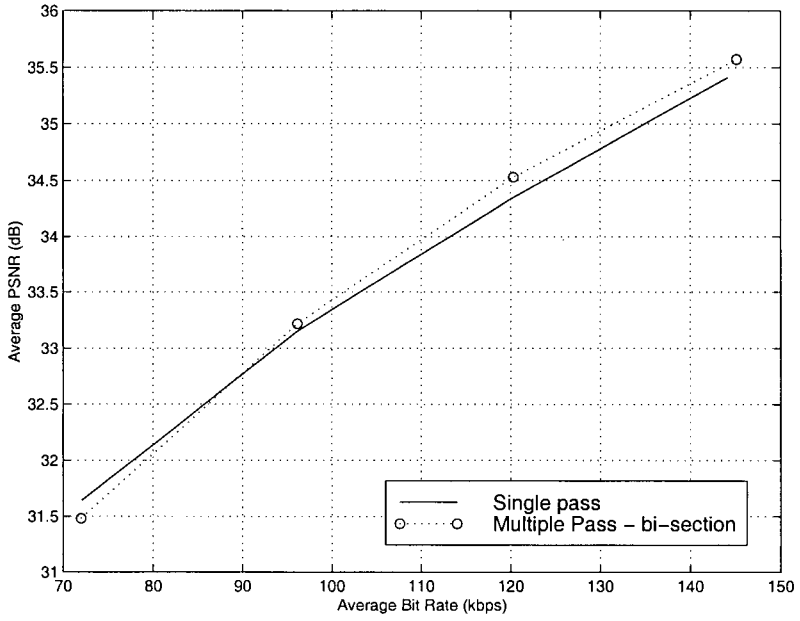


(a)

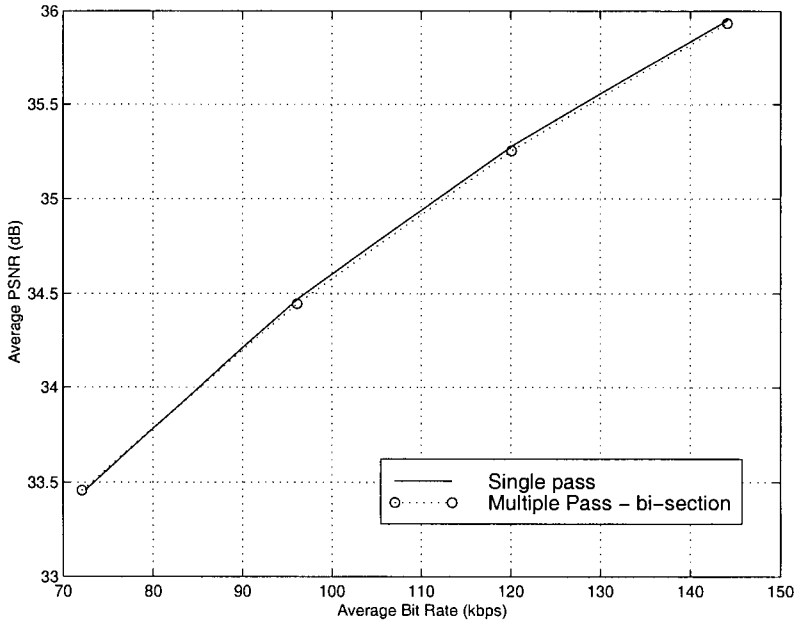


(b)

Figure 4.3: PSNR versus total bit rate, (a) FOREMAN and (b) COASTGUARD, QCIF, 10 fps, for different approaches to choosing the Lagrangian parameter.

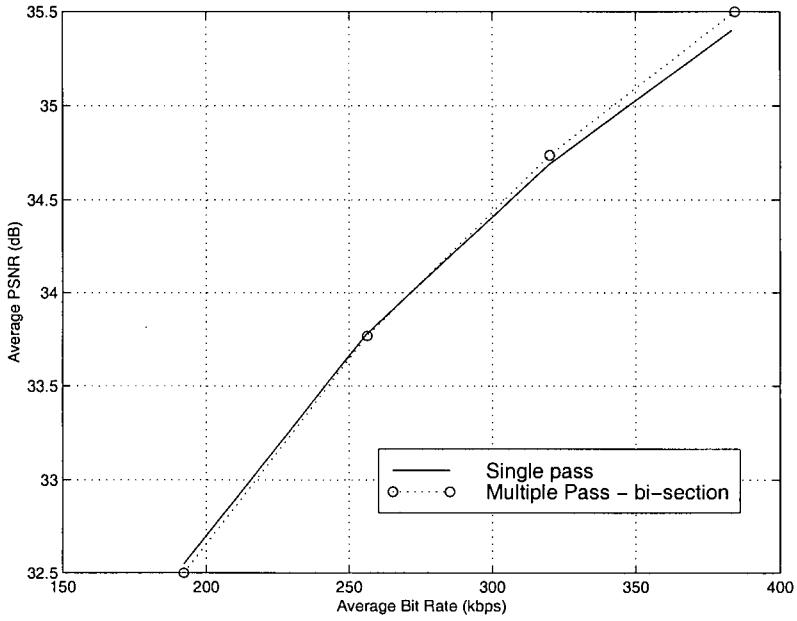


(a)

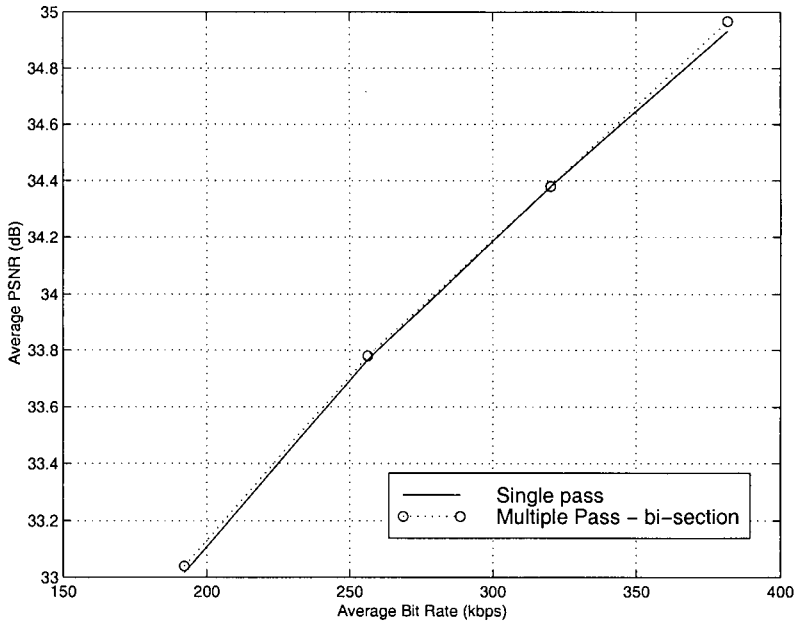


(b)

Figure 4.4: PSNR versus total bit rate, (a) FOREMAN and (b) COASTGUARD, base layer QCIF, 10 fps, enhancement layer QCIF, 10 fps, for different approaches to choosing the Lagrangian parameter, SNR scalability.



(a)



(b)

Figure 4.5: PSNR versus total bit rate, (a) FOREMAN and (b) COASTGUARD, base layer QCIF, 10 fps, enhancement layer CIF, 10 fps, for different approaches to choosing the Lagrangian parameter, spatial scalability.

level for the frame and updates this level for each macroblock. The Lagrangian parameter is then set based on this level, as specified by the above equations. In Figure 4.3 we see that the Lagrangian approximation employed for encoding a single layer achieves essentially the same rate-distortion performance as for the locally optimal bit allocation. In Figures 4.4 and 4.5 we see the same for SNR and spatial scalability respectively. Thus, the proposed approach for controlling the Lagrangian parameter, which can significantly reduce encoding complexity, can also maintain essentially the same rate-distortion performance as the optimal bit allocation approach, for both layered and non-layered coding.

4.3 Conclusion

In this chapter, we studied the complexity of the proposed rate-distortion optimization algorithms for layered video encoding in error-free environments. We re-formulated the exhaustive minimization as a cascade of local minimizations. Furthermore, we decoupled the motion estimation and mode decision optimizations. To reduce the complexity of the proposed algorithms for layered video encoding in error-free environments, we proposed a model to control the Lagrangian parameter $\lambda(1)$, for SNR and spatial scalability, using the quantization parameter. This model was the main contribution of this chapter. It was shown to significantly reduce encoding complexity while maintaining essentially the same rate-distortion performance as an exhaustive optimal bit allocation that employed the bisection search algorithm.

Chapter 5

Efficient and Robust Layered Coding for Error-Prone Environments

In this chapter, we consider layered video encoding and transport in lossy packet-switched networks. The main goal is to propose algorithms for robust layered video communications. In order to do so, we develop a framework based on the principle of layered encoding with transport prioritization. A complete layered coding and transport framework is developed, including a packetization scheme, decoder error concealment method, and prioritization mechanism. This framework is an important contribution of our work. We then introduce the general formulation for a layered video encoding algorithm for error-prone environments. This algorithm is based on the concept of operational rate-distortion optimization and can be viewed as a generalization of

the algorithm introduced for error-free environments in Chapter 3. The algorithm incorporates a statistical distortion measure that considers the channel conditions, error recovery capability of the channel codec and error concealment capability of the source decoder to optimize the video encoding mode selection. This algorithm is the main contribution of this chapter. Then, for a given layered bitstream and given channel conditions, optimal channel protection code rates are determined. This framework is shown to achieve substantial improvement in reconstructed video quality for a wide range of packet loss rates. Moreover, it is demonstrated to yield graceful degradation of reconstructed video quality with increasing packet loss rate. Finally, we study the effect of parameter mismatch on the performance of the proposed framework.

5.1 Introduction

The problem of rate-distortion optimized mode selection for video communications in error-prone environments was considered in [106]. However, this approach does not address the joint design of source and channel coder. Moreover, it is based on a non-layered video encoding algorithm. In this chapter, we present an effective framework for video communications in error-prone environments based on the principle of layered encoding with transport prioritization.

We further develop the rate-distortion optimized mode selection algorithm, presented in Chapter 3, for layered video encoding within a prioritized

transport framework. The algorithm incorporates a statistical distortion measure that considers the channel conditions, the error recovery capability of the channel codec and the error concealment capability of the source decoder to optimize the video encoding mode selection. More specifically, we want to select the coding mode for each block in each layer such that, given the different layer reliabilities and the corresponding decoder error concealment methods for these layers, the expected reconstruction distortion is minimized for a given bit rate.

First, however, key components of the framework must be developed. We introduce a packetization scheme for layered bitstreams that minimizes packetization overhead and facilitates decoder error concealment. We propose an effective error concealment method for enhancement layers that exploits the availability of more reliable base layer information. We then consider the joint design of source and channel coder. For a given layered bitstream and channel condition we determine the optimal channel protection code rate. We demonstrate that the proposed framework achieves significant improvement in and provides graceful degradation of reconstruction quality for increasing packet loss rate.

This chapter is outlined as follows. In Section 5.2 we present the various components of the proposed framework, including the packetization scheme, the decoder error concealment method and the prioritization mechanism. In Section 5.3, we further develop the rate-distortion optimized mode selection algorithm that was presented in Chapter 3. Simulation results are presented

in Section 5.4. Conclusions are stated in Section 5.5.

5.2 Background

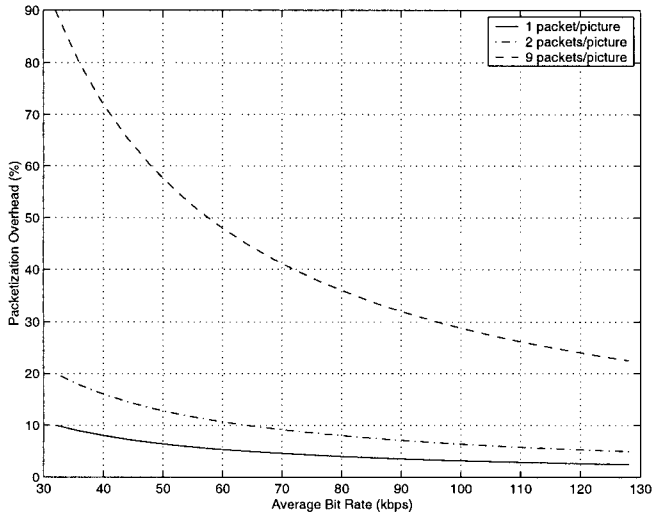
In this section we develop the low bit rate layered video encoding and prioritized transport framework that is a key component of our work. This includes the packetization scheme, the decoder error concealment method and the prioritization approach.

5.2.1 Packetization

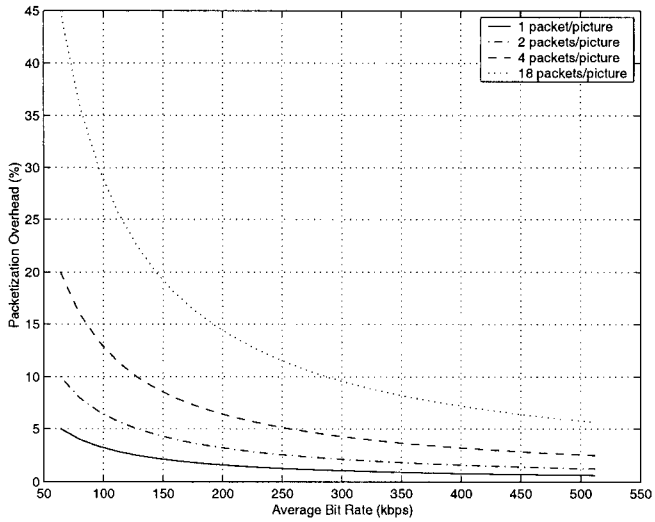
Video communications in packet-lossy networks was discussed in Section 2.5. In this section, it was stated that the packetization overhead for RTP/UDP/IP was approximately 40 bytes per packet. To minimize packetization overhead, the size of the payload data should be substantially more than the size of the header. Furthermore, considering the fragmentation limit of intermediate nodes on the Internet, the maximum size the packet should be 1500 bytes. This would allow approximately 1450 bytes, or 11600 bits, for the video data. Thus, a single coded frame could easily fit within a single packet. If we consider a sequence encoded at 10 fps, utilizing the maximum payload size for every packet, the total video bit rate would be 116 kbps. This is more than sufficient for good quality QCIF resolution video. Obviously, the total video bit rate scales linearly with increasing frame rate. This would suggest that employing one packet for each coded frame. However, from an error resilience perspective, this means that the loss of one packet means losing an entire coded frame.

Objectively, if we are maximizing the overall PSNR, then it is more desirable to limit the spatial area affected by a packet loss by dividing the coded frame over many packets. Subjectively, however, there has been very little work studying whether or not it is better to lose an entire coded frame or only part of a coded frame. It is possible that the potentially high frequency concealment artifacts introduced when only a part of the spatial area of a frame is concealed are more objectionable than would be the loss of the entire frame. Using the reference picture selection mode of H.263 [1], as discussed in the temporal error resilience paragraph in Section 2.5, would improve the robustness of a video encoding and transport framework that employed one packet per coded frame.

Therefore, conceivable lower and upper bounds for the payload data are from one row of macroblocks (GOB) per packet to one entire coded frame per packet. In the case of the former, loss of a packet can be mitigated by a good decoder error concealment method however, at low bit rates, the overhead is prohibitive. In the case of the latter, the overhead is significantly reduced however loss of a packet means loss of an entire coded frame. In Figure 5.1 we illustrate the packetization overhead resulting from various packetization approaches. In Figure 5.1(a), for QCIF resolution frames (176×144 pixels or 9 GOBs), we illustrate schemes generating nine packets per coded frame or one packet per GOB, two packets per coded frame, interleaving even and odd GOBs into separate packets as proposed in [107, 67], and one packet per coded frame. In Figure 5.1(b), for CIF resolution frames (352×288 pixels or



(a)



(b)

Figure 5.1: Packetization overhead for various packetization schemes for FOREMAN at (a) QCIF and (b) CIF resolution.

18 GOBs), we illustrate schemes generating eighteen packets per coded frame or one packet per GOB, four packets per coded frame, interleaving every four GOBs into separate packets, two packets per coded frame, interleaving even and odd GOBs into separate packets, and one packet per coded frame.

¿From the figure, it is clear that generating one packet per GOB results in excessive packetization overhead. The remaining schemes result in reasonably low packetization overhead. However, as will be demonstrated in Section 5.2.2, employing a single packet per coded frame performs poorly under increasing packet loss. The remaining schemes facilitate decoder error concealment. For these schemes if only one packet for a given coded frame is received, the decoder can perform temporal error concealment using motion information from the correctly received packet.

We should point out that, as picture header information is critical to resolve temporal reference, frame type, associated layer, as well as a number of additional coding options, we transmit a redundant picture header as part of the payload header of all packets associated with a given coded frame, at the cost of approximately eight additional bytes per packet [1, 77].

5.2.2 Error Concealment Method

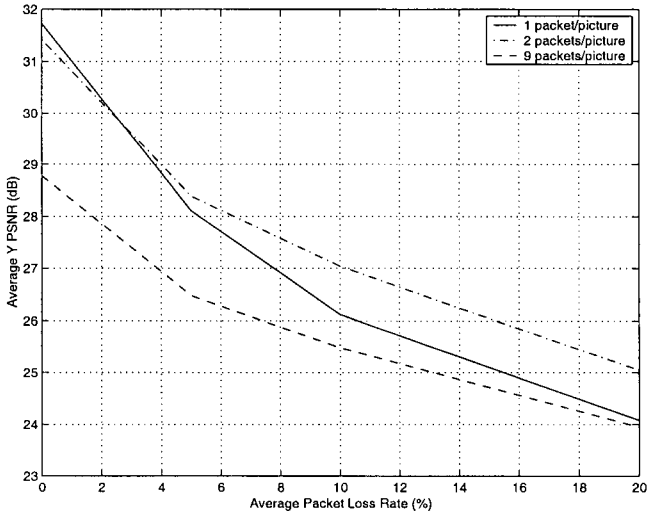
As UDP is not intended to improve quality of service, UDP-based communications often suffer substantial packet loss [108]. Thus, the communications system must be able to mitigate the effect of packet loss. This can be accomplished in part by employing error-resilient source coding and traditional

channel coding techniques. However, the source decoder must also be able to conceal any residual packet loss. Before losses can be concealed they must first be detected. This is straightforward using the sequence number field included in the RTP header. Furthermore, for packetization schemes that generate more than one packet per coded frame, resynchronization markers are necessary to provide spatial error-resilience. For H.263+, GOB headers are one method to provide such spatial error-resilience [1]. GOB headers include the associated GOB number as well as the absolute quantizer level. Moreover, the use of GOBs restricts certain predictive elements of the syntax. This limits the spatial extent of error propagation. When a missing GOB is detected, the source decoder searches for the next available synchronization marker. From this new synchronization marker, decoded motion and quantizer information will be correct. Error concealment is then performed on the missing GOB or GOBs.

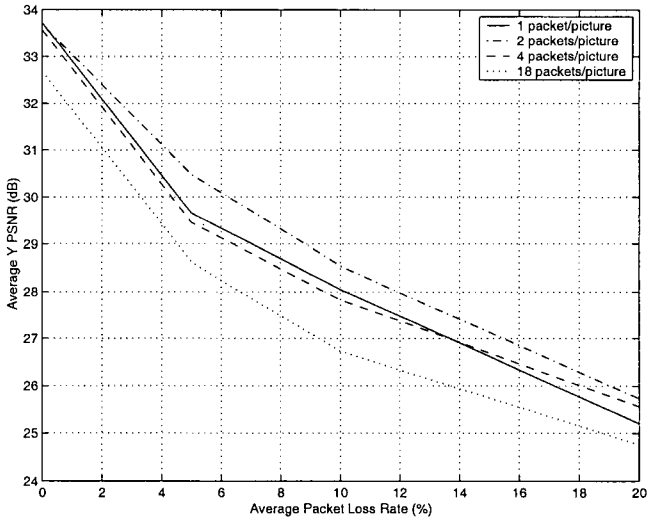
Error concealment in video communications was reviewed in Section 2.5. In that section, we discussed several temporal domain approaches to error concealment. We noted that using the median estimate for motion compensation was shown to yield better subjective quality than the averaging technique [99, 67] and that this approach would be employed in this thesis. In this approach, the motion vector for the missing block is set to the median value of the motion vectors from the blocks to the left, above and above right of the missing block. If no motion vectors are available in these positions, the estimated motion vector is set to $(0, 0)$.

In Figure 5.2, we illustrate the performance of the median estimation-based temporal error concealment method, using the packetization schemes discussed in Section 5.2.1, under a range of packet loss rates. Results are presented for non-layered scenarios using ten seconds of the video sequence FOREMAN coded at ten frames per second. Results are for QCIF resolution in Figure 5.2(a) and CIF resolution in Figure 5.2(b), for total channel bit rates of 64 and 256 kbps respectively. The actual video bit rate is obtained by deducting the packetization overhead from the total channel bit rate. Statistics are averaged from twenty simulation runs.

Clearly, from the figure, the use of one packet for each GOB results in inferior performance. While this approach facilitates error concealment by the decoder, the packetization overhead severely reduces the available video bit rate. The approach that employs one packet for each coded frame provides satisfactory performance under low packet loss rates. However, as the packet loss rate increases the performance begins to degrade significantly. This is because the loss of a packet results in the loss of an entire coded frame. Moreover, this approach regularly produces packets which exceed the desired maximum packet size of 1500 bytes. The other approaches, generating two and four packets for each coded frame, maintain reasonable performance levels over the entire range of packet loss rates. For the CIF resolution results, the approach that employs two packets per coded frame occasionally exceeds the desired maximum packet size. Therefore, we adopt the packetization scheme of generating two packets per coded frame for QCIF resolution and four packets per



(a)



(b)

Figure 5.2: PSNR versus packet loss rate for various packetization schemes for FOREMAN at (a) QCIF and (b) CIF resolution.

coded frame for CIF resolution.

This error concealment method works well for non-layered scenarios. However, for layered scenarios, we can improve the estimation of missing enhancement layer information by considering available base layer information. Obviously, since the previous enhancement layer reconstruction is generally of higher quality than the current base layer reconstruction, it should be exploited for error concealment. However this should only be done when it is expected that motion compensated error concealment will provide a reliable estimate of the missing information. For our purposes, this criterion is satisfied when the corresponding base layer region has been inter-coded. In this case, we employ the median estimator within the enhancement layer and perform motion compensated error concealment. When the corresponding base layer region has been intra-coded, we assume that motion compensation did not produce a satisfactory prediction at the encoder. In this case, the missing enhancement layer information is concealed using the available base layer reconstruction. One further consideration in our approach is that we should be able to limit temporal error propagation in the enhancement layer by exploiting the greater reliability of the base layer reconstruction. Thus, in all cases where our algorithm chooses to employ motion compensation for error concealment, we only permit this if the corresponding region in the previous enhancement layer reconstruction has not itself been concealed. If this region has been concealed, the missing enhancement layer information is instead concealed using the available base layer reconstruction. This process is outlined

Layer (Resolution)	Available Bit Rate	FEC Code	Packetization Overhead Rate	Previous Layer FEC Bit Rate	Total Video Bit Rate
1 (QCIF)	48000	(15,9)	6400	0	41600
2 (CIF)	348000	none	12800	32000	303200

Table 5.1: Layering, FEC codes, and associated rates for packetization overhead (per layer), video source bit rate, and FEC bit rate used for decoder error concealment simulations. The overall bit rate is 396 kbps.

in Figure 5.3.

We illustrate the performance of several enhancement layer error concealment methods, including the method proposed above, in Figure 5.4. Here, we plot the enhancement layer PSNR, under different packet loss rates, for the sequences FOREMAN and COASTGUARD when different decoder error concealment methods are employed. Results are presented for two layers of spatial scalability, at QCIF and CIF resolutions, coded at ten frames per second. Statistics are averaged from twenty simulation runs. As the proposed framework will be prioritized, we apply unequal error protection as outlined in Table 5.1 to evaluate the performance of the error concealment methods. How the unequal error protection is applied is discussed in detail in the next section. For these experiments it is sufficient to note that, in all cases, the total channel bit rate is approximately the same. The enhancement layer video bit rate is calculated by deducting the base layer video bit rate, the FEC bit rate, and the enhancement layer packetization bit rate from the total channel bit rate. This corresponds to the notion of *throttling* the video bit rate [109].

The first method always employs the median estimator to perform mo-

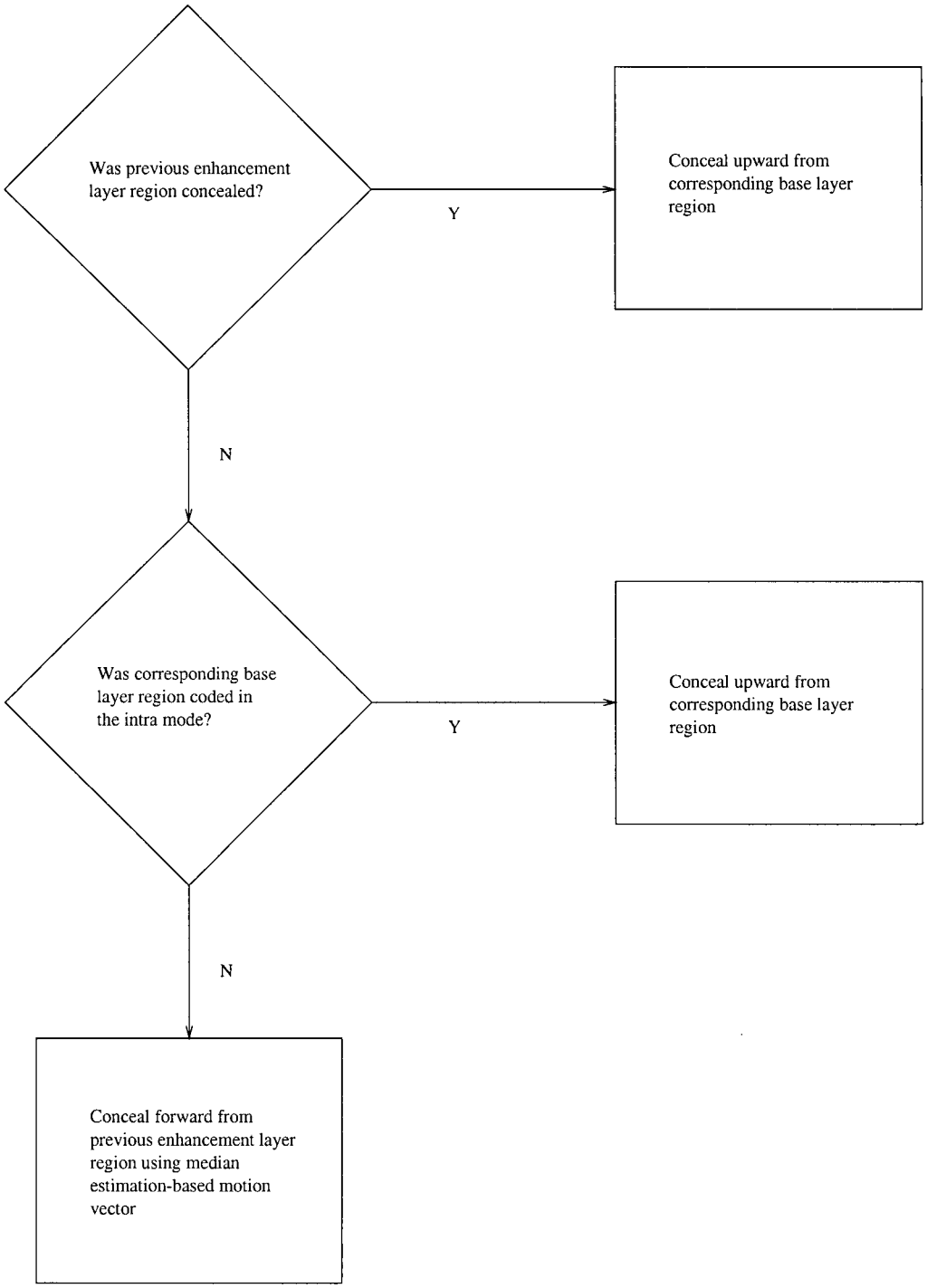
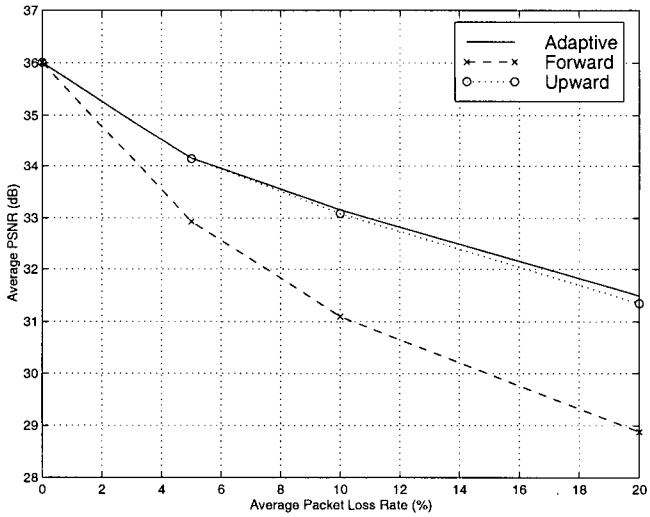
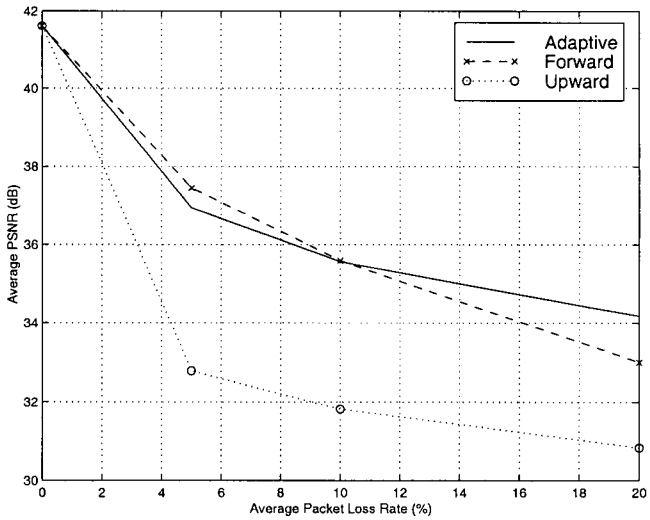


Figure 5.3: Block diagram of the proposed enhancement layer error concealment method.



(a)



(b)

Figure 5.4: PSNR versus packet loss rate for enhancement layer error concealment methods for (a) FOREMAN and (b) COASTGUARD. Spatial scalability, base layer QCIF, enhancement layer CIF resolution.

tion compensated error concealment within both layers and is labeled "forward". The second method employs the median estimator to perform motion compensated error concealment in the base layer, and relies on the base layer reconstruction for error concealment in the enhancement layer. This method is labeled "upward". The third method employs the algorithm described above and is labeled "adaptive". From Figure 5.4, we see that the relative performance of the forward and upward error concealment methods depends highly on the sequence. For the low activity sequence COASTGUARD, the forward method performs well. For the high motion sequence FOREMAN, the upward method outperforms the forward method. In both cases, the proposed adaptive error concealment method achieves essentially the same performance as the better of the forward and upward methods with one exception for the COASTGUARD sequence at 5 % packet loss rate. Here, the forward error concealment method outperforms the adaptive error concealment method. We have already pointed out that, because COASTGUARD is a low activity sequence, forward error concealment outperforms upward error concealment. This, combined with the fact that our adaptive error concealment method will select to conceal upward from the base layer when the previous enhancement layer image region has been concealed, is the source of discrepancy. For light losses and low activity sequences, performing motion compensated error concealment from an image region that has itself been concealed appears to be sufficient. We should point out that the reduced performance for our adaptive error concealment method is visible mainly as blurring artifacts, due to

the upsampling from the base layer. The forward error concealment method can still exhibit the occasional concealment artifact which is significantly more displeasing.

While the forward method exploits the higher quality enhancement layer reconstruction, it fails to consider the increased reliability of the base layer reconstruction and its associated motion information. The upward method does consider the higher reliability of the base layer reconstruction, but fails to exploit the higher quality enhancement layer reconstruction when there is an opportunity to do so. The proposed adaptive error concealment method considers the higher reliability of base layer reconstruction and its associated motion information. It uses this information to determine whether or not it is appropriate to exploit the higher quality of the available enhancement layer reconstruction.

5.2.3 Prioritization Approach

As stated previously, a layered coding framework is well suited to transport prioritization. Certain networks, such as the Internet, are not engineered to provide different levels of quality of service. Therefore, prioritization is not possible at the network layer. Therefore it must be implemented at the application layer. In this case, unequal error protection is a natural choice to achieve transport prioritization. The base layer can be assigned to a high priority class while the enhancement layers can be assigned to lower priority classes. In our approach, FEC is applied to the base layer bitstream to pro-

duce a high priority class and no protection is applied to the enhancement layer bitstream, resulting in a low priority class.

FEC-based techniques have been widely examined for video communications [93, 94, 109]. Furthermore, FEC-based techniques are currently being considered by the IETF for supporting transport of real-time media [110]. In [109], a judicious code rate selection strategy, combined with a simple error concealment method was shown to substantially enhance performance of high bit rate video communications in ATM networks for only a small set of preselected codes.

For our framework, we want to maintain the same total channel bit rate. Thus, as stated above, the FEC bit rate is deducted from the video bit rate. This will not only prevent unwanted bit rate expansion but also allow us to determine how to optimize the allocation of the total channel bit rate. We expect a rigorous code selection process, closely related to the channel conditions, to yield significant performance improvements, as a reduced FEC bit rate will increase the available video bit rate. For these results, we evaluate a range of strong, low delay codes, in order to enable recovery with minimal overhead. Thus, we employ maximal distance separable (MDS) codes, an example of which are Reed-Solomon (RS) codes [92].

The FEC is applied across packets, as depicted in Figure 5.5. For an (n, k) code, for k data packets, $n - k$ parity packets are generated. For the proposed packetization scheme, the data packet sizes are not fixed, and should be no larger than 1500 bytes. However, for a block of k data packets, the

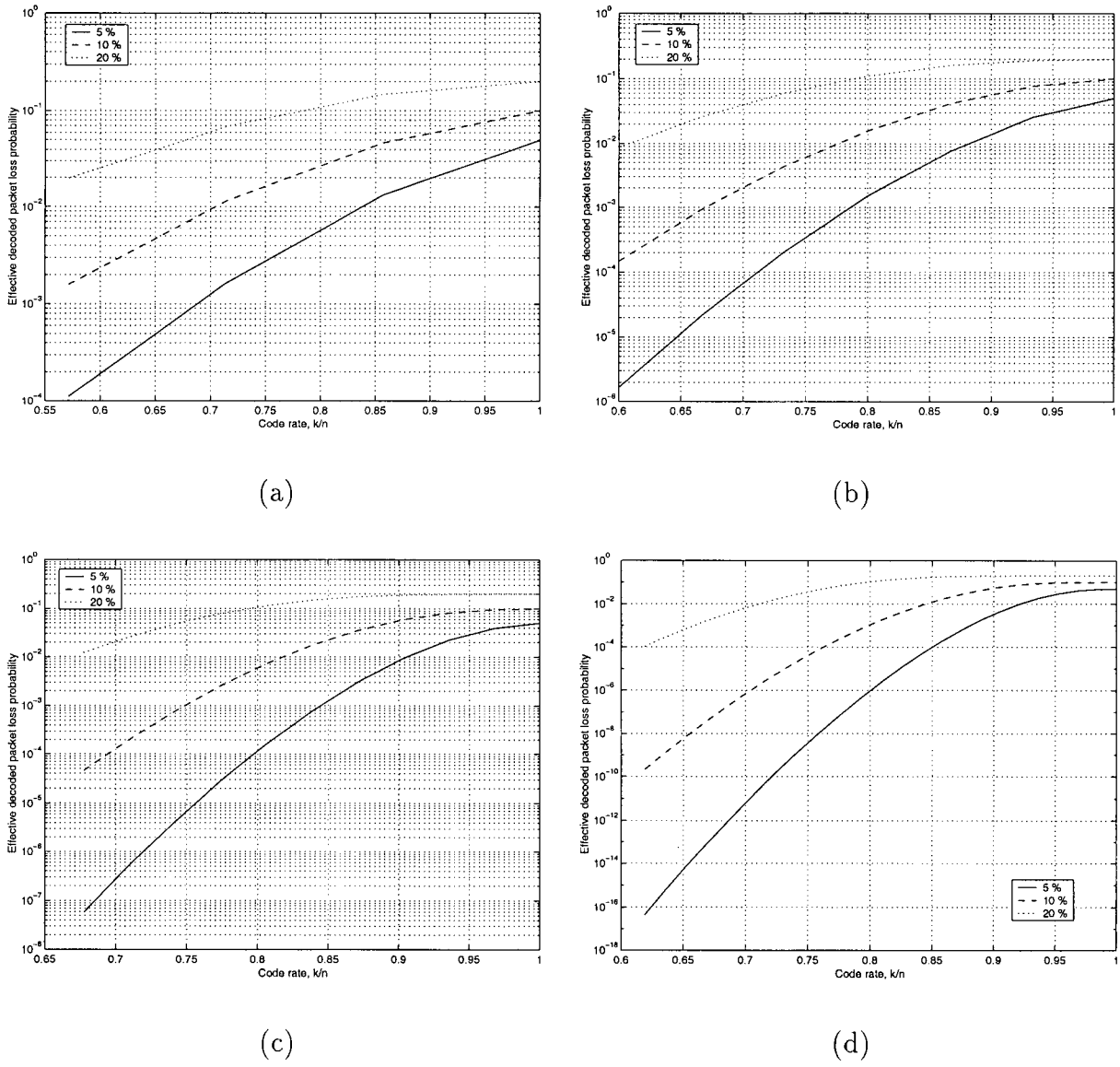
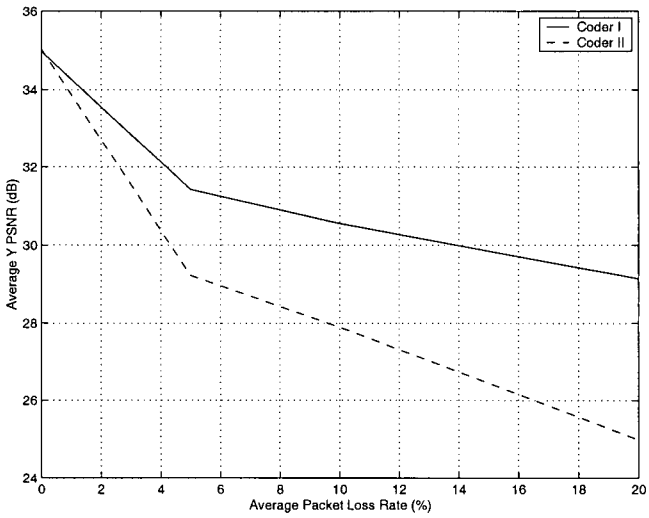
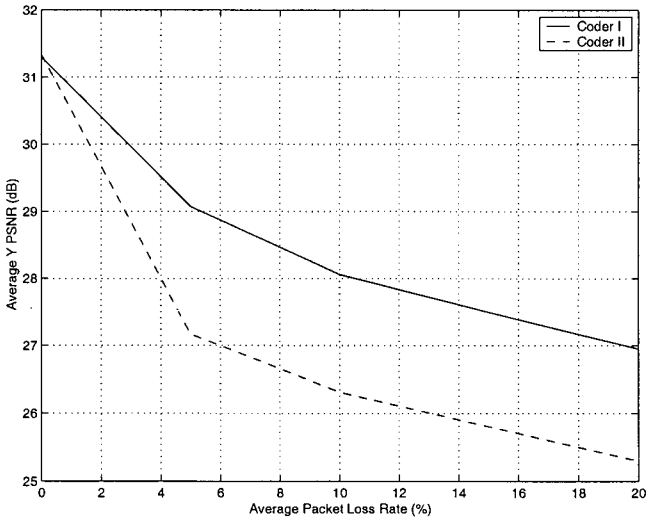


Figure 5.6: Residual packet loss probabilities for different packet loss rates and FEC code rates with code length (a) $n = 7$, (b) $n = 15$, (c) $n = 317$ and (d) $n = 63$.



(a)



(b)

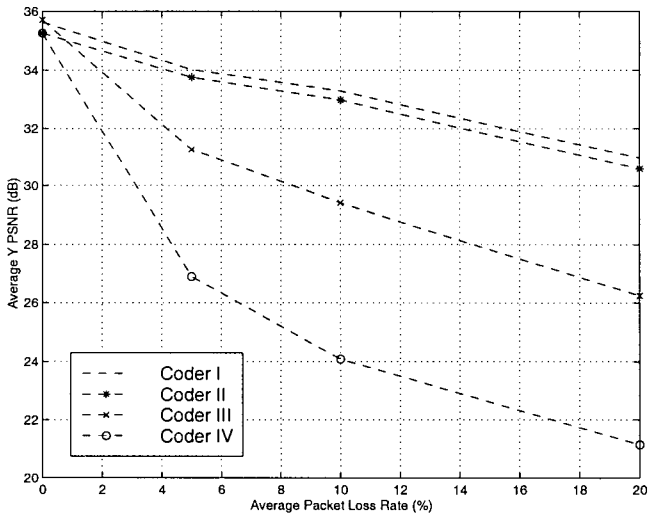
Figure 5.7: PSNR versus packet loss rate with and without unequal error protection for (a) FOREMAN and (b) COASTGUARD. Spatial scalability, base layer QCIF, enhancement layer CIF resolution.

frames per second for the sequences FOREMAN and COASTGUARD for packet loss rates of 0, 5, 10 and 20%. [1] Both frameworks employ the packetization scheme and decoder error concealment method discussed in Sections 5.2.1 and 5.2.2. Also, both frameworks use the rate-distortion optimized mode selection algorithm that is described in Section 5.3 below. The only difference is that CODER I adds unequal error protection, by applying the optimal amount of FEC, as determined in Section 5.4.1 below, to the base layer bitstream. In all cases, the total channel bit rate is approximately the same. Statistics are averaged from twenty simulation runs. Using the proposed prioritization approach, we observe a significant performance improvement, 2-4 dB, for packet loss rates above 5%.

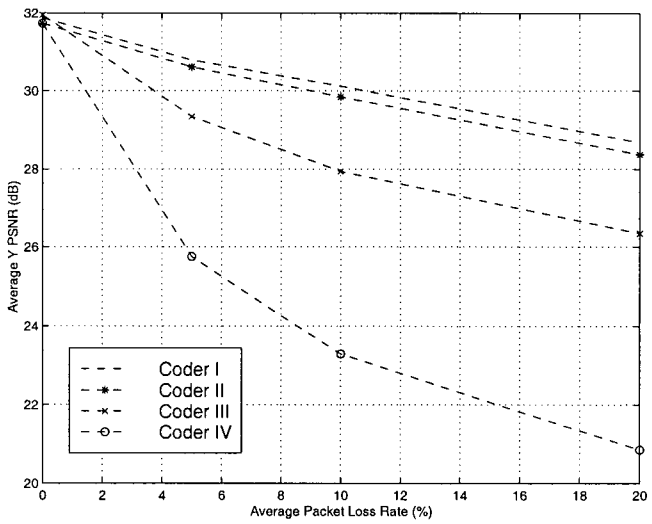
In Figure 5.8 we highlight the improvement in performance that can be realized by employing rate-distortion optimization and unequal error protection in a non-layered framework. Results are illustrated for CIF resolution, using ten seconds of video at ten frames per second for the sequences FOREMAN and COASTGUARD for packet loss rates of 0, 5, 10 and 20%. Both frameworks employ the packetization scheme and non-layered decoder error concealment method discussed in Sections 5.2.1 and 5.2.2. The coders employing rate-distortion optimization employ the method that has recently been proposed [107, 67]. The different curves correspond to

- A protected framework whose mode is rate-distortion optimized (CODER I)

[1] Recent research has shown that loss rates of 20% or more are common for many public Internet connections [111, 112, 108].



(a)



(b)

Figure 5.8: PSNR versus packet loss rate with and without rate-distortion optimization and unequal error protection for (a) FOREMAN and (b) COAST-GUARD. Single layer CIF resolution.

- A protected framework whose mode is not rate-distortion optimized (CODER II)
- An unprotected framework whose mode is rate-distortion optimized (CODER III)
- An unprotected framework whose mode is not rate-distortion optimized (CODER IV)

The protected frameworks employ packet-based FEC, by applying the optimal amount of FEC, as determined in Section 5.4.1 below. In all cases, the total channel bit rate is approximately the same. Statistics are averaged from twenty simulation runs. For the coders employing rate-distortion optimization, Coder I and Coder III, we observe a performance improvement of 4-5 dB by employing packet-based FEC for packet loss rates above 5%. For the coders not employing rate-distortion optimization, Coder II and Coder IV, we observe a performance improvement of 5-8 dB by employing packet-based FEC for packet loss rates above 5%. Thus, while in this chapter we focus on packet-based FEC for unequal error protection in a layered coding and prioritized transport framework, we have demonstrated here that packet-based FEC can provide significant performance improvements in a non-layered framework.

We should point out that, for our channel model, we have assumed that packet losses are not correlated. This assumption is reasonable as the proposed packetization scheme generates very few packets per picture. Because there is such a large interval between the time instances when successive packets

are injected into the network, we expect little correlation in the packet loss process.

5.3 Proposed Method

Rate-distortion optimization for video encoding in error-free environments was reviewed in [54]. Extending this approach to error-prone environments was discussed in [7]. Whereas the error-free case involves determining the optimal allocation of bit rate among source coding elements, the error-prone case requires optimizing the allocation between source coding and channel coding elements. Moreover, the allocation of bit rate among source coding elements should introduce appropriate error-resilience into the bitstream, related to the particular channel conditions. This is the essence of our rate-distortion optimized mode selection algorithm. The algorithm determines when and where to introduce temporal error-resilience. Then, in Section 5.4, for a given layered bitstream and different packet loss rates, we determine the optimal amount of unequal error protection by studying the performance of our proposed framework for a wide range of FEC code rates.

For the base layer, we introduce temporal error-resilience through the insertion of intra blocks. More interestingly, for the enhancement layer we introduce temporal error-resilience through the insertion of blocks predicted upward from the more reliable base layer. This saves the enhancement layer from spending expensive bits on intra-coding while providing the benefits of temporal error-resilience.

In this section we introduce an algorithm that controls the operating mode of our layered video encoder. First, a statistical distortion measure for our layered coding and prioritized transport framework is presented. Then, we describe the rate-distortion optimized mode selection algorithm.

5.3.1 Statistical Distortion Measure

In this section we introduce a statistical measure for the error introduced via packet loss and propagated via motion compensation in a layered video encoding framework. The important parameters of this measure are the network packet loss rate, the error recovery capability of the channel codec (if applicable), and the error concealment capability of the source decoder. Recall that we have assumed that the packet loss process is not correlated. Furthermore we assume that the packet loss rate is independent of packet size [113]. This assumption is not valid for wireless networks, where bit errors must be considered. In such environments, optimizing packet size is an important component to ensure robust video communication [106, 114].

We can therefore use equation (5.1) for residual packet loss rate of an (n, k) code, presented in Section 5.2.3, as the probability that a given macroblock in some previous frame has been lost. We can then compute, over a window of N previous frames in layer l_{pred}, the probability that a macroblock has been lost as follows:

$$P_{corrupt}(b, l_{pred}, t-k) = 1 - (1 - P_{loss}(l_{pred}))^{N-k+1}. \tag{5.2}$$

We can now define the statistical distortion measure that accounts for the

propagation of corrupted macroblocks due to motion compensation. For this we define the following recursive measure, computed for every macroblock b in a given frame t:

$$D_c(b, l_{pred}, t, mode) = \sum_{k=1}^{N} \sum_{r=1}^{9} P_{corrupt}(b, l_{pred}, t-k) \\ w(r, l_{pred}, t-k, mode) D_c(r, l_{pred}, t-k, mode). \quad (5.3)$$

In a prioritized framework, there will be different values of $P_{corrupt}(b, l_{pred}, t-k)$ for the different layers. This is computed as in equation (5.2), for every macroblock in every frame of every layer. This value can be thought of as assigning a decreasing reliability to macroblocks in previous frames as they become further from the most recent macroblock that has been coded with the *intra mode*. $D_c(b, l_{pred}, t, mode)$ represents the expected distortion incurred from predicting the current block from previously concealed macroblocks. Obviously, for the *intra mode*, this value is set to 0. For any of the coding modes that employ motion compensation, the motion vector determines the weighting values $w(r, l_{pred}, t-k, mode)$. These weighting values reflect the relative contribution of $D_c(b, l_{pred}, t, mode)$ for any referenced macroblocks that overlap with the predicted macroblock, based on how much their areas overlap.

Note the N is reset to zero when a macroblock is updated in *intra-mode* and is, in practice, limited to a maximum of ten. Furthermore, we must assume a particular decoder error concealment method. We employ the median estimation-based temporal error concealment method.

5.3.2 Rate-Distortion Mode Selection Algorithm

We next present the rate-distortion optimized mode selection algorithm for layered video encoding in error-prone environments. For encoding the base layer, we consider four coding modes, *skipped mode*, *inter mode*, *intra mode* and *inter4v mode* [1]. For encoding the enhancement layer, we consider five coding modes, *skipped mode*, *inter-forward mode*, *inter-upward mode*, *inter-bidirectional mode* and *intra mode* [1]. For the error-free case, this amounts to determining independently for every block b in each layer l_{pred} of a given frame t the coding mode that minimizes

$$J_{mode}(b, l, t) = D(b, l_{pred}, t, mode) + \lambda(l_{pred}, t)R(b, l_{pred}, t, mode). \tag{5.4}$$

Here D is the quantization distortion and R is the resulting bit rate from encoding block b predicted from layer l_{pred} with a given *mode*. Using this approach, the mode selection algorithm is optimal for error-free communications only. In the presence of errors, the mode selection algorithm should be able to adapt and insert a controlled amount of error-resilience.

To accomplish this we now consider two sources of distortion. The first distortion D_1 is again the quantization distortion. The second distortion D_2 is the statistical distortion measure $D_c(b, l_{pred}, t, mode)$ described above. Furthermore, in addition to a constraint on the source coding bit rate we now have a constraint on the total channel bit rate $R_s + R_c$, where R_c is calculated as the channel coding rate for a given code rate k/n and source coding rate

R_s. We can then minimize the Lagrangian as

$$J_{mode}(b, l_{curr}, f) = (1 - P_{corrupt}(l_{pred}, f-1))D_1(b, l_{curr}, f, mode) + D_2(b, l_{pred}, f, mode) + \lambda(l_{curr})(R_s(b, l_{curr}, f, mode) + R_c(n, k, R_s)). \quad (5.5)$$

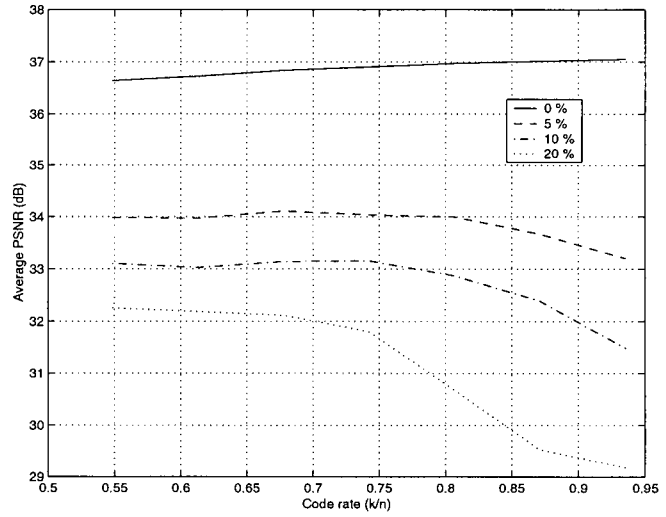
5.4 Experimental Results

In this section, we first determine experimentally, for a given layered bitstream and packet loss rate, the optimal FEC code rate. We then evaluate the performance of the proposed framework using the obtained code rates. We also compare the proposed layered framework to other layered and non-layered frameworks. Finally, we evaluate the effects of parameter mismatch.

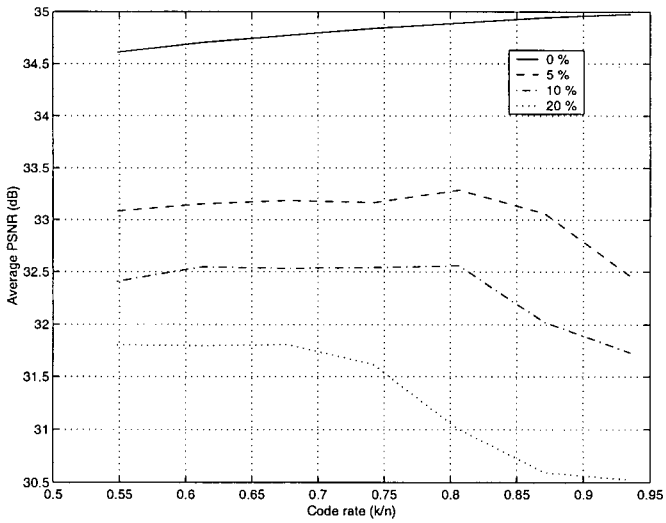
5.4.1 Determining Optimal FEC Code Rates

We first seek an appropriate code rate to be employed for a particular packet loss rate. In Figure 5.9, results are illustrated for two layers of spatial scalability, at QCIF and CIF resolution, using ten seconds of video at ten frames per second for the sequences FOREMAN and COASTGUARD. We apply different amounts of protection, as outlined in Table 5.2, for packet loss rates of 0, 5, 10 and 20%. In all cases, the total channel bit rate is approximately the same. Statistics are averaged from twenty simulation runs.

From the figure, we see that a reasonable level of quality can be maintained under even heavy packet loss rate situations by applying as little as 25-30% FEC to the base layer bitstreams. For 20% packet loss rate, the (21,31) code provides sufficient protection. The (23,31) code also provides reason-



(a)



(b)

Figure 5.9: PSNR versus code rate k/n for different packet loss rates for (a) FOREMAN and (b) COASTGUARD. Spatial scalability, base layer QCIF, enhancement layer CIF resolution. Code length $n = 31$.

Layer (Resolution)	Available Bit Rate	FEC Code	Packetization Overhead Rate	Previous Layer FEC Bit Rate	Total Video Bit Rate
1 (QCIF)	48000	(31,17)	6400	0	41600
2 (CIF)	348000	none	12800	39530	295670
1 (QCIF)	48000	(31,19)	6400	0	41600
2 (CIF)	348000	none	12800	30315	304885
1 (QCIF)	48000	(31,21)	6400	0	41600
2 (CIF)	348000	none	12800	22860	312340
1 (QCIF)	48000	(31,23)	6400	0	41600
2 (CIF)	348000	none	12800	16695	318505
1 (QCIF)	48000	(31,25)	6400	0	41600
2 (CIF)	348000	none	12800	11520	323680
1 (QCIF)	48000	(31,27)	6400	0	41600
2 (CIF)	348000	none	12800	7110	328090
1 (QCIF)	48000	(31,29)	6400	0	41600
2 (CIF)	348000	none	12800	3310	331890

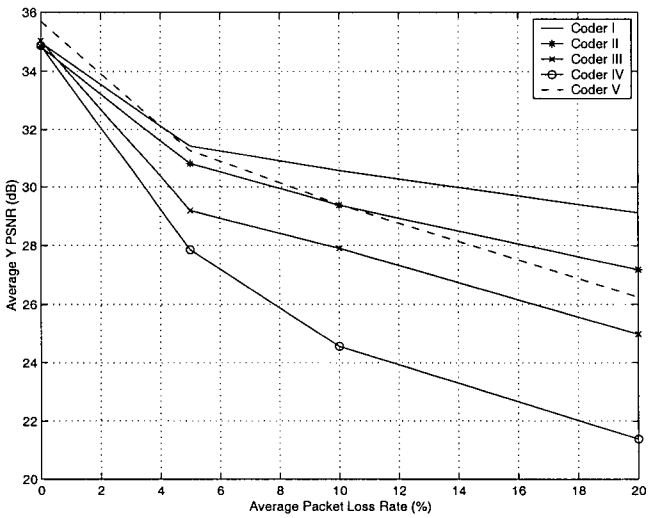
Table 5.2: Layering, FEC codes, and associated rates for packetization overhead (per layer), video source rate, and FEC rate used for packet loss versus code rate simulations. The overall rate is 396 kbps.

able protection, but the performance shows signs of beginning to deteriorate. For 10% packet loss rate, the (23,31) code provides sufficient protection. The (25,31) code also provides reasonable protection, although again the performance begins to deteriorate. For 5% packet loss rate, the (25,31) code provides good protection. Here, the (27,31) code also provides reasonable protection, with the performance beginning to deteriorate only slightly.

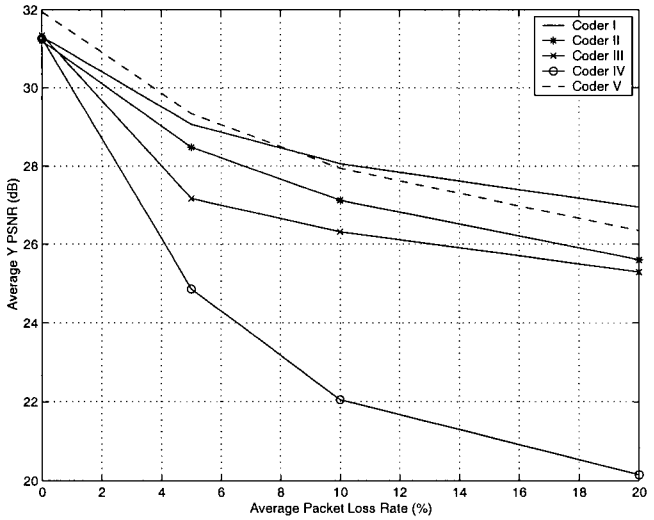
Referring to Figure 5.6(b), which illustrates the effective decoded packet loss probability for a code of length $n = 31$ and different average network packet loss rates, we see that, using the code rates determined above, the resulting effective decoded packet loss probability is less than 2%. This implies that our decoder error concealment method is capable of providing acceptable quality video when it experiences a packet loss rate of less than 2%. Moreover, this confirms that, by themselves, decoder error concealment methods can provide acceptable quality video only under light packet loss rates. Further examination reveals that, based on the code rates determined above, the resulting effective decoded packet loss probabilities are increasing slightly with increasing average network packet loss rate. This is because the temporal error-resilience of the video bitstream is also increasing with increasing average network packet loss rate. This facilitates error concealment by the decoder, permitting the framework to sustain slightly higher residual packet loss.

5.4.2 Performance of Proposed Framework

In Figure 5.10 we evaluate the performance of the proposed framework.



(a)



(b)

Figure 5.10: PSNR versus packet loss rate for five different frameworks for sequences (a) FOREMAN and (b) COASTGUARD.

Also, we include performance results for the non-layered error-resilient framework the has recently been proposed [107, 67]. Results are illustrated for two layers of spatial scalability, at QCIF and CIF resolution, using ten seconds of video at ten frames per second for the sequences FOREMAN and COASTGUARD and packet loss rates of 0, 5, 10 and 20%. All frameworks employ the packetization scheme and decoder error concealment method discussed in Sections 5.2.1 and 5.2.2. For the layered and protected frameworks, we employ the optimal level of protection as determined above and outlined in Table 5.3. The actual video bit rate is obtained by deducting the packetization overhead from the total channel bit rate. Statistics are averaged from twenty simulation runs. The different curves correspond to

- A layered and protected framework whose mode is rate-distortion optimized as proposed herein (CODER I)
- A layered and protected framework whose mode is not rate-distortion optimized (CODER II)
- A layered and unprotected framework whose mode is rate-distortion optimized (CODER III)
- A layered and unprotected framework whose mode is not rate-distortion optimized (CODER IV)
- A non-layered framework whose mode is rate-distortion optimized for error resilient Internet video as proposed in [107, 67] (CODER V)

Packet Loss Rate	Code
0	(31,31)
5	(31,25)
10	(31,23)
20	(31,21)

Table 5.3: Optimal FEC codes for given layered bitstream and packet loss rate.

The results in Figure 5.10(a) show that the proposed framework, CODER I, achieves more than 1 dB improvement in performance over the non-layered framework, CODER V, for packet loss rates greater than 10%, with an improvement of 3 dB at 20%. Compared to the unoptimized and unprotected framework, CODER IV, CODER I achieves more than 3 dB improvement for packet loss rates greater than 10%, with an improvement of 8 dB at 20%. We have already compared the performance of CODER I to the optimized and unprotected framework, CODER III, in Section 5.2.3. Finally, compared to the unoptimized and protected framework, CODER II, the proposed framework achieves more than 1 dB improvement for packet loss rates greater than 10%, with an improvement of 2 dB at 20%. The results in Figure 5.10(b) show that the proposed framework, CODER I, achieves up to 1 dB improvement in performance over CODER V for packet loss rates of 20%. Compared to CODER IV, CODER I achieves more than 6 dB improvement for packet loss rates greater than 10%. Again, we have already compared the performance of CODER I to CODER III in Section 5.2.3. Finally, compared to CODER II, the proposed framework achieves more than 1 dB improvement for packet

loss rates greater than 10%. In all cases, the performance of the proposed framework degrades gracefully with increasing packet loss rate.

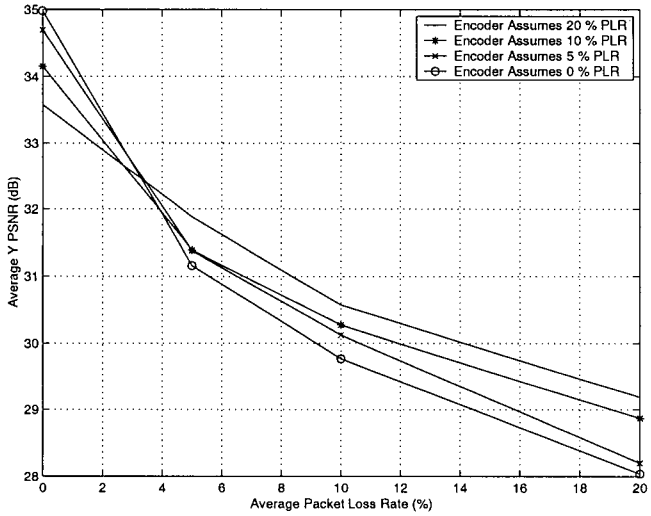
The proposed framework maintains a good performance level as the more reliable base layer information can be used either directly for error concealment or to assist in performing motion compensated error concealment as described in Section 5.2.2. In all cases, informal testing of the improvement in subjective quality of the proposed framework over the other frameworks is quite pronounced. This can be explained by the fact that, while the non-layered framework, CODER V, performs reasonably well based on quantitative results, when a loss occurs that affects any changing area of a picture, the decoded sequence exhibits significant distortion and artifacts that can be quite objectionable. Because the non-layered coding framework selects an optimal amount of intra-updating, it effectively contains the artifacts temporally. However, this does not improve the quality of images for which packet loss occurs.

We provide examples of the reconstruction quality for several frameworks in Figure 5.11. In Figure 5.11(a), a single layered representation generated by CODER V, at 396 kbps and 0% packet loss rate is displayed. At 0% packet loss rate, this corresponds to the best representation that can be obtained at 396 kbps. A single layered representation generated by CODER V is also illustrated in Figure 5.11(c). In this case, the packet loss rate is 20%. Here the artifacts discussed above are quite evident. Because this is a non-layered representation, the only option for the decoder is to perform temporal error concealment. This type of concealment performs poorly for moderate to

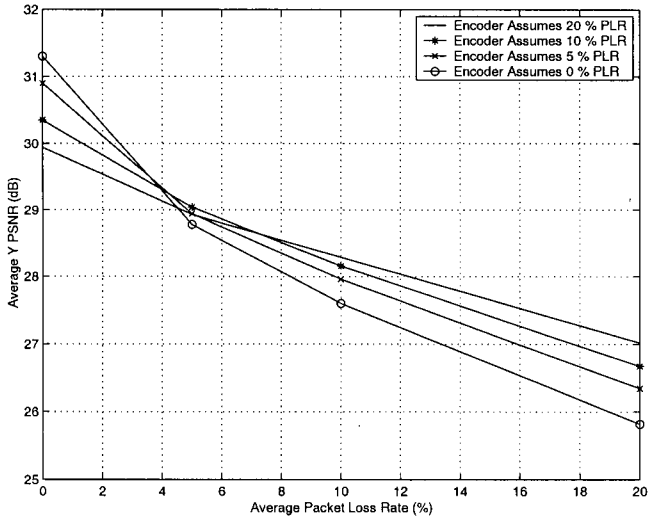
high activity sequences. In Figures 5.11(b) and 5.11(d), representations generated by CODER I and CODER II respectively, at 20% packet loss rate, are displayed. It is evident that a much more stable and acceptable image quality can be obtained using a framework based on layered coding with transport prioritization. Furthermore, the advantages of the rate-distortion optimization algorithm can be seen. The more uniform image quality observed in (b) compared to (d) is due to the algorithm considering the availability of a more reliable base layer reconstruction and increasing the amount of upward prediction.

5.4.3 Effects of Parameter Mismatch

Since the proposed framework is dependent on a number of parameters, we investigate the effects of parameter mismatch. In Figure 5.12 we illustrate the rate-distortion performance versus packet loss rate when the encoder assumes an incorrect packet loss rate. Results are illustrated for two layers of spatial scalability, at QCIF and CIF resolution, using ten seconds of video at ten frames per second for the sequences FOREMAN and COASTGUARD. For each figure, the encoder assumes a packet loss rate of 0, 5, 10 and 20%. We then transport the resulting bitstreams over networks with different actual packet loss rates. Note that when the encoder assumes a packet loss rate of 0 % it is equivalent to error-free rate-distortion optimization. We can observe that mismatch between the assumed and actual packet loss rate affects performance only slightly. There is a maximum of approximately 1.5 dB difference



(a)



(b)

Figure 5.12: PSNR versus packet loss rate for sequence (a) FOREMAN and (b) COASTGUARD with packet loss rate parameter mismatch in mode selection algorithm. Spatial scalability, base layer QCIF, enhancement layer CIF resolution.

in performance between the best and worst case performance for all combinations of assumed and actual packet loss rates. Another observation is that, for the error-free case, when the encoder assumes a lossy network, up to 1.5 dB decrease in performance can occur. However, it is worth noting that such a decrease results only in visible encoding artifacts, as opposed to concealment artifacts, thus it is less displeasing. The decrease in performance when the encoder assumes a packet loss rate that is too low occurs because the rate-distortion optimized mode selection algorithm does not introduce sufficient error-resilience into the video bitstream.

We next investigate the effects of mismatch on the rate-distortion performance of the enhancement layer between the assumed and actual decoder error concealment method. These results are illustrated in Figure 5.13, for two layers of spatial scalability, at QCIF and CIF resolution, using ten seconds of video at ten frames per second for the sequences FOREMAN in (a) and (b) and COASTGUARD in (c) and (d). For each figure, the encoder assumes a particular decoder error concealment method. For (a) and (c) the encoder assumes the median estimate or TCON for the enhancement layer, as we have done above. For (b) and (d) the encoder assumes upward concealment for the enhancement layer. We then transport the resulting bitstreams over networks with packet loss rates of 0, 5, 10, and 20 % and decode them with decoders employing different actual error concealment methods. The first simply copies the lost block from the same spatial location in the previous enhancement layer frame. The second employs the median estimate to perform motion compensated tempo-

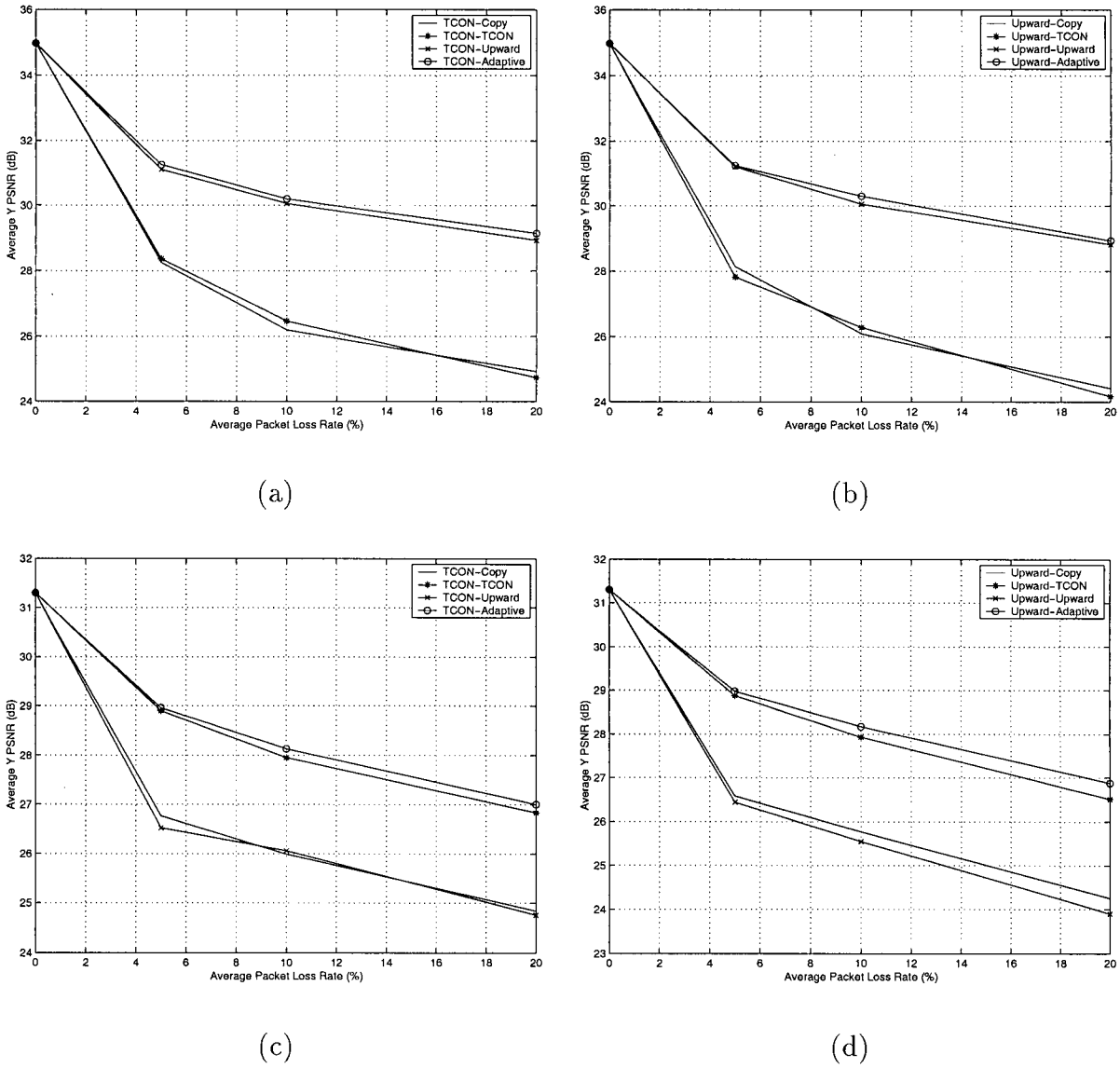


Figure 5.13: PSNR versus packet loss rate for sequence (a) FOREMAN, (b) FOREMAN, (c) COASTGUARD, and (d) COASTGUARD with error concealment method mismatch between encoder and decoder. Spatial scalability, base layer QCIF, enhancement layer CIF resolution.

ral error concealment based on the previous enhancement layer frame. The third conceal a lost enhancement layer macroblock from the corresponding base layer region. The fourth employs the adaptive algorithm introduced in Section 5.2.2 to conceal lost enhancement layer macroblocks. In all cases, independent of the assumed error concealment method at the encoder, the decoder employing the proposed adaptive error concealment method achieves the best performance. Furthermore, it is usually the decoder employing the concealment methods that simply copies the lost block from the same spatial location in the previous enhancement layer frame that produces the worst performance. The results here also correspond to the observations in Section 5.2.2. There we saw that the upward error concealment method performed better for high activity sequences while the forward error concealment method performed better for low activity sequences. There is a maximum of approximately 3 dB difference in performance between the best and worst case performance for all combinations of assumed and actual concealment methods. Thus, we see that it is important that the encoder assume an error concealment method. However, the particular method that is assumed is not as critical as the actual method employed. This is because the assumed method is necessary only for our statistical distortion measure. Any assumed method will cause the measure to have the desired effect of increasing the error resilience of the resulting bitstream. Also, as expected, a better actual error concealment method will yield provide better performance for any assumed error concealment method.

5.5 Conclusion

We have proposed an effective framework for robust Internet video communications based on the principle of layered coding with transport prioritization. This framework and its components are important contributions of our work. The main contribution of this chapter is a rate-distortion optimized mode selection algorithm that selects the optimal amount of temporal error resilience to insert into the source bitstream, using knowledge of the channel packet loss rate, the FEC code rate, and the corresponding decoder error concealment method.

For the framework components, we have proposed an effective packetization scheme for layered bitstreams that minimizes packetization overhead and facilitates decoder error concealment. We have introduced an enhancement layer temporal error concealment method that exploits high reliability base layer information to determine the appropriate course of action for concealment. We have also presented an approach to unequal error protection that uses packet-based FEC. Finally, we have determined that appropriate amount of error protection strength to be applied to the base layer source bitstream depending on the channel packet loss rate.

The proposed framework was demonstrated to achieve significant performance improvement over other layered and non-layered coding frameworks, for a wide range of packet loss rates. The resulting algorithms were shown to produce a significantly improved reconstructed image quality. Also, the performance was shown to degrade gracefully for increasing packet loss rates.

We investigated the effects of parameter mismatch on the proposed framework. We observed that when the encoder assumes an incorrect packet loss rate, the performance can deteriorate by up to 1.5 dB. However, this deterioration is generally visible in the form of coding artifacts as opposed to concealment artifacts. We also observed that mismatch between the error concealment method assumed by the encoder and the actual error concealment method employed by the decoder does not significantly affect performance.

Chapter 6

Conclusions

6.1 Thesis Contributions

In this dissertation, we presented lossy video encoding algorithms for efficient and robust layered video coding and transport in error-free and error-prone networks. We optimized the video encoding mode selection within and between layers, trading off source coding efficiency for bitstream error resilience, based on the local statistics of the video data, the error recovery capability of the channel codec, the error concealment capability of the source decoder, and the expected distortion caused by the channel. For error-free environments, this reduced to selecting parameters that maximized the source coding efficiency.

The most successful low bit rate video coding algorithms were discussed. One particular approach, H.263, was summarized in detail, as it was employed throughout this dissertation as the framework for testing the proposed algorithms. Relevant techniques for layered video encoding, rate-distortion op-

timized video encoding and robust video encoding and transport from the literature were then reviewed.

We evaluated the key technical features of layered video encoding algorithms. We showed that the flexibility to select at the macroblock level the source for prediction, from either the current base layer or previous enhancement layer reconstruction, yielded substantial improvement in compression efficiency over traditional methods. We then proposed an algorithm to improve the source coding efficiency of the resulting layered video encoder. Based on the principles of rate-distortion optimization, the algorithm selected the locally optimal encoding parameters, including motion vectors, coding mode, and quantization level. Next, we presented a model to control the operational mode of a layered video encoder. This model allowed the encoder to compute *a priori* the rate-distortion optimized parameters such that a target bit rate could be achieved.

We then developed a prioritized transport framework for robust layered video communications in error-prone packet-switched networks. We proposed a packetization technique for a layered bitstream that minimizes packetization overhead while facilitating error concealment by the decoder. Results for different video sequences and packet loss rates were presented, demonstrating the superior performance of the proposed method over other packetization methods. We presented an adaptive error concealment method for lost enhancement layer blocks. Within a non-guaranteed but prioritized transport environment, the proposed method exploited information from the current,

more reliable, base layer reconstruction and from the previous, higher quality enhancement layer reconstruction to conceal missing blocks. Results for different video sequences and packet loss rates were presented, demonstrating the superior performance of the adaptive method over traditional methods. Finally we developed a prioritization approach, based on unequal error protection of the individual layer bitstreams. We applied unequal error protection through packet-based forward-error correction. Reed-Solomon codes were applied over a block of video data packets to produce FEC packets.

We then introduced a rate-distortion optimized layered video encoding algorithm for error-prone environments. This algorithm was also based on the principles of rate-distortion optimization and was a generalization of the algorithm introduced for error-free environments. The algorithm incorporated a statistical distortion measure that considered the error recovery capabilities of the channel codec, the channel conditions and the error concealment capabilities of the source decoder to optimize the video encoding mode selection. For these parameters, we employed the prioritization approach and error concealment method developed for the transport framework. Then, for a given layered bitstream and given channel conditions, optimal channel protection code rates were determined. Experimental results demonstrated that the layered encoding algorithm and transport framework provided substantial improvement in reconstructed video quality for a wide range of packet loss rates, for Internet video communications. Moreover, it was demonstrated to yield graceful degradation of reconstructed video quality.

Because the techniques proposed in this thesis are fully standard compliant, they can immediately benefit industry applications, particularly those in the area of multi-point Internet video communications. Moreover, they outperform the state-of-the-art in terms of the efficiency of DCT-based low bit rate video encoding algorithms, the efficiency of layered video encoding algorithms, and the robustness of Internet video communications.

To summarize, the main contributions of this thesis are

- A rate-distortion optimized layered video encoding algorithm for error-free environments. The algorithm was demonstrated to achieve a significant improvement in rate-distortion performance.

- A model to control the Lagrangian parameter $\lambda(1)$, for SNR and spatial scalability, using the quantization parameter. This model was shown to significantly reduce encoding complexity while maintaining essentially the same rate-distortion performance as an exhaustive optimal bit allocation that employed the bisection search algorithm.

- A framework for robust Internet video communications based on the principle of layered coding with transport prioritization. The framework components include

  - An effective packetization scheme for layered bitstreams. The scheme minimizes packetization overhead and facilitates decoder error concealment.

  - An enhancement layer temporal error concealment method. The

method exploits high reliability base layer information to determine the appropriate course of action for concealment.

  - A packet-based FEC mechanism to achieve unequal error protection. This mechanism was studied to determined the appropriate amount of error protection strength to be applied depending on the channel packet loss rate.

- A rate-distortion optimized layered video encoding algorithm for error-prone environments. The algorithm incorporates a statistical distortion measure and knowledge of the proposed framework parameters to optimize the video encoding mode selection. This algorithm was demonstrated to provide significant improvement in reconstructed video quality for a wide range of packet loss rates.

6.2 Future Research Directions

This thesis has addressed two very significant research areas, error resilient layered video encoding and prioritized transport. Layered and prioritized video encoding frameworks are expected to become more widespread in order to satisfy the non-uniformity and sub-optimality of the current network infrastructure. In fact, layered video encoding capabilities are being included in emerging video encoding standards in order to satisfy the growing demand for streaming applications.

From a purely source coding perspective, it is interesting to note that

the current trend has been towards sacrificing encoding efficiency in order to provide fine granularity in the degree of scalability. This effectively means that forward or motion compensated prediction is not permitted within the enhancement layer. We expect that improved coding efficiency of such approaches will be a popular topic of study. For example one approach that could be investigated is to arrange the block DCT coefficients of the residual or "error" image into a uniform subband structure and then apply well-known subband coding techniques. However, given the loss of coding efficiency in the enhancement layer, we do not believe that such approaches will gain widespread acceptance. Clearly, some form of ability to predict the current enhancement layer signal from previously decoded enhancement layer signals is necessary to achieve reasonable coding efficiency. We believe that methods for improving these prediction models are more deserving of further attention. For example, compared to two-layered scalability using unoptimized H.263 scalability [67], both our approach and that proposed in [35] have been demonstrated to yield improved coding efficiency. While our approach selects the source for prediction at the macroblock level, the approach in [35] does so at the pixel level. It is possible that the flexibility to employ an intermediate amount of granularity in choosing the source for prediction, for example on 4×4 or 8×8 blocks of pixels, would yield improved rate-distortion tradeoffs.

From an error-resilient source coding perspective, we expect that more contributions will include a optimized frameworks such as the one presented in this thesis. However, better error propagation models for motion compensated

and transform-based layered video coding frameworks, such as the one recently proposed in [115], are needed. A model that accurately describes how packet loss affects spatial and temporal error propagation could replace the statistical distortion measure we develop in Section 5.3.1. This model could then be folded into our rate-distortion optimized layered mode selection algorithm to more appropriately introduce bit stream error resilience. Furthermore, an analytical framework for optimizing the tradeoff between source and channel coding for motion compensated and transform-based layered video encoding and transport frameworks are needed. Such a method has recently been introduced for non-layered video encoding and transport in [116]. Unfortunately, this method employs a model that cannot be derived from commonly used statistical measures such as variance and correlation. Instead, its parameters must be estimated by fitting the model to a subset of measured data points from the actual rate-distortion curve.

Subjective video quality assessment is another important research area. Objective measures, such as PSNR, are still more widely used than subjective measures in the video coding research community. While subjective assessment yields accurate results, its main premise is the use of human observers. This results in a costly and time consuming process. Moreover, it is impossible to employ subjective assessment for the in-service continuous monitoring of video quality. It would be extremely beneficial to both the image and video coding research communities to have an objective method to measure subjective quality. Such a method would of course have to be well-behaved, accurate,

and consistently well-correlated to actual subjective assessments. Furthermore, such a method would have to consider the new types of visual artifacts introduced due to the proliferation of digitally compressed video systems. Currently, there is an effort within the video coding standardization community to define such a measure [101]. In the initial phase, ten methods were proposed and evaluated. Interestingly, none of these methods was demonstrated statistically to outperform PSNR as an objective measure of subjective quality. We expect that these methods will be refined and additional proposals will be made in the next phase of testing. However, this area will remain extremely active for quite some time.

From a channel coding perspective, scalable channel coding is one further method to provide flexible error resilience. In the same manner that individual receivers can receive different levels of video quality, they could also receive different levels of channel coding protection. As one example, using our proposed framework, if the base layer of video was protected with a (31,21) code, individual receivers could receive only five FEC packets when the additional FEC was unnecessary, permitting them to recover from up to five packet losses. Under poor conditions, receivers could choose to receive all ten FEC packets.

Finally, we summarize the above research directions for layered video encoding and transport frameworks:

1. Better source models for improved coding efficiency of enhancement layer or residual data.

2. Better models for error propagation due to packet loss and motion compensation.

3. An analytical framework to optimize the tradeoff between source and channel coding.

4. An objective method to measure subjective quality of digitally compressed video.

5. A layered channel coding framework for scalable error recovery.

Bibliography

[1] ITU Telecom. Standardization Sector of ITU, "Video Coding for Low Bitrate Communication," *ITU-T Recommendation H.263 Version 2*, January 1998.

[2] ISO/IEC JTC1/SC29/WG11 N2202, *Coding of audio-visual objects: Video.* ISO/IEC, March, 1998.

[3] C. E. Shannon, "A mathematical theory of communication," *Bell Systems Technical Journal 27*, pp. 379–423 & 623–656, July 1948.

[4] S. Wenger, G. Knorr, J. Ott, and F. Kossentini, "Error resilience support in H.263+," *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 8, pp. 867–877, Nov. 1998.

[5] R. Talluri, "Error resilient video coding in the MPEG-4 standard," *IEEE Comm. Magazine*, vol. 26, pp. 112–119, June 1998.

[6] Y. Wang and Q. Zhu, "Error control and concealment for video communication: A review," *Proceeding of IEEE*, vol. 86, pp. 974–997, May 1998.

[7] A. Ortega and K. Ramchandran, "Rate-distortion methods for image and video compression," *IEEE Signal Proc. Magazine*, vol. 15, pp. 23–50, Nov. 1998.

[8] M. Effros, "Optimal modelling for complex system design," *IEEE Signal Proc. Magazine*, vol. 15, pp. 51–73, Nov. 1998.

[9] G. J. Sullivan, "Multi-hypothesis motion compression for low bit-rate video coding," in *Proc. IEEE Int. Conf. Acoust., Speech, and Signal Processing*, vol. V, pp. 437–440, 1993.

[10] B. Girod, "Motion compensating prediction with fractional-pel accuracy," *IEEE Trans. on Communications*, vol. 41, pp. 604–612, Apr. 1993.

[11] J. Ribas-Corbera and D. Neuhoff, "Optimizing motion vector accuracy in block-based video coding," *IEEE Trans. on Circuits and Systems for Video Technology*, vol. submitted for publication, Mar. 1999.

[12] H. Musmann, "Advances in picture coding," *Proc. of the IEEE*, vol. 73, pp. 523–548, Apr. 1985.

[13] M. Orchard and G. Sullivan, "Overlapped block motion compensation: an estimation-theoric approach," *IEEE Trans. on Image Processing*, vol. 3, pp. 693–699, Sept. 1994.

[14] M. Karczewicz, J. Nieweglowski, and P. Haavisto, "Video coding using motion compensation with polynomial motion vector fields," *Signal Processing: Image Communication*, vol. 10, pp. 63–91, 1997.

[15] K. P. Rao and P. Yip, *Discrete Cosine Transfroms:Algorithms, Advantages, Applications.* New York: Academic Press, 1990.

[16] M. Vetterli and J. Kovacevic, *Wavelets and Subband Coding.* Upper Saddle River, NJ: Prentice-Hall, 1995.

[17] V. Bhaskaran and K. Konstantinides, *Image and Video Compression Standards: Algorithms and Architecture.* Boston: Kluwer Academic Publishers, 1995.

[18] A. Gersho, "Optimal nonlinear interpolative vector quantization," *IEEE Trans. on Communications*, pp. 1285–1287, Sept. 1990.

[19] N. Ahmed, T. Natarajan, and K. Rao, "Discrete cosine transform," *IEEE Transactions Comput.*, vol. C-23, pp. 90–93, 1974.

[20] W. H. Chen, C. H. Smith, and S. C. Fralick, "A fast computational algorithm for the discrete cosine transform," *IEEE Trans. on Communications*, vol. COM-25, pp. 1004–1009, Sept. 1977.

[21] K. R. Rao and J. J. Hwang, *Techniques and Standards for Image, Video and Audio Coding.* Upper Saddle River, NJ: Prentice-Hall, 1996.

[22] J. J. N. Jayant and R. Safranek, "Signal compression based on models of human perception," *Proceedings of the IEEE*, vol. 81, pp. 1385–1422, Oct. 1993.

[23] T. I. Telegraph and T. C. Committee, "Video codec for autiovisual services at p $\times$ 64 kbits/s; Recommendation H.261," 1990.

[24] B. Girod, E. Steinbach, and N. Faerber, "Comparison of the H.263 and H.261 video compression standards," in *Standards and Common Interfaces for Video Information Systems, K.R. Rao,editor, Critical reviews of optical science and technology*, vol. 60, (Philadelphia, Pennsylvania), pp. 233–251, Oct. 1995.

[25] N. F. B. Girod, E. Steinbach, "Performance of the H.263 video compression standard," *To Appear in Journal of VLSI Signal Processing: Systems for Signal, Image, and Video Technology, Special Issue on Recent Development in Video: Algorithms, Implementation and Applications*, 1997.

[26] J. Wen and J. Villasenor, "A class of reversible variable length codes for robust image and video coding," in *International Conference on Image Processing*, (Santa Barbara, CA), Oct. 1997.

[27] M. Ghanbari, "Two-layer coding of video signals for VBR networks," *IEEE Journal on Selected Areas in Communications*, vol. 7, pp. 771–781, June 1989.

[28] D. Wilson and M. Ghanbari, "Optimization of two-layer SNR scalability for MPEG-2 video," in *Proc. IEEE Int. Conf. Acoust., Speech, and Signal Processing*, pp. 2637–2640, 1997.

[29] M. Ghanbari, "An adapted H.261 two-layer video codec for ATM networks," *IEEE Trans. on Communications*, vol. 40, pp. 1481–1490, Sept. 1992.

[30] D. Taubman and A. Zakhor, "Multirate 3-D subband coding of video," *IEEE Trans. on Image Processing*, vol. 3, pp. 572–588, Sept. 1994.

[31] J. Y. Tham, S. Ranganath, and A. Kassim, "Highly scalable wavelet-based video codec for very low bit-rate environment," *IEEE Journal on Selected Areas in Communications*, vol. 16, pp. 12–27, Jan. 1998.

[32] ISO/IEC JTC1/SC29/WG11, "Verification Model of ISO/IEC 14496-2 MPEG-4 Video Fine Granularity Scalability v4.0," (N3317, 51st MPEG meeting, Noordwijkerhout, NL), Mar. 2000.

[33] U. Horn, B. Girod, and B. Belzer, "Scalable video coding with multiscale motion compensation and unequal error protection," in *Proceedings of the International Symposium on Multimedia Communications and Video Coding*, (New York, USA), Oct. 1995.

[34] U. Horn and B. Girod, "Performance analysis of multiscale motion compensation techniques in pyramid coders," in *International Conference on Image Processing*, vol. 3, pp. 255–258, 1996.

[35] K. Rose and S. L. Regunathan, "Towards optimal scalability in predictive video coding," in *International Conference on Image Processing*, vol. 3, (Chicago, Illinois, USA), pp. 929–933, Oct. 1998.

[36] C. I. Podilchuk, N. S. Jayant, and N. Farvardin, "Three dimensional subband coding of video," *IEEE Trans. on Image Processing*, vol. 4, pp. 125–138, Feb. 1995.

[37] Q. Wang and M. Ghanbari, "Scalable coding of very high resolution video using the virtual zerotree," *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 7, pp. 719–727, Oct. 1997.

[38] K. Shen and E. J. Delp, "Wavelet based rate scalable video compression," *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 9, pp. 109–122, Feb. 1999.

[39] G. Côté, B. Erol, M. Gallant and F. Kossentini, "H.263+: Video coding at low bit rates," *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 8, pp. 849–866, Nov. 1998.

[40] M. Walker and M. Nilsson, "A study of the efficiency of layered coding using H.263," in *Packet Video '99*, (New York, NY, USA), Apr. 1999.

[41] L. Yang, F. C. Martins, and T. R. Gardos, "Improving H.263+ scalability performance for very low bit rate applications," in *SPIE Proc. Visual Communications and Image Processing*, vol. 3653, (San Jose, CA, USA), pp. 768–779, Jan. 1998.

[42] C. E. Shannon, "Coding theorems for a discrete source with a fidelity criterion," in *IRE National Convention Record, Part 4*, pp. 142–163, 1959. Also in *Information and Decision Processes*, R. E. Machol, Ed. New York, NY: McGraw-Hill, 1960, pp. 93-126.

[43] T. Berger and J. Gibson, "Lossy source coding," *IEEE Transactions on Information Theory*, vol. IT-44, pp. 2693–2723, Oct. 1998.

[44] T. Berger, *Rate Distortion Theory.* New Jersey: Prentice-Hall, Inc., 1971.

[45] T. Cover and J. Thomas, *Elements of Information Theory.* New York: John Wiley and Sons, Inc, 1991.

[46] H. E. III, "Generalized Lagrange multiplier method for solving problems of optimum allocation of resources," *Operation Research*, vol. 11, pp. 399–417, 1963.

[47] Y. Shoham and A. Gersho, "Efficient bit allocation for an arbitrary set of quantizers," *IEEE Trans. on Acoustics, Speech, and Signal Processing*, vol. 36, pp. 1445–1453, Sept. 1988.

[48] P. A. Chou, T. Lookabaugh, and R. M. Gray, "Entropy-constrained vector quantization," *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. ASSP-37(1), pp. 31–42, Jan. 1989.

[49] K. Ramchandran and M. Vetterli, "Best wavelet packet bases in a rate-distortion sense," *IEEE Trans. on Image Processing*, vol. 2, pp. 160–174, April 1993.

[50] T. Wiegand, M. Lightstone, D. Mukherjee, T. Campbell, and S. Mitra, "Rate-distortion optimized mode selection for very low bit rate video coding and the emerging H.263 standard," *IEEE Trans. on Circuits and Systems for Video Technology*, pp. 182–190, Apr. 1996.

[51] S. Haykin, *Adaptive Filter Theory.* Englewood Cliffs, New Jersey: Prentice-Hall, 1986.

[52] J. Choi and D. Park, "A stable feedback control of the buffer state using the Lagrangian multiplier method," *IEEE Transactions on Image Processing: Special Issue on Image Sequence Compression*, vol. 3, pp. 546–558, Sept. 1994.

[53] Y. Lee, F. Kossentini, M. Smith, and R. Ward, "Predictive RD-optimized motion estimation for very low bit rate video coding," *IEEE Journal on Selected Areas in Communications*, vol. 15, pp. 1752–1763, Dec. 1997.

[54] G. Sullivan and T. Wiegand, "Rate-distortion optimization for video compression," *IEEE Signal Proc. Magazine*, pp. 74–90, Nov. 1998.

[55] G. J. Sullivan and R. L. Baker, "Rate-distortion optimized motion compensation for video compression using fixed or variable size blocks," in *Global Telecomm. Conf. {GLOBECOM'91}*, pp. 85–90, Dec. 1991.

[56] T. Wiegand, M. lightstone, T. Campbell, and S. Mitra, "Efficient mode selection for block-based motion compensated video coding," in *ICIP95*, (Washington, DC, USA), Oct. 1995.

[57] G. D. Forney, Jr., "The Viterbi algorithm," *Proc. IEEE*, vol. 61, pp. 268–278, March 1973.

[58] M. C. Chen and A. N. Wilson, "Rate-distortion optimal motion estimation algorithm for video coding," in *Proc. IEEE Int. Conf. Acoust.,*

Speech, and Signal Processing, (Atlanta, USA), pp. 2096–2099, May 1996.

[59] M. C. Chen and A. N. W. Jr., "Rate-distortion optimal motion estimation algorithms for motion compensated transform video coding," *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 8, pp. 147–158, Apr. 1998.

[60] A. Schuster and A. Katsaggelos, "A video compression scheme with optimal bit allocation between displacement vector field and displaced frame difference," in *Proc. IEEE Int. Conf. Acoust., Speech, and Signal Processing*, (Atlanta, USA), pp. 1967–1970, May 1996.

[61] A. Schuster and A. Katsaggelos, "A video compression scheme with optimal bit allocation among segmentation, motion, and residual error," *IEEE Trans. on Image Processing*, vol. 6, pp. 1487–1502, Nov. 1997.

[62] A. Schuster and A. Katsaggelos, "A theory for the optimal bit allocation between displacement vector field and displaced frame difference," *IEEE Journal on Selected Areas in Communications*, vol. 15, pp. 1739–1751, Dec. 1997.

[63] W. Chung, F. Kossentini, and M. Smith, "An efficient motion estimation technique based on a rate-distortion criterion," in *ICASSP96*, vol. 4, (Atlanta, GA, USA), pp. 1926–1929, May 1996.

[64] A. Gersho and R. M. Gray, *Vector Quantization and Signal Compression.* Boston: Kluwer Academic Publishers, 1992.

[65] G. Sullivan and T. Wiegand, "Efficient scalar quantization of exponential and Laplacian random variables," *IEEE Trans. on Information Theory*, vol. 42, pp. 1365–1374, Sept. 1996.

[66] S.-W. Wu and A. Gersho, "Enhanced video compression with standardized bit stream syntax," in *Proc. IEEE Int. Conf. Acoust., Speech, and Signal Processing*, vol. I, (Minneapolis, MN, USA), pp. 103–106, Apr. 1993.

[67] ITU Telecom. Standardization Sector of ITU, "Video Codec Test Model Near-Term, Version 11 (TMN11), Release 2," *H.263 Test-Model Ad Hoc Group*, October 1999.

[68] K. Ramchandran and M. Vetterli, "Rate-distortion optimal fast thresholding with complete MPEG/JPEG decoder compatibility," *IEEE Trans. on Image Processing*, vol. 3, pp. 700–704, Sept. 1994.

[69] J. Wen, M. Luttrell, and J. Villasenor, "Trellis-based R-D optimal quantization in H.263+," *IEEE Trans. on Image Processing*, submitted for publication 1998.

[70] K. Ramchandran, A. Ortega, and M. Vetterli, "Bit allocation for dependent quantization with applications to to multiresolution and MPEG video coders," *IEEE Trans. on Image Processing*, vol. 3, pp. 533–545, Sept. 1994.

[71] A. Katsaggelos, F. Ishtiaq, L. P. Kondi, M.-C. Hong, M. Banham, and J. Brailean, "Error resilience and concealment in video coding," in *Eu-*

ropean Signal Processing Conference, EUSIPCO-98, (Rhodes, Greece), pp. 221–228, Sept. 1998.

[72] J. Liao and J. Villasenor, "Adaptive intra update for video coding over noisy channels," in *International Conference on Image Processing*, (Lausanne, Switzerland), Sept. 1996.

[73] G. Côté and F. Kossentini, "Optimal intra coding of blocks for robust video communication over the Internet," *EURASIP Journal for Image Communication, Special Issue on Real-time Video over the Internet*, vol. 15, pp. 25–34, Sept. 1999.

[74] N. Faerber and B. G. E. Steinbach, "Robust H.263 compatible transmission for mobile video server access," in *International Workshop on Wireless Image/Video Communications*, (Loughborough, U.K.), pp. 8–13, Sept. 1996.

[75] H. Schulzrinne, S. Casner, R. Frederick, and V. Jacobson, "RTP: A transport protocol for real-time applications," *RFC 1889*, Jan. 1996. Available from ftp://ftp.isi.edu/in-notes/rfc1889.txt.

[76] C. Zhu, "RTP payload format for H.263 video streams," *RFC 2190*, Sept. 1997. Available from ftp://ftp.isi.edu/in-notes/rfc2190.txt.

[77] C. Bormann, L. Cline, G. Deisher, T. Gardos, C. Maciocco. D. Newell, J. Ott, G. Sullivan, S. Wenger and C. Zhu, "RTP payload format for the 1998 version of ITU-T rec. H.263 video (H.263+)," *RFC 2429*, May 1998. Available from ftp://ftp.isi.edu/in-notes/rfc2429.txt.

[78] G. Karlsson and M. Vetterli, "Packet video and its integration into network architecture," *IEEE Journal on Selected Areas in Communications*, vol. 7, pp. 739–751, June 1989.

[79] K. Ramchandran, A. Ortega, K. M. Uz, and M. Vetterli, "Mutiresolution broadcast of digital HDTV using joint source/channel coding," *IEEE Journal on Selected Areas in Communications*, vol. 11, pp. 6–23, Jan. 1993.

[80] G. Ungerboeck, "Channel coding with multilevel/phase signals," *IEEE Trans. Inform. Theory*, vol. IT-28, pp. 55–67, January 1982.

[81] L. H. Kieu and K. N. Ngan, "Cell-loss concealment techniques for layered video codecs in an ATM network," *IEEE Trans. on Image Processing*, vol. 3, pp. 666–677, Sept. 1994.

[82] R. Aravind, M. R. Civanlar, and A. R. Reibman, "Packet loss resilience of MPEG-2 scalable video coding algorithms," *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 6, pp. 426–435, Oct. 1996.

[83] ISO/IEC 13818-2—ITU-T Rec. H.262, *Generic Coding of Moving Pictures and Associated Audio Information: Video.* ISO/IEC, 1995.

[84] F. Kishino, K. Manabe, Y. Hayashi, and H. Yasuda, "Variable bit rate coding of video signals for ATM networks," *IEEE Journal on Selected Areas in Communications*, vol. 7, pp. 801–806, June 1989.

[85] Y. Chen, K. Sayood, and D. Nelson, "A robust coding scheme for packet video," *IEEE Trans. on Communications*, vol. 40, pp. 1491–1501, Sept. 1992.

[86] Q-F Zhu and Y. Wang and L. Shaw, "Coding and cell-loss recovery in DCT-based packet video," *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 3, pp. 248–258, June 1993.

[87] ISO/IEC 11172-2: Video, *Coding of Moving Pictures and Associated Audio for Digital Storage Media at up to about 1.5 Mbit/s.* ISO/IEC, 1991.

[88] T. Kinoshita, T. Nakahashi, and M. Maruyama, "Variable bit rate HDTV codec with ATM cell loss compensation," *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 3, pp. 230–237, June 1993.

[89] J. D. Villasenor and D. S. Park, "Proposed draft text for the H.263 Annex V data partitioned slice mode," in *Q15-I-14, ITU-T Q15/SG16*, (Red Bank, New Jersey), Oct. 1999.

[90] D. Park, J. Park, J. Kim, and Y. Kim, "Error-resilient video coding in H.263+ against error-prone mobile channels," in *SPIE Proc. Visual Communications and Image Processing*, vol. 3653, (San Jose, CA, USA), pp. 200–207, Jan. 1999.

[91] R. Talluri, I. Moccagatta, Y. Nag, and G. Cheung, "Error concealment by data partitioning," *Signal Processing: Image Communications Magazine*, vol. 14, pp. 505–518, May 1999.

[92] S. Wicker, *Error Control Systems for Digital Communication and Storage.* Toronto: Prentice Hall Canada Inc., 1995.

[93] H. Ota and T. Kitami, "A cell loss recovery method using FEC in ATM networks," *IEEE Journal on Selected Areas in Communications*, vol. 9, pp. 1471–1482, Dec. 1991.

[94] E. Biersack, "Performance evaluation of forward error correction in an ATM environment," *IEEE Journal on Selected Areas in Communications*, vol. 11, pp. 631–640, May 1993.

[95] S. Wenger, "Video redundancy coding in H.263+," in *Audio-Visual Services over Packet Networks*, (Scotland, UK), Sept. 1997.

[96] P. Haskell and D. Messerschmitt, "Resynchronization of motion compensated video affected by ATM cell loss," in *Proc. IEEE Int. Conf. Acoust., Speech, and Signal Processing*, vol. 3, (San Francisco, CA, USA), pp. 545–548, Mar. 1992.

[97] N. Naka, S. Adachi, M. Saigusa, and T. Ohya, "Improved error resilience in mobile audio-visual communications," in *IEEE International Conference on Universal Personal Communications*, vol. 1, (Tokyo, JAPAN), pp. 702–706, Nov. 1995.

[98] M. Wada, "Selective recovery of video packet loss using error concealment," *IEEE Journal on Selected Areas in Communications*, vol. 7, pp. 807–814, June 1989.

[99] H. R. Rabiee, H. Radha, and R. L. Kashyap, "Error concealment of still image and video stream with multi-directional recursive non-linear filters," in *Proc. IEEE Int. Conf. Acoust., Speech, and Signal Processing*, (Atlanta, USA), pp. 37–40, May 1996.

[100] ISO/IEC 10918-1—ITU-T Rec. T.81, *Digital Compression and Coding of Continuous-tone Still Images: Requirements and Guidelines.* ISO/IEC, 1994.

[101] P. Corriveau, J. Lubin, J. C. Pearson, and A. Webster, "Video quality experts group: Current results and future directions," in *SPIE Visual Communications and Image Processing*, (Perth, Australia), June 2000.

[102] P. Kuhn, "A highly portable instruction level profiler," *available via anonymous ftp to ftp.lis.e-technik.tu-muenchen.de/pub/iprof.*

[103] M. Gallant, G. Côté, and F. Kossentini, "Description of and results for rate-distortion optimized coder," in *Q15-D-49, ITU-T Q15/SG16*, (Tampere, Finland), Apr. 1998.

[104] T. Wiegand and B. Andrews, "An improved H.263 coder using rate-distortion optimization," in *Q15-D-13, ITU-T Q15/SG16*, (Tampere, Finland), Apr. 1998.

[105] J. Ribas-Corbera and S. Lei, "Optimal quantizer control in DCT video coding for low-delay video communications," in *Picture Coding Symposium*, (Berlin, Germany), Sept. 1997.

[106] G. Côté, S. Shirani, and F. Kossentini, "Optimal mode selection and synchronization for robust video communications over error prone networks," *IEEE Journal on Selected Areas in Communications*, vol. 18, pp. 952–965, June 2000.

[107] S. Wenger and G. Côté, "Using RFC2429 and H.263+ at low to medium bit-rates for low-latency applications," in *Packet Video '99*, (New York, NY, USA), Apr. 1999.

[108] J. Ott and S. Wenger, "Application of H.263+ video coding modes in lossy packet network environments," *EURASIP Journal for Visual Communications*, 1998. Accepted for publication.

[109] V. Parthasarathy, J. Modestino, and K. Vastola, "Design of a transport coding scheme for high-quality video over ATM networks," *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 7, pp. 358–376, Apr. 1997.

[110] J. Rosenberg and H. Schulzrinne, "An RTP payload format for generic forward error correction," *RFC 2733*, Dec. 1999. Available from ftp://ftp.isi.edu/in-notes/rfc2733.txt.

[111] J. M. Boyce and R. D. Gaglianello, "Packet loss effects on MPEG video sent over the public internet," in *ACM MULTIMEDIA 98*, (Bristol, UK), Sept. 1998.

[112] M. Handley, "An examination of mbone performance," *UCL/ISI Research Report*, Jan. 1997.

[113] S. Wenger, "Proposed error patterns for internet experiments," *ITU-T Study Group 16 H.263+ Video Experts Group*, vol. Q15I09, Oct. 1999.

[114] G. de los Reyes, A. Reibman, J. Chuang, and S. F. Chang, "Video transcoding for resilience in wireless channels," in *International Conference on Image Processing*, (Chicago, Illinois, USA), Oct. 1998.

[115] G. de los Reyes, A. Reibman, S. F. Chang, and J. Chuang, "Error-resilient transcoding for video over wireless channels," *IEEE Journal on Selected Areas in Communications*, vol. 18, pp. 1063–1074, June 2000.

[116] K. Stuhlmuller, N. Farber, M. Link, and B. Girod, "Analysis of video transmission over lossy channels," *IEEE Journal on Selected Areas in Communications*, vol. 18, pp. 1012–1032, June 2000.