UNIVERSITY OF CALIFORNIA
Santa Barbara

ρ-Domain Rate-Distortion Analysis and Rate Control for Visual Coding and Communication

A Dissertation submitted in partial satisfaction of the requirements for the degree of

Doctor of Philosophy

in

Electrical and Computer Engineering

by

Zhihai He

Committee in charge:

Professor Sanjit K. Mitra, Chairperson
Professor Edward Chang
Professor Kenneth Rose
Doctor Amir Said

June, 2001

The dissertation of Zhihai He is approved:

Commitee Chairperson

April, 2001

April, 2001

To my wife, Xiuzhen, for her constant support and love,
my parents, for their encouragement, guidance, and love.

ACKNOWLEDGEMENTS

I am extremely grateful to Professor Sanjit K. Mitra, chairman of my thesis committee, for his excellent guidance, constant encouragement, and support during the course of this research. I am also very thankful to Professor B. S. Manjunath and Professor K. Rose for treating me like one of their own Ph.D students. I would like to thank Prof. Edward Chang and Dr. Amir Said for many valuable suggestions and assistance to my graduate research at UCSB.

I take this opportunity to thank Dr. Tian-Hu Yu for his assistance and Dr. Yong Kwan Kim for his cooperation during my graduate research work.

I thank the current members of the Image Processing and Vision Research Laboratories: Sitaram Bhagavathy, Qingwei Cheng, Motaz El-Saban, Mylene Farias, Gabriel Gomez, Serkan Hatipoglu, Luca Lucchese, Michael Moore, Shawn Newsam, Baris Sumengen, Xinding Sun, Jelena Tesic, and Peng Wu; former members: Yining Deng, Rajeev Gandhi, Debargha Muhkerjee, and all other former or short-term students and visitors. They have all made my life and work at UCSB an enjoyable experience.

I would also like to thank my wife Xiuzhen who had been at my side in a supporting role giving me all the love I needed. Finally, I would like to thank my parents who, from day one of my education, have provided tremendous support, constant encouragement, and great guidance.

VITA

March 1973	Born, Hunan, P. R. China.
July 1994	**Bachelor of Science** Department of Mathematics, Beijing Normal University, Beijing, P. R. China.
July 1997	**Master of Science** Institute of Computational Mathematics, Chinese Academy of Sciences, Beijing, P. R. China.
August 1997 – June 1998	Department of Applied Mathematics, University of Iowa.
October 1998 – June 1999	Teaching Assistant Department of Electrical and Computer Engineering, University of California, Santa Barbara.
October 1999 – May 2001	Research Assistant Department of Electrical and Computer Engineering, University of California, Santa Barbara.
June 2000 – September 2000	Employed as a summer intern in the Digital Design group at National Semiconductor Corporation, Santa Clara, California.
June 2001	**Doctor of Philosophy** Department of Electrical and Computer Engineering, University of California, Santa Barbara.

PUBLICATIONS

[1] Zhihai He and S. K. Mitra, "Optimum quantization error feedback filter," *Proceedings of International Conference on Image Processing*, September 2000.

[2] Zhihai He, Tian-Hu Yu, and Sanjit K. Mitra, "Blockwise zeros mapping image coding," *Proceedings of International Conference on Image Processing*, September 2000.

[3] Tian-Hu Yu, Zhihai He, and S. K. Mitra, "Simple and efficient wavelet image compression," *Proceedings of International Conference on Image Processing*, September 2000.

[4] Zhihai He and Bernd Meyer, "National Semiconductor Corporation MPEG-4 video compression for wireless video applications," *National Semiconductor Corporation Technical Report,* September 2000.

[5] Zhihai He, Tian-Hu Yu, and S. K. Mitra, "A fast, accurate and forward rate prediction and control algorithm for wavelet image coders," *Proceedings of 34th Asilomar Conference on Signals, Systems, and Computers,* October 2000.

[6] Tian-Hu Yu, Zhihai He, and S. K. Mitra, "A novel coding scheme for wavelet image compression," *Proceedings of 34th Asilomar Conference on Signals, Systems, and Computers,* October 2000.

[7] Zhihai He, Tian-Hu Yu, and S. K. Mitra, "Wavelet image coding using trellis coded quantization and blockwise binary classification," *Proceedings of 34th Asilomar Conference on Signals, Systems, and Computers,* October 2000.

[8] Zhihai He, YongKwan Kim, and Sanjit K. Mitra, "Low-delay rate control and smoothing for video coding via p-domain source modeling," accepted by *IEEE Transactions on Circuits and System on Video Technology*, October 2000.

[9] Zhihai He, YongKwan Kim, and Sanjit K. Mitra, "A novel source modeling framework for bit rate and picture quality control in DCT visual coding," accepted by *Picture Coding Symposium 2001,* January 2001.

[10] Zhihai He and Sanjit K. Mitra, "A linear source model and a unified rate control algorithm for H.263/MPEG-2/ MPEG-4," accepted by *International Conference on Acoustics, Speech, and Signal Processing,* February 2001.

[11] Zhihai He, YongKwan Kim, and Sanjit K. Mitra, "Rate control for video encoding via p-domain source modeling," accepted by *International Conference on Acoustics, Speech, and Signal Processing,* February 2001.

[12] Zhihai He and Sanjit K. Mitra, "Novel rate-distortion analysis framework for bit rate and picture quality control in DCT visual coding," accepted by *IEE Proceedings - Vision, Image and Signal Processing*, February 2001.

[13] Zhihai He and Sanjit K. Mitra, "Computational source model for wavelet image coding," submitted to *IEEE Transactions on Image Processing*, March 2000.

[14] Zhihai He and Sanjit K. Mitra, "Fast and Accurate Rate Prediction and Control for JPEG Coding of Color Images," submitted to *IEEE Transactions on Circuits and System on Video Technology*, April 2000.

[15] Zhihai He and Sanjit K. Mitra, "A linear source model and a unified rate control algorithm for DCT video coding," submitted to *IEEE Transactions on Circuits and System on Video Technology*, September 2000.

[16] Zhihai He and Sanjit K. Mitra, "ρ-domain bit allocation and rate control for real-time video coding," submitted to *Internation Conference on Image Processing 2001*, January 2001.

[17] Zhihai He, YongKwan Kim, and Sanjit K. Mitra, "Rate-distortion optimized MPEG-4 video coding and rate control," submitted to *IEEE Transactions on Circuits and System on Video Technology*, March 2001.

ABSTRACT

ρ-Domain Rate-Distortion Analysis and Rate Control for Visual Coding and Communication

by

Zhihai He

Rate-distortion (R-D) analysis and rate control play a key role in video coding and communication systems by providing the R-D optimized compression performance, assuring the successful network transmission of the coded video data, and achieving the best visual quality at the receiver. In the conventional R-D analysis, the bit rate R and distortion D are considered as functions of the quantization parameter q. That is to say, the source models are developed in the q-domain. These source models either have very high computational complexity or suffer from relatively large estimation and control error. In this dissertation, a new framework for R-D analysis, called ρ-domain analysis, is developed, where R and D are studied as functions of ρ which is the percentage of zeros among the quantized transform coefficients. We observe that in the ρ-domain the R-D functions have unique properties which enable us to model and estimate them accurately and robustly. First, we show that in any typical transform coding systems the rate function in the ρ-domain, denoted by $R(\rho)$, is a linear function. Based on Shannon's source coding theorem, we provide a theoretical justification for the ρ-domain linear rate model. Based on this rate model, a unified rate control algorithm is developed for all standard video coding systems, such as MPEG-2, H.263, and MPEG-4 video coding. Our extensive experimental results show that the proposed rate control algorithm outperforms other algorithms reported in the literature by providing much more accurate and robust rate control and buffer regulation. Within the framework of ρ-domain analysis, a generic distortion model is also developed for transform coding of images and videos. Based on this distortion model and the linear rate model described

above, an optimum bit allocation scheme is developed in the ρ-domain. It is applied to the MPEG-4 coding at the object level and to the H.263 coding at the macroblock level. Experimental results are presented to show that the proposed optimum bit allocation significantly improves the coded picture quality.

To estimate the R-D functions without context information, we develop a unified source modeling framework in the ρ-domain by introducing the new concepts of characteristic rate curves and rate curve decomposition. With this framework, the R-D functions of the image/video encoder can be accurately estimated before quantization and coding with very low complexity. Based on the estimated R-D functions, a frame-level rate control algorithm is developed for video coding which outperforms the macroblock-level standard TMN8 rate control algorithm. In addition, an encoder-based rate shape smoothing algorithm is developed to control the video encoder such that the output bit stream has both a smoothed rate shape and a consistent picture quality. This is highly desirable in practical video coding and communications.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Recent advances in computing and communication technology stimulate the research interest in digital techniques for recording and transmitting visual information. The exponential growth in the amount of visual data to be stored, transferred, and processed has created a huge need for data compression. Compression of visual data, such as images and videos, can significantly improve the utilization efficiency of the limited communication channel bandwidth or storage capacity. For example, in cable TV applications [1], higher resolution TV programs can be delivered with the required bandwidth if the program video data is compressed before transmission. In communications over the Internet, images and videos consume the most bandwidth. Here, efficient data compression is necessary to reduce the impact and burden on the network traffic.

The demand for image and video compression has triggered the development of several compression standards, such as JPEG [2, 3], JPEG-2000 [4, 5], MPEG-2 [6, 7], H.263 [8, 9], and MPEG-4 [10]. The JPEG standard is essentially designed for the compression of continuous-tone still images. It can also be used to encode video sequences, where it is often called motion-JPEG. In this case, it makes no use of the

temporal correlation between the neighboring frames. The MPEG-2 video compression standard aims to provide generic methods for coding moving pictures for such applications as storing videos on video CD (VCD) and on digital versatile disk (DVD). It has also become a popular compression tool for high definition TV (HDTV) and digital video broadcasting (DVB) [11]. H.263 is mainly designed and widely used for video conferencing [9]. This standard targets real-time video coding at very low bit rates with low delay latency. The recently finalized JPEG-2000 image coding standard is fundamentally different from the JPEG coding standard. Specifically, JPEG is based on the blockwise discrete cosine transformation (DCT) [12] while JPEG-2000 employs the discrete wavelet transform (DWT) and subband coding techniques [13, 14]. It provides significantly improved coding efficiency and greater functionality, such as rate scalability, error resilience, and compression domain manipulation of regions of interest (ROI) [5].

1.1 Problem Addressed

In both the compression standards and the algorithms reported in the literature [16, 18, 19], transform coding has become the dominant approach for image and video compression [15]. A generic transform coding system is depicted in Fig. 1.1. The transform, either DWT or DCT, is applied to the input picture. Here, a picture can be either a still image or a motion-compensated video frame. After quantization, the quantization coefficients are converted into symbols according to some data representation scheme. For example, zig-zag scan and run-level data representation are employed in JPEG and MPEG coding [2, 6]. In embedded zero-tree wavelet (EZW) coding [16], all insignificant coefficients in a spatial orientation tree are represented by one zero-tree symbol. After data representation, the output symbols are finally encoded by a Huffman, or

arithmetic coder [17].

Many coding algorithms have been reported in the literature. Some are well known and widely used, such as wavelet-based EZW, set partitioning in hierarchical trees (SPIHT) [18] and stack-run [19] image coding, DCT-based JPEG image coding [2], MPEG-2 [6], H.263 [9] and MPEG-4 [10] video coding. In this dissertation, we will take these coding algorithms for samples to investigate the R-D behavior of transform coding of images and videos. Hence, we refer to them as *typical transform coding systems*.

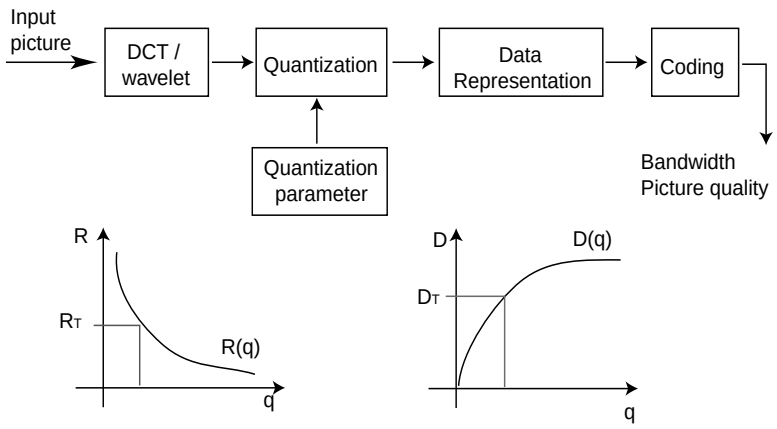

Figure 1.1: A generic transform coding system for images and videos.

In visual coding and communication, the two most important factors are the coding bit rate and picture quality. The coding bit rate, denoted by R, determines the channel bandwidth required to transfer the coded visual data. One direct and widely used measure for the picture quality is the mean square error between the coded image/video and the original one. The reconstruction error introduced by compression, often referred as distortion, is denoted by D. In typical transform coding, both R and D are controlled by the quantization parameter of the quantizer, denoted by q. The major issue here is how to determine the value of q to achieve the target coding bit rate, or picture quality. To do this, we need to analyze and estimate the R-D behavior of

the image/video encoder; this behavior is characterized by its rate-quantization (R-Q) and distortion-quantization (D-Q) functions, denoted by $R(q)$ and $D(q)$, respectively [20, 21]. In this dissertation, they are collectively called *R-D functions* or *curves*. Based on the R-D functions, as shown in Fig. 1.1, the quantization parameter q can be readily determined to achieve the target bit rate R_T or picture quality D_T [22, 23]. Therefore, the major issue here becomes this: how to analyze and estimate the R-D functions for the image/video encoder.

Analysis and estimation of the R-D functions have potential applications in visual coding and communication. First, with estimated R-D functions we can adjust the quantization setting of the encoder and control the output bit rate or picture quality according to the channel condition, the storage capacity, or the user's requirement [23, 24, 25, 26]. Second, based on the estimated R-D functions, optimum bit allocation as well as other R-D optimization procedures can be performed to improve the efficiency of the coding algorithm and, consequently, improve the image quality or video presentation quality [27, 28, 29, 30].

1.2 Related Works

In the classic R-D analysis [20, 21], the coding bit rate is approximated by the entropy of the quantized coefficients. However, in transform coding of images and videos, especially at very low bit rates, there is a large mismatch between the theoretical entropy and the actual coding bit rate. Fig. 1.2 compares the theoretical entropy and the actual coding bit rate for images "Lena" and "Peppers" coded by JPEG. We can see that the relative error between them is very large. This is because the theoretical entropy formulation does not take the efficient JPEG coding algorithm into account. Since it is difficult to develop a close-form expression to model directly the coding

algorithms such as EZW, SPIHT, JPEG, or MPEG [38], the empirical approach is often employed in the R-D models and rate control algorithms reported in the literature.

Many R-D estimation and control algorithms have been developed within the context of video coding. Some of them have been adopted as international standards; typical of these are the MPEG-2 Test Model Version 5 (TM5) rate control algorithm [32], the H.263 Test Model Near-term Version 8 (TMN8) algorithm [33], and the MPEG-4 Verification Model Version 8 (VM8) algorithm [34, 35]. Since they are widely used in practical coding applications and regarded as the state-of-the-art rate control algorithms, we will provide a brief review of them in Chapter 2. Throughout this dissertation we will also use them for performance comparisons with the proposed algorithms. Besides these standard rate control algorithms, many other algorithms have been proposed to target different applications. A parametric R-D model has been proposed by Tao et al. [36] for frame-level MPEG video coding. An approach based on a normalized parametric R-D model [37] has been developed for H.263-compatible video codecs. The authors claim that the proposed model offers an efficient approach that requires less memory to approximate the rate and distortion characteristics for all quantization parameters.

Among the bits-quantization models used in these R-D analysis and control algorithms, some are based on the modified version of the classical R-D functions which lead to logarithmic expressions [22, 23, 36, 39]. Mathematical expressions of other types, such as power [24], spline [38], and polynomial [35, 40, 41], have also been employed. It can be seen that these R-D models have complicated and highly nonlinear expressions. In addition, for different coding algorithms, the R-D models and control algorithms are quite different from each other.

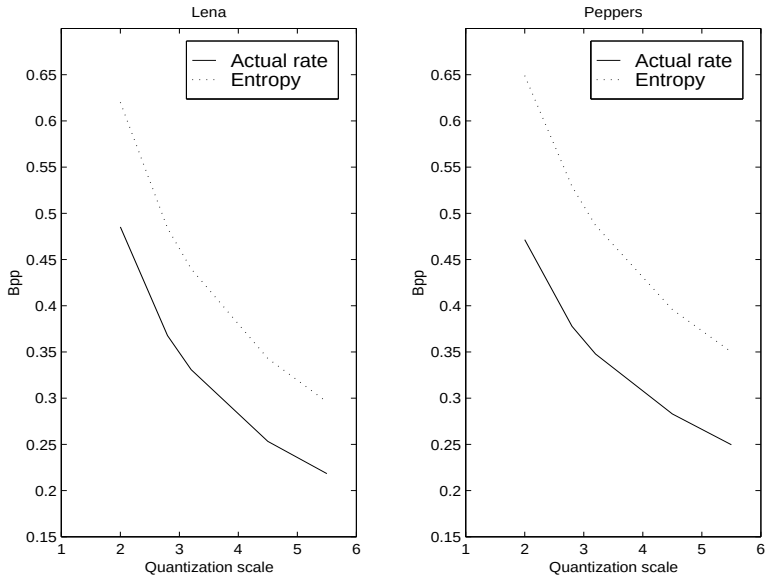

Figure 1.2: Comparison between the theoretical entropy and the actual coding bit rate for Lena and Peppers coded by JPEG.

1.3 Objective

The objective of this dissertation to develop a unified framework for analysis, estimation and control of the R-D behavior of typical transform coding systems. Towards this objective, three major issues are addressed: unified rate modeling and control, optimum bit allocation, and forward estimate of the R-D functions.

A. Unified Rate Modeling and Control

The rate models reported in the literature try to use some statistics of the input source data, such as variance, to describe the input image or video data [20, 22, 23]. They also try to analyze and model each step of the coding algorithms and formulate an explicit expression of the coding bit rate. To achieve high coding performance, an efficient coding algorithm must often employ a sophisticated data representation scheme as well as an entropy coding scheme. To model these coding algorithms more accurately, these rate models are getting more and more complex [22, 23, 39, 40]. However, with complex

and highly nonlinear expressions, the estimation and rate control process becomes increasingly complicated and even unstable with the image-dependent variation [38] .

As mentioned in the previous section, for different coding algorithms, the R-D models and rate control algorithms reported in the literature are quite different. It would be ideal to develop a simple, accurate, and unified rate model for any typical transform coding systems. Based on this simple model, we could then develop a unified rate control algorithm which could be applied to any typical transform coding system. To this end, we need to uncover the common rules that govern the R-D behaviors of all transform coding systems. Obviously, this will provide us with valuable insights into the mechanism of transform coding. Practically, the simple and unified rate model and control algorithm would enable us to control the image/video encoder accurately and robustly with very low computational complexity and implementation cost.

B. Optimum Bit Allocation

Without accurate R-D model, the optimum bit allocation cannot be carried out [24]. In other words, if the model itself is not accurate, the bit allocation result cannot be truly optimum. Therefore, to achieve the best performance in the optimum bit allocation, we need to have an accurate and robust estimate of the R-D functions. In issue A addressed above, we set the goal to develop an accurate and robust rate model. Therefore, the only remaining issue is to develop a simple and accurate distortion model and, consequently, to develop an optimum bit allocation scheme. Note that the rate control algorithm discussed in issue A is mainly designed to achieve the target coding bit rate. When coupled with the optimum bit allocation scheme, its functionality is further extended.

C. Forward R-D Estimation

In video coding, the rate control algorithm often employs a parametric source model

whose parameters can be adaptively estimated using the coding statistics of the previous video frames or macroblocks (MBs) [23, 24, 35]. However, for still image coding, the typical coding algorithms (such as JPEG, EZW and Stack-Run) do not use the adaptive quantization scheme [2, 16, 19]. Therefore, the adaptive estimation of the model parameters cannot be carried out. To estimate the R-D functions without context information, some operational approaches have been proposed in the literature [38, 24]. In these approaches, the image is often coded several times. The coding statistics are then used to estimate the model parameters. Obviously, they have very high computational complexity. In addition, they do not provide us with insights into the R-D behaviors of the coding systems [22]. Little has been done in the literature to develop a general theoretical framework which can provide accurate and forward estimation of the R-D functions for the image/video encoder. In this dissertation, we set the goal to develop a unified source modeling framework which allows fast and accurate estimation of the R-D functions for any typical transform coding systems. Here, the estimation should be done before quantization and coding which implies it is forward. In addition, the estimation process should be much less complex than the actual quantization and coding process.

1.4 Approach

To accomplish the objective described in Section 1.3, we propose a novel methodology for R-D analysis which is called *ρ-domain analysis*. Explained briefly here, it will be discussed in detail in Chapter 2. In the R-D models presented in the literature, the rate R and distortion D are treated as functions of the quantization parameter q, denoted by $R(q)$ and $D(q)$, respectively. Studying rate and distortion as functions of q is called *q-domain analysis*. In this way, we say that the conventional source models are based

on the q-domain analysis.

It is observed that zeros play a key role in transform coding, especially at low bit rates [42, 43]. All typical coding algorithms treat zeros in a special way and address most of the effort to efficient coding of them. For example, in JPEG and MPEG coding, run length representation and a special symbol of end-of-block (EOB) are employed to code the zeros [2, 6]. In H.263 video coding, a special binary flag named "LAST" is introduced to signal that all the remaining coefficients in a zig-zag order are zeros [9]. In EZW, a zero-tree symbol is introduced to code all of the zeros in a spatial orientation tree [16]. After the transform coefficients are quantized with q (as shown in Fig. 1.1), let ρ be the percentage of zeros among the quantized coefficients. Note that in typical transform coding systems ρ monotonically increases with q. (A detailed description of this monotonical relationship between q and ρ will be presented in Chapter 2.) Hence, there is a one-to-one mapping between them. This implies that, mathematically R and D are also functions of ρ, denoted by $R(\rho)$ and $D(\rho)$. Studying the rate and distortion as functions of ρ is called ρ-domain analysis.

As we will see in subsequent chapters, in the ρ-domain the rate and distortion functions have unique behaviors. After being mapped from the q-domain into the ρ-domain, the picture-dependent variation and the highly nonlinear behaviors of the R-D functions have been significantly removed. This enables us to analyze, model, and estimate them accurately and robustly in the ρ-domain, and consequently to accomplish the objective set in the previous section.

1.5 Summary of Contributions

The main accomplishments of this dissertation are as follows:

- Development of a new methodology for R-D analysis, called ρ-domain analysis. It

serves an alternative to the traditional q-domain analysis. We demonstrate that, after mapping from the q-domain to the ρ-domain, the image-dependent variation and highly nonlinear behaviors of the R-D functions have been significantly removed.

- Development of a linear rate model in the ρ-domain. Based on our extensive simulation results, we show that it is a unified rate model for any typical transform coding system. With Shannon's source coding theorem [20, 21], we give a theoretical proof for the linear rate model. Also, we have shown that the model parameter which is the slope of the linear rate function is directly related to the image content.

- Development of a unified rate control algorithm for all typical video coding systems, such as MPEG-2, H.263, and MPEG-4 video coding. The proposed algorithm has very low computational complexity and implementation cost. When compared to other rate control algorithms reported in the literature, our rate control algorithm provides much more accurate and robust rate control and buffer regulation. In addition, the improved picture quality is due to its accurate rate model and robust selection of the quantization parameter.

- Development of a distortion model and an optimum bit allocation scheme in the ρ-domain. The proposed bit allocation scheme is applied to efficiently distribute the bit budget among different video objects, which leads to a scalable rate control algorithm for MPEG-4 video coding. Coupled with macroblock classification, the bit allocation scheme is also applied to H.263 video coding to improve the coded picture quality.

- Development of a novel ρ-domain source modeling framework for transform coding of images and videos. It allows us to estimate the R-D functions before

quantization and coding with very low complexity. The relative estimation error is less than 5%. Based on the estimated R-D functions, the output bit rate or picture quality of the still image encoder is accurately controlled. A frame-level rate control algorithm is also proposed for video coding. In addition, with the forward estimation of the R-D functions, a rate shape smoothing algorithm is developed. This smoothing algorithm allows the encoder to produce a bit stream that has both a smoothed rate shape and a consistent picture quality.

1.6 Dissertation Outline

The dissertation is organized as follows:

Chapter 2 summarizes the typical transform coding systems employed in image/video coding as well as some state-of-the-art source modeling and rate control algorithms. These coding systems are to be extensively used in the subsequent chapters. The existing rate control algorithms are referred to and compared with the algorithms developed in this dissertation. After the background review, the concept of the ρ-domain analysis is formally introduced and further explained. Its implementation details are then discussed.

Chapter 3 shows that, for any typical transform coding system, the rate curve in the ρ-domain is approximately linear based on our extensive simulation results. A theoretical justification for the linear rate model is provided. The physical meaning of the model parameter, which is the slope of the linear function, is also discussed.

Chapter 4 presents a unified rate control algorithm for MPEG-2, H.263, and MPEG-4 video coding. Complexity analysis is provided to show that the proposed algorithm has very low computational complexity. The implementation details are discussed. Extensive experimental results and comparisons with other rate control algorithms are

provided.

Chapter 5 introduces a distortion model in the ρ-domain. Based on this distortion model and the linear rate model developed in Chapter 3, an optimum bit allocation scheme in the ρ-domain is developed. The proposed bit allocation scheme further extends the capability of the rate control algorithm presented in Chapter 4. It is applied to MPEG-4 object-based video coding and H.263 coding to improve the coded video quality.

Chapter 6 presents a unified source modeling framework for all typical transform coding systems. The linear rate model and distortion model, presented respectively in Chapters 3 and 5, both have model parameters which can be adaptively estimated in video coding. To explicitly estimate the R-D functions before quantization and coding for transform coding (especially for still images coding), a novel ρ-domain source modeling framework is introduced in this chapter. It is based on the new concepts of characteristic rate curves and rate curve decomposition. With the estimated R-D functions, the output bit rate of the image encoder can be accurately controlled. For video coding, a frame-level rate control algorithm is proposed. Another application of the estimated R-D functions is encoder-based rate shape smoothing. A smoothing algorithm is developed to control the encoder such that the output bit stream has both a smoothed rate shape and a consistent picture quality.

Chapter 7 summarizes the studies presented in this dissertation. Concluding remarks are provided, and future directions are discussed.

Chapter 2

Transform Coding and ρ-Domain R-D Analysis

Transform coding has become a dominant approach for image and video compression. Many efficient transform coding algorithms have been proposed in the literature, such as wavelet-based EZW, SPIHT and stack-run image coding, DCT-based JPEG coding, MPEG-2, H.263, and MPEG-4 video coding. In this dissertation, these coding systems are referred to as typical transform coding systems. To control the R-D behavior of the image/video encoder, many rate control algorithms have been developed. Some have been adopted as international standards, such as MPEG-2 TM5, H.263 TMN8, and MPEG-4 VM8 rate control algorithms. In this chapter, following a brief overview of the typical transform coding systems and the standard rate control algorithms, the ρ-domain R-D analysis is formally introduced and explained in detail. Its software implementation is also discussed.

2.1 Typical Transform Coding Systems

According to the type of the input source data (either images or videos), and the type of transform employed in the coding scheme, the typical transform coding systems can be categorized into the three groups: wavelet-based image coding, DCT-based image coding, and DCT-based video coding. In the following, we give a brief review of these coding systems.

2.1.1 Wavelet-Based Image Coding

EZW, SPIHT and stack-run are three examples of the wavelet-based image coding systems considered in our investigation. DWT is employed in these systems because of its superior performance in energy compaction [50, 51]. After multiple-level dyadic subband decomposition, the wavelet coefficients are uniformly quantized. Note that, although SPIHT and EZW are progressive coding algorithms, they actually employ an implicit uniform quantization scheme. In other words, we can also apply the EZW and SPIHT data representation schemes to the quantized wavelet coefficients after the uniform quantization.

To efficiently code the quantization coefficients, EZW explores the correlation among the coefficients across different scales along a certain direction. During the coding process, the magnitudes of the wavelet coefficients are compared with a threshold T, which is reduced by half in the next coding pass. A coefficient is termed significant if its magnitude is larger than T. Otherwise, it is insignificant. If a coefficient is insignificant and all of its descendents are also insignificant, they are jointly represented by one symbol called *zero-tree root* (ZTR). The introduction of the ZTR symbol greatly improves the coding efficiency [16, 44]. In SPIHT, the coding efficiency is further enhanced by a sophisticated set partitioning algorithm [18]. However, both are based on the concept of efficient prediction of the insignificance information of the wavelet coefficients

[18, 45]. The outputs of the EZW and SPIHT algorithms are further compressed by an arithmetic encoder.

In the stack-run coding proposed in [19], the non-zero coefficients and the run length of zeros are represented by symbols from a quaternary intermediate alphabet. After wavelet transform and uniform quantization, the wavelet coefficients are rearranged into a 1-D array according to a raster scan order. Each non-zero coefficient is represented in a binary form with symbols {'0', '1'}. For differentiation, each run length number of zeros is represented in binary form with symbols {'–', '+'} instead of {'0', '1'}. After the binary representation, the output symbol stream is further compressed by a first-order adaptive arithmetic encoder. Despite its very low addressing and implementation complexity, this method performs better than EZW and is competitive with SPIHT. But, its coding performance will degrade at relatively high bit rates, especially when applied to some images with a lot of detail, such as "Barbara" [19].

2.1.2 DCT-Based Image Coding

The DCT-based JPEG image coding have been used successfully in various image coding applications, such as Web publishing, medical imaging, image transmission, and digital cameras. The input picture is first partitioned into 8×8 blocks and each block is coded separately. After DCT, the coefficients are quantized by a perceptual quantization scheme. A detailed description of the JPEG quantization scheme is given in Section 2.4. The 64 quantized DCT coefficients are rearranged into a 1-D array in a zig-zag scan order. Each non-zero coefficient is jointly encoded with its preceding zeros according to a precomputed Huffman table. The zeros at the end of the block are represented by the "EOB" symbol [2]. JPEG is a general-purpose compression standard that has relatively low computational complexity and a low memory requirement.

2.1.3 DCT-Based Video Coding

So far, the most popular coding algorithms for video sequences with successful applications are based on motion compensation (MC) and DCT. MPEG-2, H.263, and MPEG-4 are the three major international standards for video coding. In a video sequence, two major types of correlation are explored by the coding algorithm. The first is the temporal correlation among the neighboring video frames. This is explored by motion estimation and compensation [6, 8]. The other type of correlation is the spatial correlation which is explored by the DCT. After motion compensation and DCT, in MPEG-2 the DCT coefficients are coded in a way similar to the coding method in JPEG. In H.263, the coding efficiency is significantly improved by replacing the EOB symbol in JPEG with a binary flag and a 3-D variable length coding (VLC) scheme [8, 9]. MPEG-4 video coding introduces the concept of video object. Each video frame is segmented into several video objects. Each object is then coded separately, as in H.263 [10, 46, 47]. For detailed treatment of these video coding standards, see [6, 9, 10].

2.2 Standard Rate Control Algorithms

In video coding, rate control is employed to control the output bit rate of the video encoder according to the network condition and to improve the video presentation quality [23, 35]. As mentioned in Chapter 1, the key task in rate control and R-D optimization is to analyze and model the R-D behavior of the image/video encoder. In the following, we provide a brief review of three well-known rate control algorithms which have been adopted as international standards and are widely used in practical applications. Performance of these algorithms will be compared to our algorithms proposed in the subsequent chapters.

2.2.1 TM5 Rate Control Algorithm

The TM5 rate control algorithm is designed for bit rate control in MPEG-2 video coding. In the MPEG-2 coding syntax, the input video sequence is segmented into groups of pictures (GOPs). The first frame of each GOP is intracoded and called I-frame. The rest are either prediction-coded frames (P-frames) or bidirectionally interpolated frames (B-frames) [6]. The TM5 rate control algorithm consists of two major steps. In the first step, the target bit rate for each video frame inside the GOP is obtained by a frame-level bit allocation scheme. In the second step, the quantization parameter for each macroblock is determined from the buffer status and the spatial activity of the macroblocks. The whole rate control algorithm is based on the following assumptions [32, 48]:

1. The distortion D increases linearly with the quantization parameter q.

2. To maintain a consistent video presentation quality, the quantization parameters for I, P, and B frames, denoted as q_I, q_P, q_B, are related by

$$\frac{q_I}{1.0} = \frac{q_P}{k_P} = \frac{q_B}{k_B}, \tag{2.1}$$

   where k_P and k_B are constants. By default, they are set to be 1.0 and 1.4, respectively.

3. The coding bit rate R is inversely proportional to the distortion D. In other words,

$$R \cdot D = constant. \tag{2.2}$$

It can be seen that extremely simplified R-D models are employed in the TM5 algorithm. Therefore, it cannot achieve accurate and robust rate control [48].

2.2.2 VM8 Rate Control Algorithm

In video coding, the coding statistics of previous frames provides valuable information for estimating the R-D behavior of the current frame, since the characteristics of the neighboring frames are very close to each other when the picture activity is low. The VM8 [34] rate control algorithm designed for MPEG-4 video coding is based on this observation. It employs the following assumptions in its R-D analysis [34, 49]:

1. Neighboring video frames of the same type are very similar to each other, and have the same rate and distortion curves.

2. The rate curve $R(q)$ of each video frame is approximated by a quadratic formula,

$$R(q) = a_1 \times q^{-1} + a_2 \times q^{-2}. \tag{2.3}$$

   After a video frame is encoded, the average quantization parameter and the total coding bit rate are known. Such coding statistics of a number of previous frames are then used to estimate the model parameters a_1 and a_2 for the current frame. Once the model parameters are obtained by linear regression, the rate model given by Eq. (2.3) can be applied for rate control.

Due to the first assumption, the VM8 rate control algorithm often suffers from severe performance degradation at scene changes, because this assumption actually no longer holds at scene changes. Besides this, the VM8 algorithm also suffers from relatively large control error due to the limited accuracy and robustness of its rate model given by Eq. (2.3) [23].

2.2.3 TMN8 Rate Control Algorithm

The TMN8 rate control algorithm operates at the macroblock level. The coding statistics of previous macroblocks are utilized to update the model parameter for the current

macroblock. The TMN8 algorithm is based on the following logarithmic R-Q model,

$$R(q) = \begin{cases} \frac{1}{2}\log_2(2e^2\frac{\sigma^2}{q^2}), & if \quad \frac{\sigma^2}{q^2} > \frac{1}{2e}, \\ \frac{e}{\ln 2}\frac{\sigma^2}{q^2}, & if \quad \frac{\sigma^2}{q^2} \leq \frac{1}{2e}, \end{cases} \tag{2.4}$$

which is a modified version of the classical R-D formula [20, 21]. Compared to the VM8 rate control algorithm, the TMN8 algorithm can meet the target bit rate much more accurately, and maintain a much steadier buffer level [23]. However, in the TMN8 algorithm, there is no regulation on the dynamic range of the quantization parameter of each macroblock. In addition, because of the limited accuracy of its rate model, the algorithm also suffers from relatively large control error, especially at scene changes in active videos at low coding bit rates.

The TMN8 rate control algorithm is originally designed for the P-frames in H.263 video coding. For I-frames, the algorithm employs very rough rate and distortion models [23]. However, in practical video coding applications, to improve the error robustness or allow flexible playback, frequent selection of the intra coding mode is needed. Therefore, the TMN8 algorithm has limited practical applications.

2.3 ρ-Domain Analysis

All of the rate control algorithms described in the previous section, as well as other algorithms reported in the literature [22, 24, 36, 37, 38, 39, 48], study the R-D functions in the q-domain. To improve the estimate accuracy, the expression for the coding bit rate is getting more and more complex. In this dissertation, we develop a novel framework for R-D analysis which provides a totally new viewpoint of the source modeling and rate control problem.

It is well known that zeros play a very important role in transform coding, especially at low coding bit rates. The state-of-the-art coding algorithms (such as EZW, SPIHT,

JPEG, MPEG, and H.263), put most of their efforts on the efficient coding of zeros. As mentioned in Chapter 1, the percentage of zeros among the quantized transform coefficients, denoted by ρ, has a substantial effect on the R-D behavior of the image/video encoder. Also, there is a one-to-one mapping between q and ρ. Therefore, we can map the R-D functions in the q-domain $R(q)$ and $D(q)$ into the ρ-domain and denote them as $R(\rho)$ and $D(\rho)$. As we will see in the subsequent chapters, in the ρ-domain, the R-D functions have unique properties.

2.3.1 Typical Quantization Schemes

To map the R-D functions between the q-domain and the ρ-domain, we first need to obtain the one-to-one mapping between q and ρ. Note that this mapping is determined by the quantization scheme. In the following, we briefly review the quantization schemes employed by the typical transform coding systems before discussing the computation of the mapping between q and ρ.

Quantization in Wavelet Image Coding

In the typical wavelet-based image coding systems, uniform threshold quantization (UTQ) is often used. In this case, the quantization parameter q refers to the UTQ stepsize. Let Δ be the UTQ dead zone threshold. In general, Δ is proportional to q. For any transform coefficient x, its UTQ output index is given by

$$I[x] = UTQ[q, \Delta; x] = \begin{cases} 0 & \text{if} \quad |x| \leq \Delta; \\ \lceil \frac{x-\Delta}{q} \rceil & \text{if} \quad x > +\Delta; \\ \lfloor \frac{x+\Delta}{q} \rfloor & \text{if} \quad x < -\Delta. \end{cases} \tag{2.5}$$

H.263 Quantization Scheme

The quantization scheme employed by H.263 video coding is similar to UTQ. To be more specific, the quantization index of x in the H.263 style quantization scheme is given by

$$I[x] = \begin{cases} \text{Round}(\frac{x}{8}) & \text{if } x \text{ is a DC coefficient in an intra-MB;} \\ UTQ(2q, 2q;\, x) & \text{if } x \text{ is an AC coefficient in an intra-MB;} \\ UTQ(2q, 2.5q;\, x) & \text{if } x \text{ is a coefficient in an inter-MB.} \end{cases} \tag{2.6}$$

Note that the range of the unquantized DC coefficient is 0 to 2040, which implies the range of its differential value is –2040 to 2040. In H.263 coding, it is quantized by a uniform quantizer with fixed step size 8, as shown in Eq. (2.6).

JPEG Quantization Scheme

In JPEG still image coding, a perceptual quantization scheme is employed. Each of the 64 DCT coefficients is quantized by a different uniform quantizer (UQ). The actual step sizes for the coefficients in the luminance component are associated with a quantization matrix, denoted by $[w_J^l(i,j)]_{1\le i,j\le 8}$, where

$$[w_J^l(i,j)] = \begin{bmatrix} 16 & 11 & 10 & 16 & 24 & 40 & 51 & 61 \\ 12 & 12 & 14 & 19 & 26 & 58 & 60 & 55 \\ 14 & 13 & 16 & 24 & 40 & 57 & 69 & 56 \\ 14 & 17 & 22 & 29 & 51 & 87 & 80 & 62 \\ 18 & 22 & 37 & 56 & 68 & 109 & 103 & 77 \\ 24 & 35 & 55 & 64 & 81 & 104 & 113 & 92 \\ 49 & 64 & 78 & 87 & 103 & 121 & 120 & 101 \\ 72 & 92 & 95 & 98 & 112 & 100 & 103 & 99 \end{bmatrix}. \tag{2.7}$$

Let $x(i,j)$ be the DCT coefficient located at (i,j) inside a luminance block. Its quantization output is given by

$$I[x(i,j)] = \text{Round}\left[\frac{x(i,j)}{q \cdot w_J^l(i,j)}\right], \tag{2.8}$$

where the quantization parameter q functions as a scaling factor which controls the coding bit rate and the picture quality. If $x(i,j)$ is from a chrominance block, $w_J^l(i,j)$ in Eq. (2.8) is then replaced by the chrominance quantization matrix $w_J^c(i,j)$ where

$$[w_J^c(i,j)] = \begin{bmatrix} 17 & 18 & 24 & 47 & 99 & 99 & 99 & 99 \\ 18 & 21 & 26 & 66 & 99 & 99 & 99 & 99 \\ 24 & 26 & 56 & 99 & 99 & 99 & 99 & 99 \\ 47 & 66 & 99 & 99 & 99 & 99 & 99 & 99 \\ 99 & 99 & 99 & 99 & 99 & 99 & 99 & 99 \\ 99 & 99 & 99 & 99 & 99 & 99 & 99 & 99 \\ 99 & 99 & 99 & 99 & 99 & 99 & 99 & 99 \\ 99 & 99 & 99 & 99 & 99 & 99 & 99 & 99 \end{bmatrix}. \tag{2.9}$$

MPEG Quantization Scheme

In MPEG-2 coding, the JPEG-style perceptual quantization scheme is employed. The quantization matrixes for intracoded and intercoded macroblocks, respectively denoted

by $w_M^0(i,j)$ and $w_M^1(i,j)$, are given in the following,

$$[w_M^0(i,j)] = \begin{bmatrix} 8 & 16 & 19 & 22 & 26 & 27 & 29 & 34 \\ 16 & 16 & 22 & 24 & 27 & 29 & 34 & 37 \\ 19 & 22 & 26 & 27 & 29 & 34 & 34 & 38 \\ 22 & 22 & 26 & 27 & 29 & 34 & 37 & 40 \\ 22 & 26 & 27 & 29 & 32 & 35 & 40 & 48 \\ 26 & 27 & 29 & 32 & 35 & 40 & 48 & 58 \\ 26 & 27 & 29 & 34 & 38 & 46 & 56 & 69 \\ 27 & 29 & 35 & 38 & 46 & 56 & 69 & 83 \end{bmatrix}, \tag{2.10}$$

$$[w_M^0(i,j)] = \begin{bmatrix} 16 & 16 & 16 & 16 & 16 & 16 & 16 & 16 \\ 16 & 16 & 16 & 16 & 16 & 16 & 16 & 16 \\ 16 & 16 & 16 & 16 & 16 & 16 & 16 & 16 \\ 16 & 16 & 16 & 16 & 16 & 16 & 16 & 16 \\ 16 & 16 & 16 & 16 & 16 & 16 & 16 & 16 \\ 16 & 16 & 16 & 16 & 16 & 16 & 16 & 16 \\ 16 & 16 & 16 & 16 & 16 & 16 & 16 & 16 \\ 16 & 16 & 16 & 16 & 16 & 16 & 16 & 16 \end{bmatrix}. \tag{2.11}$$

Unlike JPEG quantization, in the same macroblock, both the luminance and chrominance component use the same quantization matrix. In MPEG-2 coding, the quantization index of the DCT coefficient $x(i,j)$ is given by,

$$I[x(i,j)] = \begin{cases} \text{Round}(\frac{x}{8}) & \text{if } x \text{ is a DC coefficient in an intra-MB;} \\ \text{Round}\left[\frac{16 \cdot x(i,j)}{2 \cdot q \cdot w_M^0(i,j)}\right] & \text{if } x \text{ is an AC coefficient in an intra-MB;} \\ \text{Round}\left[\frac{16 \cdot x(i,j)}{2 \cdot q \cdot w_M^1(i,j)}\right] & \text{if } x \text{ is a coefficient in a non-intra MB.} \end{cases} \tag{2.12}$$

In MPEG-4 standard, both the H.263 style and the MPEG-2 style quantization are adopted. The user needs to configure the encoder to choose the quantization scheme.

It can be seen that all the quantization schemes listed in the above are very close to the uniform threshold quantization.

2.3.2 The Mapping Between q and ρ

The one-to-one mapping between q and ρ can be directly computed from the distribution information of the transform coefficients. This is because in all typical transform coding systems each transform coefficient is quantized separately. In the following, we describe in detail how to compute the one-to-one mapping between q and ρ for different coding systems.

Wavelet Image Coding

The wavelet-based image coding schemes such as EZW and stack-run (SR) employ the uniform threshold quantization scheme given by Eq. (2.5). Let the distribution of the wavelet coefficients be $\mathcal{D}(x)$. Here, we assume $\mathcal{D}(x)$ is a positive continuous function. After quantization, the percentage of zeros among the quantized transform coefficients is given by

$$\rho(q) = \frac{1}{M} \int_{-\Delta}^{+\Delta} \mathcal{D}(x)\, dx, \tag{2.13}$$

where M is the image size.

H.263 Coding

The H.263 quantization scheme is given by Eq. (2.6). Let $\mathcal{D}_0(x)$ and $\mathcal{D}_1(x)$ be the distributions of the DCT coefficients in the intracoded and intercoded macroblocks, respectively. Note that in general the DC coefficients from the intracoded macroblocks will not be quantized to zeros because of their relatively large values. Therefore, for any quantization parameter q, the corresponding percentage of zeros ρ can be obtained

as follows:

$$\rho(q) = \frac{1}{M}\int_{-2q}^{+2q} \mathcal{D}_0(x)\,dx + \frac{1}{M}\int_{-2.5q}^{+2.5q} \mathcal{D}_1(x)\,dx, \tag{2.14}$$

where M is the number of coefficients in the current video frame. Note that in the H.263 codec [53], the DCT coefficients are rounded to integers. Therefore, $\mathcal{D}_0(x)$ and $\mathcal{D}_1(x)$ are actually histograms of the DCT coefficients, and Eq. (2.14) becomes

$$\rho = \frac{1}{M}\sum_{|x|<2q} \mathcal{D}_0(x) + \frac{1}{M}\sum_{|x|<2.5q} \mathcal{D}_1(x). \tag{2.15}$$

JPEG and MPEG Coding

Perceptual quantization is employed in the JPEG image coding, and in MPEG-2 and MPEG-4 video coding. Detailed descriptions are given in Eqs. (2.8) and (2.12). After DCT, we first divide each DCT coefficient by its associated perceptual weight, then generate the distribution of these scaled DCT coefficients. After scaling, the perceptual quantization becomes uniform, as we can see from Eqs. (2.8) and (2.12). Therefore, Eqs. (2.13) and (2.15) can be also used to compute the value of ρ from q for JPEG and MPEG coding algorithms, respectively.

2.3.3 Implementation

From the distribution of the transform coefficients, for any given quantization parameter q we can compute the corresponding ρ. In software implementation, we can store the one-to-one mapping between q and ρ in a look-up table. For example, in H.263 and MPEG video coding, the possible values of q are 1, 2, $\cdots$, 31. So, the look-up table has at most 31 entries. Using this look-up table, we can easily map the R-D functions between the q-domain and the ρ-domain.

2.4 Summary

In this chapter, following a brief description of typical transform coding systems and the standard approaches for rate control, we introduced the methodology of ρ-domain R-D analysis. For different quantization schemes in different coding systems, we have shown that the mapping between q and ρ can be computed directly from the distribution of the transform coefficients. Based on this mapping, we map the R-D functions from the q-domain into the ρ-domain. As we observe in the subsequent chapters, the R-D functions have unique properties in the ρ-domain.

Chapter 3

Linear Rate Model

In the previous two chapters, we observed that, in typical transform coding systems, there is a one-to-one mapping between the quantization parameter q and the percentage of zeros ρ. Using this mapping, we can transform the R-D functions from the q-domain into the ρ-domain, or vice versa. In this chapter, based on our extensive experimental results, we first show that $R(\rho)$ is a linear function for different image/video coding systems and different types of source data. Based on Shannon's source coding theorem, we then provide a theoretical justification for the linearity of the rate function. Finally, the physical background of the model parameter, which is the slope of the linear rate function, is discussed.

3.1 Experimental Studies

As mentioned in Section 1.1, the typical transform coding systems included in our studies are EZW, SPIHT, stack-run, JPEG, MPEG-2, H.263, and MPEG-4. They can be categorized into three major groups: wavelet-based image coding (EZW, SPIHT, and

stack-run), DCT based image coding (JPEG), and DCT-based video coding (MPEG-2, H.263, and MPEG-4). The types of source data involved in these coding algorithms are still image; motion compensated picture, such as P- or B- picture; video object plane; base layer; enhancement layer; and so on. In this section, based on our extensive simulation results, we show that, for different coding systems and different types of source data, the actual coding bit rate R is a linear function of ρ.

3.1.1 Wavelet-Based Image Coding

We randomly select 24 sample images which have a wide range of R-D characteristics. The sample images are shown in Fig. 3.1. Each sample image is first decomposed by a 5-level dyadic scheme with the 9-7 wavelet [13, 52]. The decomposed image is uniformly quantized and then coded by the EZW algorithm [16]. We encode each sample image at a series of quantization parameters $\{q_i \ | 1 \leq i \leq L\}$. For each q_i, let the R_i be the corresponding coding bit rate. During the quantization process, we also count the number of zeros produced by the quantizer and then compute the percentage of zeros ρ_i. In this way, for each sample image, we have a series of points $\{(\rho_i, R_i) | 1 \leq i \leq L\}$ on the rate curve $R(\rho)$ which are plotted in Fig. 3.2. It can be seen that $R(\rho)$ is almost a straight line. In addition, this line passes through the point [1.0, 0.0]. This is because, when ρ is 1.0, all of the coefficients are quantized to zeros and the corresponding coding bit rate R should become zero. Therefore, in the ρ-domain, the rate function has the following expression,

$$R(\rho) = \theta \cdot (1 - \rho), \tag{3.1}$$

where θ is a constant. We have also performed the above experiment over many other images coded by other wavelet-based coding algorithms, such as SPIHT and stack-run coding. The above linear rate model has been found to hold.

For comparison, in Fig. 3.3, we plot the rate curve in the q-domain for each sample

image shown in Fig. 3.1. For different images the patterns of $R(q)$ are quite different from each other. In addition, for each sample image, $R(q)$ has a very complex nonlinear behavior. This image-dependent variation and nonlinear behavior make it very hard to develop an accurate and robust source model in the q-domain. This is the reason that the source models in the literature are getting ever more complex although their estimation accuracy remains low. However, in the ρ-domain, the rate curve is a linear function which is extremely simple. This is one of the advantages of the proposed ρ-domain R-D analysis.

In Fig. 3.4, we plot the mapping $\rho(q)$ for each sample image. For different images, $\rho(q)$ are quite different from each other. In addition, $\rho(q)$ has a highly nonlinear behavior. Based on $\rho(q)$, each rate function in the q-domain is mapped into the ρ-domain. We observe that the image-dependent variation and the highly nonlinear behavior of the rate function in the q-domain are largely removed by the mapping from q to ρ. From another point of view, this also implies that the coding bit rate of an image/video encoder is much more closely and directly related to ρ than to q.

Figure 3.1: Sample images selected for our simulations.

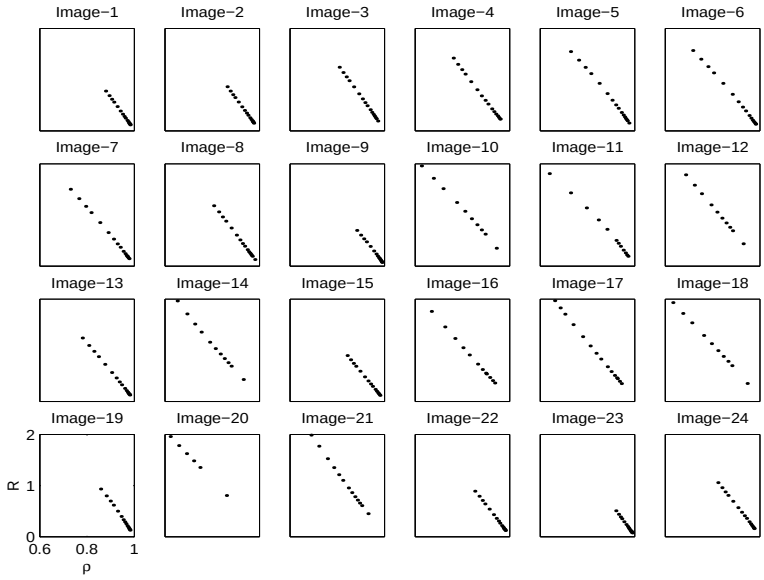

Figure 3.2: The linear relationship between the percentage of zeros ρ and the coding bit rate R in wavelet image coding with EZW. The x-axis represents ρ while the y-axis represents R. All the plots have the same coordinate system.

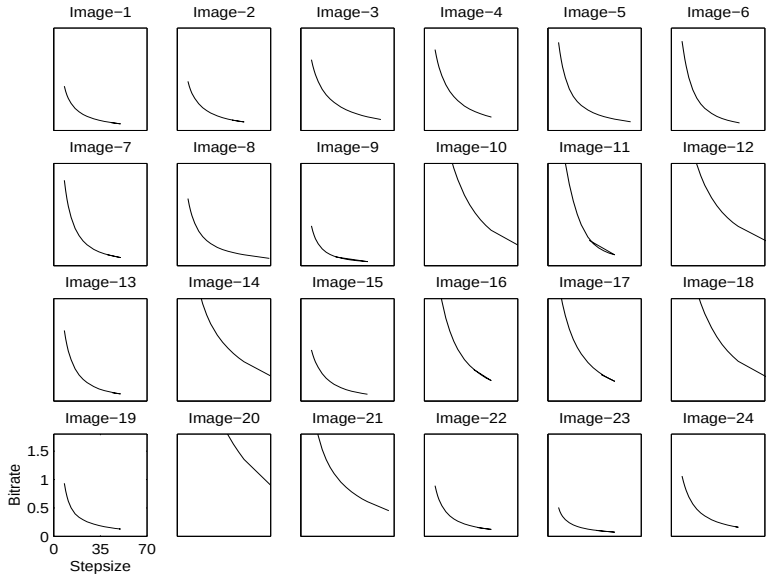

Figure 3.3: The plot of the rate curve $R(q)$ in the q-domain for each sample image coded by EZW. The x-axis represents q while the y-axis represents R. All the plots have the same coordinate system.

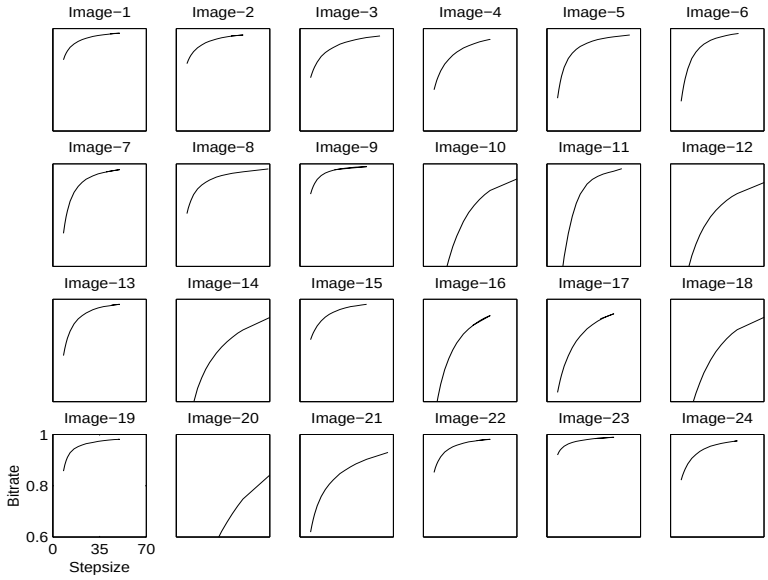

Figure 3.4: The plot of the mapping function $\rho(q)$ for each sample image shown in Fig. 3.1. The x-axis represents q while the y-axis represents ρ. All the plots have the same coordinate system.

3.1.2 DCT-Based Image Coding

From Section 2.1 we know the JPEG coding is quite different from the wavelet-based image coding algorithms described above. However, as we can see in the following experiment, in the ρ-domain they are governed by the same rate model. In Fig. 3.5, we plot the rate function in the ρ-domain $R(\rho)$ for each sample image shown in Fig. 3.1 coded by JPEG. Note that the linear rate model holds for the JPEG coding; when ρ is 1.0, all the coefficients are quantized to zeros, the corresponding coding bit rate is $\frac{1}{16}$ bpp instead of zero. This is because JPEG employs the symbol "EOB" to represent a run of zeros at the end of each block in the zig-zag scan order. This special symbol takes 4 bits when it is Huffman coded. So the actual rate curve is shifted upwards by $\frac{4}{64}$ bpp from the point [1.0, 0.0]. A generalized ρ-domain rate model is give by

$$R(\rho) = \theta \cdot (1 - \rho) + R_h, \tag{3.2}$$

where R_h is a constant which does not depend on the quantization parameter. For JPEG coding, $R_h = \frac{1}{16}$.

3.1.3 DCT-Based Video Coding

As discussed in Section 2.1, DCT-based video coding employs the MC-DCT coding scheme which is different from the coding of still images. The common types of source data involved in video coding are I-frame, motion compensated error picture (such as P- or B- picture), base layer and enhancement layer. In addition, in most cases, especially in active videos, a video frame has both intracoded and intercoded macroblocks. In the object-based MPEG-4 video coding, each frame contains one or more video objects. Although, complex coding procedures are employed in typical video coding systems, and various types of source data are involved, their rate functions in the ρ-domain have unique and simple behaviors. In other words, as we can see from the experiments described in the following, the linear rate model also holds in these typical video coding systems.

With the H.263 video codec [53], we encode the test video sequence at a series of quantization stepsizes $\{q_i | 1 \leq i \leq L\}$. Let R_i be the coding bit rate excluding the motion vectors (MVs) and the header information bits. That is to say, we consider only the coding rate related to the DCT coefficients. It should be noted that the number of bits for MVs and header information is already determined before quantization. In Fig. 3.6, we plot $\{(\rho_i, R_i)\}$ for several frames from the "Foreman" video sequence. It can be seen that, in the ρ-domain, R is approximately a linear function of ρ.

To examine this linear relationship more systematically, we study the correlation

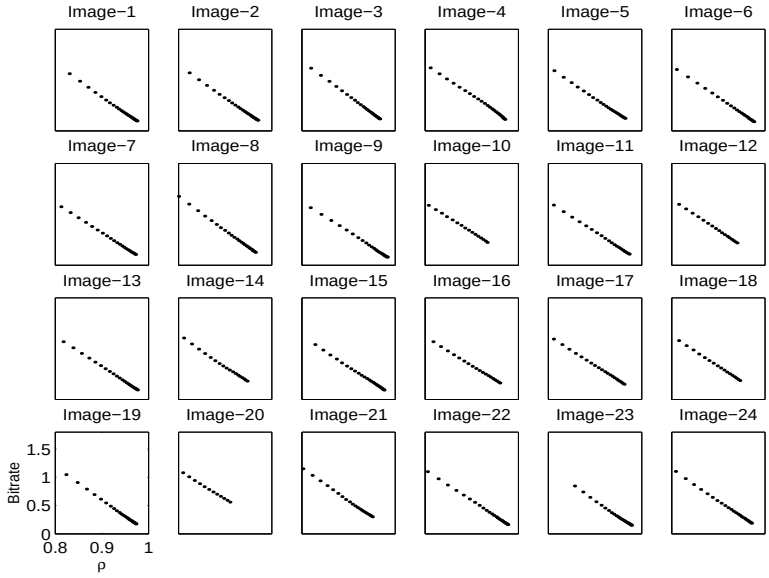

Figure 3.5: The linear relationship between the percentage of zeros ρ and the coding bit rate R in JPEG image coding. The x-axis represents ρ while the y-axis represents R. All the plots have the same coordinate system.

coefficient between ρ_i and R_i, denoted by $\mathcal{C}(\rho_i, R_i)$. By definition we have

$$\mathcal{C}(\rho_i, R_i) = \frac{\sum_{i=1}^{L} \rho_i R_i}{\sqrt{\sum_{i=1}^{L} \rho_i^2} \cdot \sqrt{\sum_{i=1}^{L} R_i^2}}. \tag{3.3}$$

$\mathcal{C}(\rho_i, R_i)$ becomes ± 1 if and only if there is an exact linear relationship between ρ_i and R_i. In Fig. 3.7, we plot the values of $-\mathcal{C}(\rho_i, R_i)$ for each frame of the "Carphone", "Salesman", "Coastguard" and "Akiyo" video sequences coded by MPEG-2. In Fig. 3.8, we plot the values of $-\mathcal{C}(\rho_i, R_i)$ for each frame of "News", "Akiyo" and each VOP in the two objects of "News" coded by MPEG-4. It should be noted that in MPEG-4 coding R_i does not include the bits for the shape information. From Figs. 3.7 and 3.8, we can seen that $-\mathcal{C}(\rho, R)$ is always larger than 0.99. For most of the frames or VOPs, it is even larger than 0.995 which is extremely close to 1. This implies that the linear rate model given in Eq. (3.1) also holds for the DCT-based video coding systems.

These extensive simulation results with various coding systems and a wide range of

image/video data have shown that the rate function is linear in the ρ-domain.

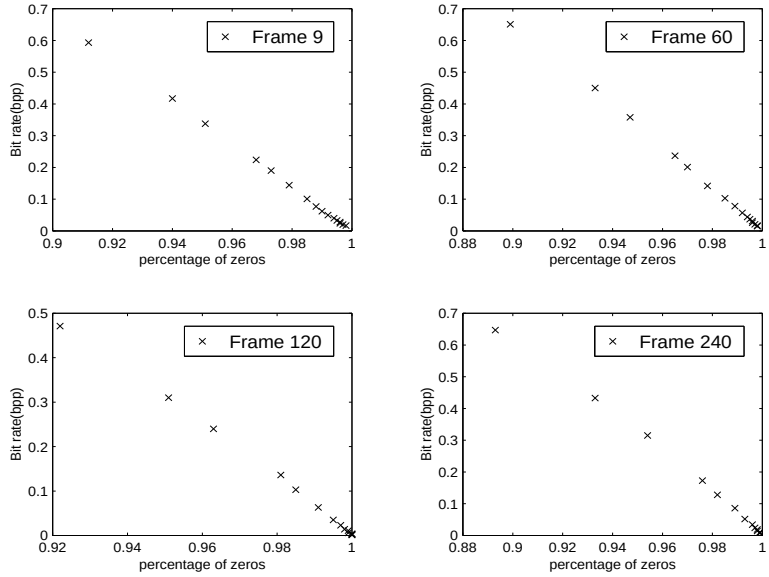

Figure 3.6: The linear relationship between the percentage of zeros ρ and the coding bit rate R in H.263 video coding. The test frames are from the Foreman QCIF video.

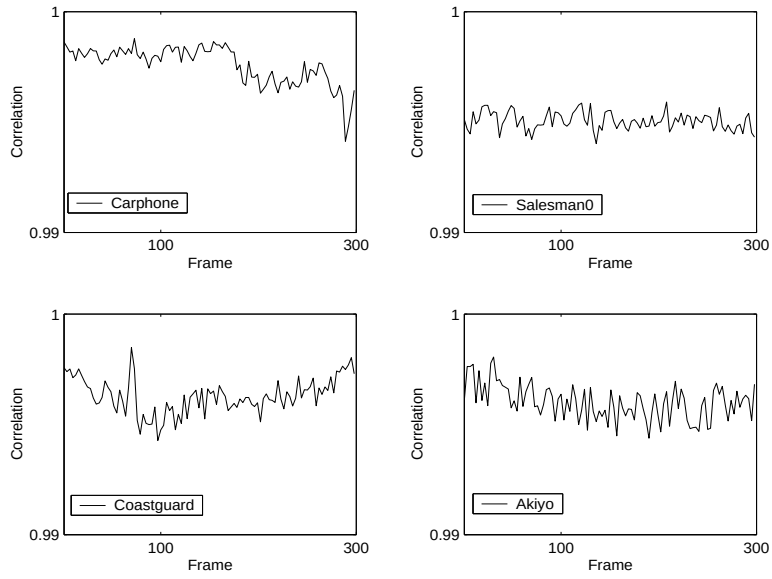

Figure 3.7: The correlation coefficient (inverse) of each frame between the coding bit rate R and ρ in MPEG-2 video coding.

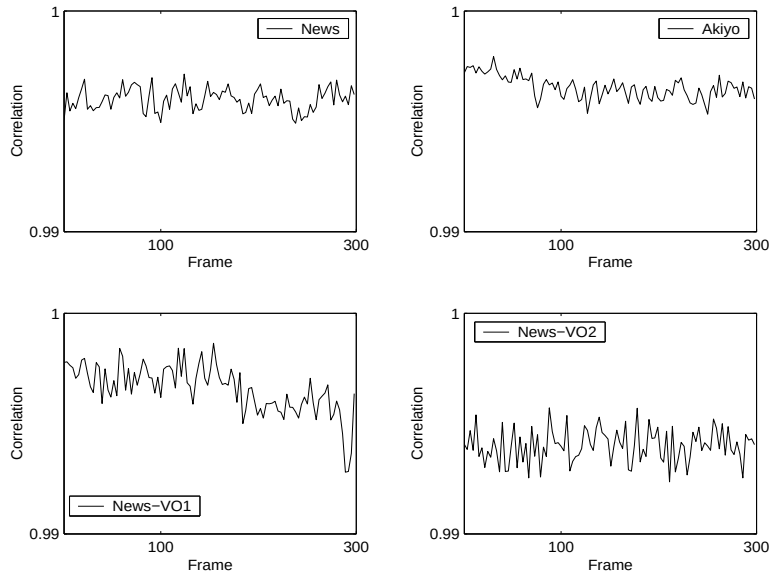

Figure 3.8: The correlation coefficient (inverse) of each frame between the coding bit rate R and ρ in MPEG-4 object-based video coding.

3.2 Theoretical Justification

In this section, based on Shannon's source coding theorem [20, 21], we provide a theoretical justification for the ρ-domain linear rate model in Eq. (3.1). The literature [54, 56, 57] indicates that transform coefficients have a generalized Gaussian distribution which is given by

$$p_{gg}(x) = \left[\frac{\nu\eta(\nu,\sigma)}{2\Gamma(1/\nu)}\right] \cdot e^{-[\eta(\nu,\sigma)|x|]^{\nu}}, \tag{3.4}$$

where

$$\eta(\nu,\sigma) = \sigma^{-1}\left[\frac{\Gamma(3/\nu)}{\Gamma(1/\nu)}\right]^{1/2}, \quad 1 \le \nu \le 2. \tag{3.5}$$

Here σ is the standard deviation of the transform coefficients, and ν is a model parameter which controls the shape of the distribution. For example, when $\nu = 2.0$, $p_{gg}(x)$ is a Gaussian distribution given by

$$p_g(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}(\frac{x}{\sigma})^2}. \tag{3.6}$$

When $\nu = 1.0$, $p_{gg}(x)$ becomes a Laplacian distribution given by

$$p_l(x) = \frac{\lambda}{2} e^{-\lambda|x|}. \tag{3.7}$$

In DCT-based image/video coding, Lam and Goodman [55] have mathematically derived that the DCT coefficients have a Laplacian distribution. In the following, before studying the generalized Gaussian source, we first consider its two special cases: the Laplacian and the Gaussian sources.

3.2.1 Laplacian Source

For the Laplacian source, let us define the distortion measure as $D(x, \tilde{x}) = |x - \tilde{x}|$ where x is the input symbol and $\tilde{x}$ is the output symbol of the quantizer. According to Shannon's source coding theorem [21, 58, 59], if a distortion D is allowable, the minimum number of bits needed to represent a symbol from a Laplacian source is given by

$$R(D) = \log_2 \left(\frac{1}{\lambda D} \right). \tag{3.8}$$

First consider a uniform quantizer. For a given quantization stepsize q, the corresponding distortion is

$$D(q) = 2 \int_0^{0.5q} p_l(x) x \, dx + 2 \sum_{i=1}^{\infty} \int_{(i-0.5)q}^{(i+0.5)q} p(x) |x - iq| \, dx. \tag{3.9}$$

With Eq. (3.7), we have

$$D(q) = \frac{1}{\lambda} [\, 1 + \frac{e^{-\lambda q}}{1 - e^{-\lambda q}} (2 - e^{-0.5\lambda q} - e^{0.5\lambda q}) - e^{-\lambda q} \,]. \tag{3.10}$$

Note that the percentage of zeros is given by

$$\begin{aligned} \rho &= \int_{-0.5q}^{0.5q} \frac{\lambda}{2} e^{-\lambda|x|} \, dx \\ &= 1 - e^{-0.5\lambda q}. \end{aligned} \tag{3.11}$$

After changing the independent variable from q to ρ, Eq. (3.10) becomes

$$D(\rho) = \frac{1}{\lambda} \cdot \frac{1-(1-\rho)}{1+(1-\rho)}. \tag{3.12}$$

With Eqs. (3.8) and (3.12), we have

$$R(\rho) = \log_2 \left[\frac{1+(1-\rho)}{1-(1-\rho)} \right]. \tag{3.13}$$

In our extensive simulations we observe that, in transform coding of images and videos at low bit rates, such as 1.5 bpp or less, the corresponding percentage of zeros ρ is mostly larger than 0.8, as we can see from Figs. 3.2 and 3.5. Hence, $1-\rho$ is mostly smaller than 0.2 which is close to zero. For $1-\rho$ close to zero, a Taylor expansion of Eq. (3.13) yields

$$R(\rho) = 2 \cdot \log_2 e \cdot (1-\rho) + O([1-\rho]^3), \tag{3.14}$$

which is approximately a linear function.

The above mathematical formulation is for the uniform quantizer with a dead zone threshold $0.5q$. In image and video coding, a uniform threshold quantizer with a larger dead zone is often used to produce more quantized zeros in order to reduce the coding bit rate. Suppose the dead zone threshold is $\Delta = (0.5+b)q$ where b is some positive constant. The corresponding quantization distortion is given by,

$$D(q) = 2\int_0^{\Delta} p_l(x)x\,dx + 2\sum_{i=0}^{\infty}\int_{iq+\Delta}^{(i+1)q+\Delta} p_l(x)|x-(i+\frac{1}{2})q-\Delta|\,dx. \tag{3.15}$$

In this case, the percentage of zeros is given by

$$\begin{aligned} \rho &= \int_{-\Delta}^{\Delta} \frac{\lambda}{2} e^{-\lambda|x|}\,dx \\ &= 1-e^{-\lambda\Delta}. \end{aligned} \tag{3.16}$$

With Eqs. (3.15), (3.17) and (3.8), the expression of $R(\rho)$ becomes

$$R(\rho) = \log_2 \frac{1+x^a}{1+x^a-2x+(1-a)(1+x^a)x\ln x}, \tag{3.17}$$

where

$$a = \frac{0.5}{0.5 + b}, \quad x = 1 - \rho. \tag{3.18}$$

We plot $R(\rho)$ for $b = 0.0$, 0.5 and 0.75 in Fig. 3.9. It can be seen that the plots are all very close to being straight lines. This theoretically justifies the linear rate model given by Eq. (3.1) for the Laplacian source.

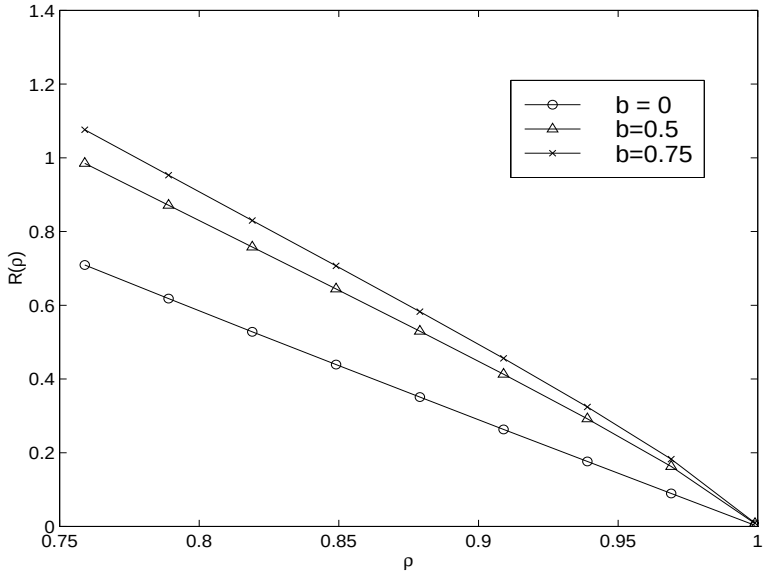

Figure 3.9: Plots of the function $R(\rho)$ given by Eq. (3.17) for different dead zone threshold values.

3.2.2 Gaussian Source

Next we consider the Gaussian source which is another special case of the generalized Gaussian source. For the Gaussian distribution we need to employ the square error distortion

$$D(x, \tilde{x}) = (x - \tilde{x})^2. \tag{3.19}$$

According to Shannon's source coding theorem [58, 59], if a square distortion D given by Eq. (3.19) is allowed, the minimum number of bits needed to represent a symbol

from a Gaussian source is given by

$$R(D) = \begin{cases} \frac{1}{2}\log_2 \frac{\sigma^2}{D} & 0 \leq D \leq \sigma^2, \\ 0 & D > \sigma^2 \end{cases} \tag{3.20}$$

For a uniform threshold quantizer with a step size q and dead zone Δ, the corresponding distortion is given by

$$D(q) = 2\int_0^{\Delta} p_g(x)x^2\,dx + 2\sum_{i=0}^{\infty}\int_{iq+\Delta}^{(i+1)q+\Delta} p_g(x)[x - (i+\frac{1}{2})q - \Delta]^2\,dx. \tag{3.21}$$

It might be very difficult to derive an explicit expression of $R(\rho)$ in the same way as for the Laplacian distribution. Instead, we evaluate $R(\rho)$ numerically and plot it for different dead zone thresholds in Fig. 3.10. It can be seen that the rate function in the ρ-domain for a Gaussian source is also also very close to being a linear function.

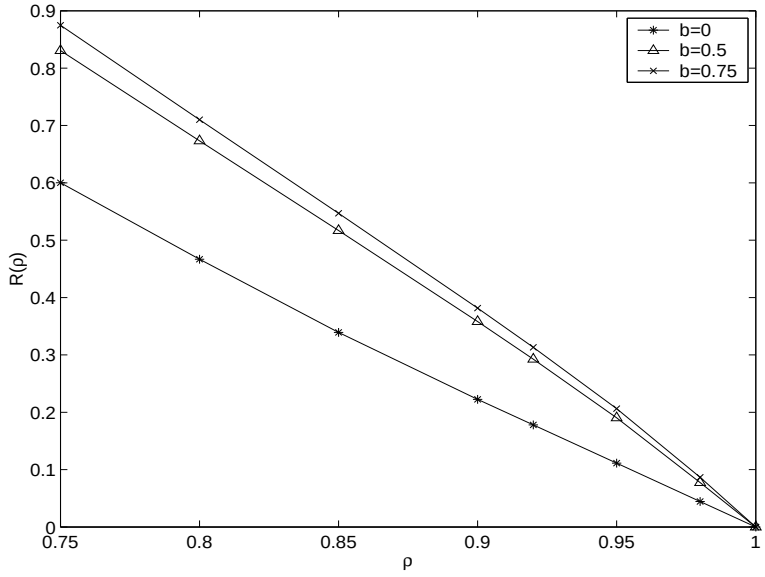

Figure 3.10: Plots of the function $R(\rho)$ for a Gaussian source at different dead zone threshold values.

3.2.3 Generalized Gaussian Distribution

Laplacian and Gaussian sources are two special cases of the generalized Gaussian source. For these two types of sources coupled with appropriate distortion measures, based on the source coding theorem, we have the explicit expressions for their R-D functions as given in Eqs. (3.8) and (3.20). However, for a generalized Gaussian distribution, due to the complex expression of itself, it is difficult to do so. Instead, we can obtain the lower and upper bounds of its R-D function [59]. To be more specific, for a generalized Gaussian source X with mean zero and variance σ^2, we have

$$h(X) - \frac{1}{2}\log_2(2\pi e)D \leq R(D) \leq \frac{1}{2}\log_2\frac{\sigma^2}{D}. \tag{3.22}$$

Here $h(X)$ is the differential entropy of X which is given by

$$h(X) = -\int_{-\infty}^{\infty} p_{gg}(x)\log_2 p_{gg}(x)\,dx. \tag{3.23}$$

The distortion D in Eq. (3.22) refers to the square error distortion. Following the same procedure as for the Gaussian source, we numerically evaluate the lower and upper bounds of the rate function in the ρ-domain and plot them in Fig. 3.11. Here the variance of the source is set to be $\sigma^2 = 100$ and the distribution control parameter is set to be $\nu = 1.5$. It can be seen that the lower and the upper bounds of the rate function are very close to each other. In addition, both are approximately linear functions. Since the actual rate function should lie between them, it should therefore also be approximately a linear function. This justifies the linear rate model given in Eq. (3.1) for a generalized Gaussian source.

3.3 Physical Background of the Model Parameter

The only parameter of the proposed source model in Eq. (3.1) is the slope θ. Fig. 3.12 shows the value of θ for each sample image listed in Fig. 3.1. As we expect, different

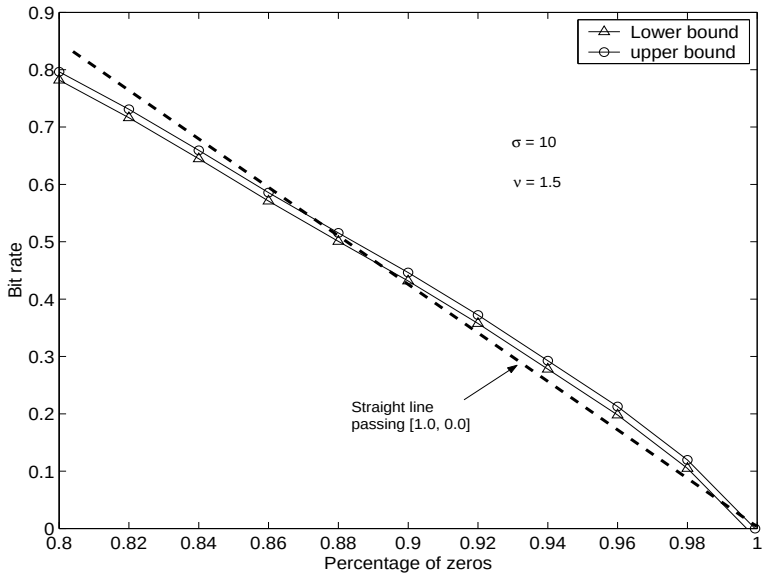

Figure 3.11: Plots of the lower and upper bounds of the ρ-domain rate function $R(\rho)$ for a generalized Gaussian source with $\sigma = 10$ and $\nu = 1.5$.

images have different values of θ; the variation of θ is relatively large. From Eq. (3.1) we know θ must be related to some characteristics of the input source data which have a determining effect on the coding bit rate. In this section, we try to find out what are the characteristics and their physical background. To this end, we sort all of these sample images according to the value of θ. In Fig. 3.13, the sorted images are listed in a raster scan order with θ increasing from the smallest to the largest. It can be seen that the images in the first half have more high-frequency texture than those in the second half. The images in the second half are smoother and more structured. This suggests us that the value of θ is closely related to the amount of texture presented in the corresponding image.

In the frequency domain, an image with more texture normally has a relatively larger amount of middle- or high-frequency components [60]. In other words, the energy is more distributed to the middle- or high- frequency subbands. For smoother and more structured images with less texture, the energy is more concentrated in the lower-

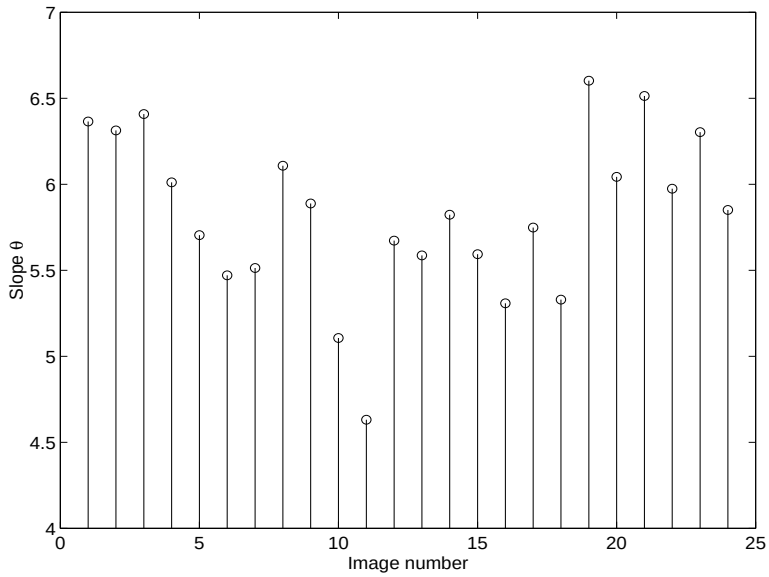

Figure 3.12: Plot of the slope θ of the linear rate function for each sample image shown in Fig. 3.1.

frequency subbands. Consider the wavelet transform as an example. After a 5-level dyadic subband decomposition, there is a total of 16 subbands, denoted by $\{S_i | \ 1 \leq i \leq 16\}$. Let σ_i^2 be the variance of the wavelet coefficients in S_i. Let σ_a^2 and σ_g^2 be the arithmetic mean and geometric mean of $\{\sigma_i^2 | 1 \leq i \leq 16\}$, respectively. We define the energy compaction measure Ω as

$$\Omega = \frac{\sigma_a^2}{\sigma_g^2} = \sum_{i=1}^{16} \sigma_i^2 \Big/ \left(\prod_{i=1}^{16} \sigma_i^2 \right)^{\frac{1}{16}}. \tag{3.24}$$

Obviously, larger Ω corresponds to more compacted energy and fewer texture components. Actually, Ω is often used as a feature variable for texture analysis [60]. In Fig. 3.14, we plot the pair of [Ω, θ] for each sample image listed in Fig. 3.1. We can see that there is a strong linear correlation between Ω and θ. The correlation coefficient between them is 0.845, which is very high. This strong correlation explains the physical meaning of θ, which is the only parameter of the proposed source model.

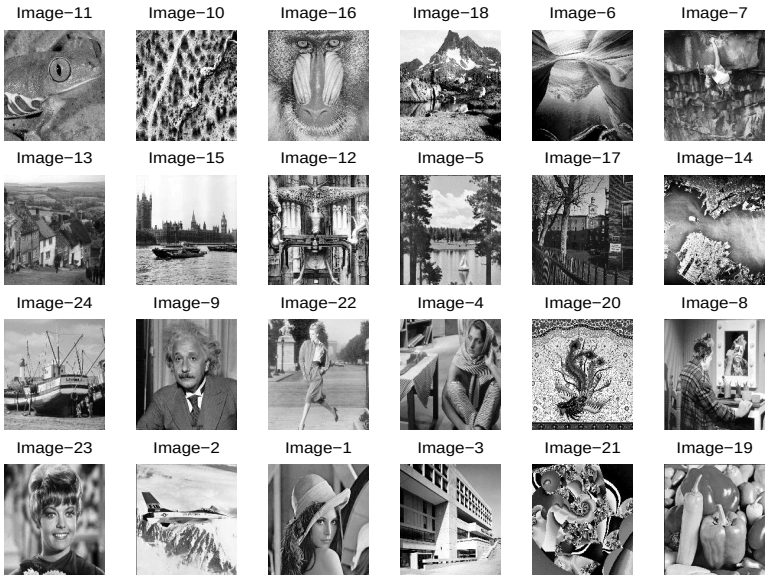

Figure 3.13: The samples images sorted by θ. The images are listed in the raster scan order.

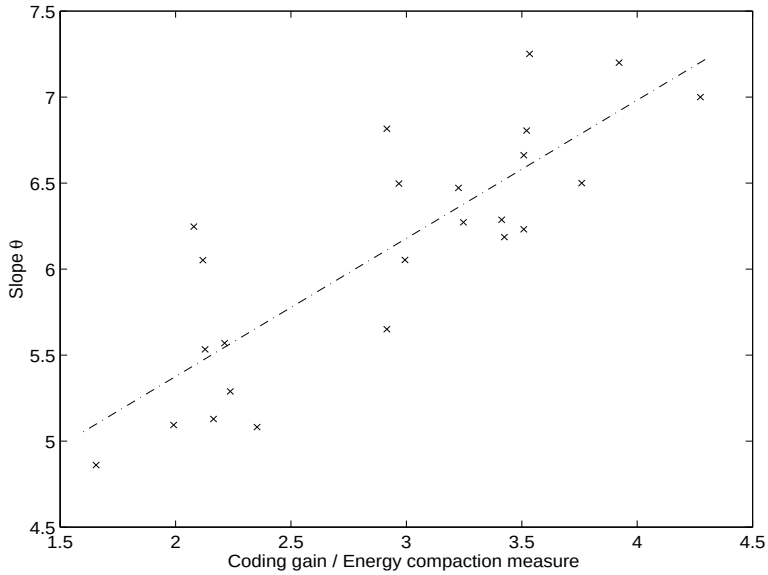

Figure 3.14: The linear correlation between the energy compaction measure Ω and the slope θ.

3.4 Summary

In this chapter, we have shown both experimentally and theoretically that in the ρ-domain the rate function is linear. This linear rate model is a unified model for any typical transform coding systems and all types of source data. Compared to other rate models proposed in the literature, it is much simpler. However, it is very accurate because it is derived directly from the actual coding results. We have also discussed the physical meaning of its only model parameter θ. In the next chapter, based on this linear rate model, a unified rate control algorithm for MPEG-2, H.263, and MPEG-4 video coding will be developed.

Chapter 4

ρ-Domain Rate Control for Video Coding

In video coding and transmission, the coding behavior of the video encoder has to be controlled to satisfy the network transmission condition, to meet the video presentation quality requirement, or both. To this end, we need to have sufficient knowledge about the R-D behavior of the video encoder which is characterized by its R-D functions. In Chapter 3, we developed a linear rate model in the ρ-domain. Based on this simple but very accurate rate model, we will develop a unified rate control algorithm for MPEG-2, H.263, and MPEG-4 video coding. In this chapter, following a brief background review of the rate control framework for video coding and transmission, we present the ρ-domain rate control algorithm. Extensive simulation results are provided to demonstrate the superior performance of the proposed rate control algorithm. Its computational complexity and implementation cost are also discussed.

4.1 Rate Control for Video Coding and Transmission

Recent advances in high-speed network, high-capacity storage, and computing technology have stimulated the wide use of digital video in multimedia communication. In the digital video technology, the raw video data is first compressed, then stored on the local storage devices or transmitted through the network channel to the end user. Compression and transmission of the video data over the network is subject to the channel bandwidth as well as to certain time and delay constraints. This is because each coded video frame has to be transmitted to the decoder before it is scheduled to be decoded and displayed.

4.1.1 Video Applications

According to the way that the video data is compressed, transmitted, decoded, and displayed, the video applications can be categorized into the following three types.

Type A. Storage video for local usage

The raw video data is compressed and stored on a local storage device, such as hard drive or compact disk (CD). During the process of playback, the coded video data is transmitted through the internal data bus whose data transmission speed is sufficiently large to meet the decoding and displaying requirement. This type of video application includes the storage video on CD and on digital versatile disk (DVD). The coded video data stored in a remote computer can also be downloaded as a whole through the network onto the local temporary folder before being decoded and displayed. This kind of video application should also be regarded as stored video for local usage because no network transmission is involved during the decoding process.

Type B. Real-time transmission of stored video

The video data is compressed and stored to be accessed and used by a remote user through the network. To display the video, the user does not need to download the whole video which often takes a unacceptable period of time. In contrast, after a certain amount of video data has been buffered in the user's local computer, the decoding and displaying process immediately starts. One important requirement in this kind of real-time video transmission is that each video frame must arrive at the user's end before it is scheduled to be displayed. Otherwise, the video player has to pause the decoding process and wait for this video frame, which is often called "rebuffering". Frequent interrupts of the playing process due to data rebuffering significantly degrade the video presentation quality. This type of video application includes video-on-demand [81] and broadcast of existing programs, such as news, movies, and sports coverage.

Type C. Real-time transmission of live video

The video data is captured, compressed, and transmitted through the communication channel to the end user in real-time. This type of video application includes live TV or video broadcasts. Obviously, they require even more efficient video coding and transmission schemes than Types A and B. A even more challenging case is the interactive application, such as videophone, video conferencing, and interactive classroom which require a relatively small end-to-end delay.

4.1.2 Functionality of Rate Control

In general, the coded video data has a variable bit rate due to the scene activities. In video transmission as shown in Fig. 4.1, a buffer (called encoder buffer) is introduced at the encoder to temporarily store the coded video data which cannot be transmitted because of the mismatch between the encoder bit rate and the channel bandwidth.

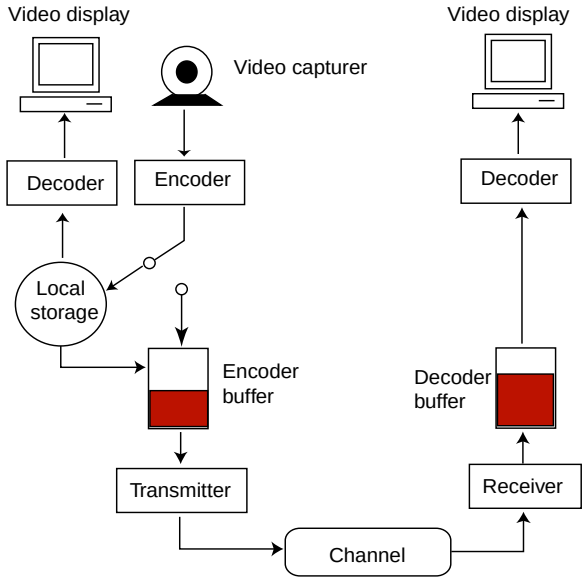

Figure 4.1: A generic video coding and transmission system.

Another buffer is used at the decoder side to temporarily store the video data which cannot be decoded immediately because of the mismatch between the channel bandwidth and the decoding speed. Practically, buffers are limited in size. This requires an efficient rate control algorithm to regulate the coding behavior of the encoder to avoid buffer overflow and underflow.

In general, rate control has two basic functions. The first is to control the output bit rate of the video encoder according to the network condition. Its objective is to maintain a successful transmission of the coded video data by adjusting the encoder setting. Accurate and robust rate control is requisite for Types B and C video applications, especially for interactive video technologies, which require a very small end-to-end delay and, consequently, small encoder and decoder buffers. Let the target coding bit rate for the encoder be R_T, which is determined by buffer status and the channel bandwidth. Let R be the actual coding bit rate. The control error is then given by $R - R_T$. This amount of video data is accommodated by the encoder buffer.

Small buffer size requires the rate control algorithm to accurately estimate the coding behavior of the video encoder, adjust its quantization settings, and match R to R_T as closely as possible.

The other function of rate control is to optimize the video presentation quality given the constraint of storage capacity or channel bandwidth. This applies to all the three types of video applications. Based on estimated R-D functions of the encoder, the rate control algorithm can be employed to maximize the video quality by optimum allocation of bits and accurate selection of the quantization parameters for the encoder.

4.2 ρ-Domain Rate Control Algorithms

A generic video encoder coupled with a rate controller is illustrated in Fig. 4.2. Based on the encoder buffer level B, the channel bandwidth C and the frame rate F, the target bit rate R_T for the current video frame is determined as follows,

$$R_T = \frac{C}{F} - B + \tau \cdot B_T, \tag{4.1}$$

where B_T is the buffer size and τ is the target buffer level. In buffer regulation, it is necessary to keep a certain number of bits τB_T inside the buffer to avoid possible underflow [23]. By default, τ is set to be 0.2. Note that R_T is the number of bits for coding all the information of the current frame which includes the motion vectors, header information, and the DCT coefficients. However, the motion vectors and header information bits, denoted by R_{MV} and R_H, are independent of quantization. They can be determined before our rate control process begins. Let

$$R_C = R_T - R_{MV} - R_H. \tag{4.2}$$

Obviously, R_C is the target number of bits for coding the DCT coefficients.

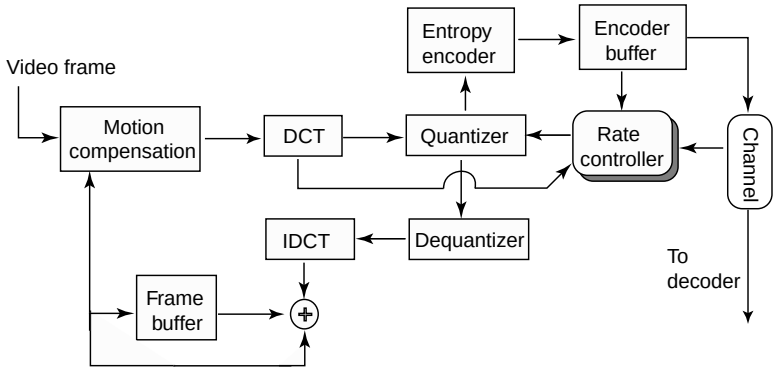

Figure 4.2: A generic rate control framework for video coding and transmission.

Based on the estimated rate function of the video encoder, the rate control algorithm can determine the quantization parameter to achieve the target bit rate R_C. In Chapter 3, in the ρ-domain, we developed a linear rate model given by Eq. (3.1). The simplicity and accuracy of this model allow robust and precise rate control. The only parameter of the rate model is θ. In this section, based on the adaptive estimation of θ and adaptive selection of the quantization parameter, two different ρ-domain rate control algorithms are presented. Table 4.1 summarizes the notations to be used in the following algorithm description.

4.2.1 Algorithm 1: Adaptive Estimation and Control

In this algorithm, a macroblock-level adaptive scheme is developed to estimate the value of θ and, in turn, to select the quantization parameter for each macroblock. According to Eq. (3.1), the actual value of θ is given by

$$\theta = \frac{R}{1-\rho}, \tag{4.3}$$

where R and ρ are the actual coding bit rate and percentage of zeros of the current frame. But, before encoding the frame, we have no information about R and ρ. How-

Table 4.1: Description of notations.

M	Picture size. For a QCIF picture, $M = 176 \cdot 144 \cdot 1.5$ $= 38016$.
N	The number of macroblocks in one video frame. For a QCIF frame, $N = 99$.
N_m	The number of macroblocks which have already been coded in the current frame.
R_m	The number of bits used by the already coded macroblocks in the current frame.
ρ_m	The number of zeros produced by the already coded macroblocks in the current frame.
$\mathcal{D}_0^{N_m}(x)$	The distribution of the DCT coefficients in the intracoded macroblocks which are not yet coded in the current frame.
$\mathcal{D}_1^{N_m}(x)$	The distribution of the DCT coefficients in the intercoded macroblocks which are not yet coded in the current frame.
QP	The quantization parameter of the current macroblock.
R_{MV}	The total number of bits for coding the motion vectors in the current frame.
R_H	The total number of bits for coding the header information in the current frame.
R_C	The total number of bits for coding the DCT coefficients in the current frame.
R^K	The actual bit rate for coding the DCT coefficients in the K-th video frame.
ρ^K	The percentage of zeros produced by the coded K-th frame
$\bar{Q}^K$	The average quantization parameter of the K-th frame

ever, during the encoding process, we do know the values of R_m and ρ_m for the N_m coded macroblocks in the current frame. Let

$$\theta_m = \frac{R_m}{N_m \cdot 384 - \rho_m}. \tag{4.4}$$

Here, "384" refers to the number of DCT coefficients in one macroblock. Obviously, we have

$$\theta_m \to \theta, \quad \text{when} \quad N_m \to N. \tag{4.5}$$

Therefore, we use θ_m as an estimation of θ and apply it to the current macroblock to determine its quantization parameter as illustrated in Fig. 4.3.

With the adaptive estimation of θ, the rate control algorithm turns out to be simple and straightforward. The quantization parameter for each macroblock is determined in the following steps:

Step 1. *Initialization.* Before encoding the first macroblock of the current video frame, set $N_m = R_m = \rho_m = 0$. Generate the distributions $\mathcal{D}_0^{N_m}(x)$ and $\mathcal{D}_1^{N_m}(x)$ which are defined in Table 4.1. Compute R_C based on Eq. (4.2). Set $\theta = 7$ which is its average value in our extensive simulations.

Step 2. *Determine the quantization parameter q.* According Eq. (3.1), the number of zeros to be produced by the quantization of the remaining macroblocks should be

$$\rho = 384 \cdot (N - N_m) - \frac{R_C - R_m}{\theta}. \tag{4.6}$$

Based on the ρ-q one-to-one mapping which is computed from $\mathcal{D}_0^{N_m}(x)$ and $\mathcal{D}_1^{N_m}(x)$, the corresponding quantization parameter q is then determined. The current macroblock is quantized with $QP = q$ and encoded by the video coding algorithm.

Step 3. *Update.* Let ρ_0 and R_0 be the number of zeros and number of bits produced by the current macroblock, respectively. Set $\rho_m = \rho_m + \rho_0$, $R_m = R_m + R_0$, and

$N_m = N_m + 1$. If $N_m \geq 0.2 \cdot R_C$, update the value of θ by θ_m which is given by Eq. (4.4). At the same time, subtract the frequencies of the DCT coefficients in the current macroblock from $\mathcal{D}_0^{N_m}(\mathrm{x})$ if the current macroblock is an intracoded macroblock, or from $\mathcal{D}_1^{N_m}(\mathrm{x})$ if it is an intercoded macroblock.

Step 4. *Loop.* Repeat steps 2 and 3 for the next macroblock until all of the macroblocks in the current frame are encoded.

We can see that in this algorithm no information from the previous frames is utilized during the rate control process of the current frame. In other words, the rate control for each frame is totally independent from others. Therefore, it does not suffer from any performance degradation at scene changes. When determining the quantization parameter for each macroblock, the algorithm always divides the whole picture into two parts: the coded macroblocks and the uncoded macroblocks, and then considers the information from both parts. As we will see from the experimental results presented in Section 4.3, this type of rate control mechanism is very robust.

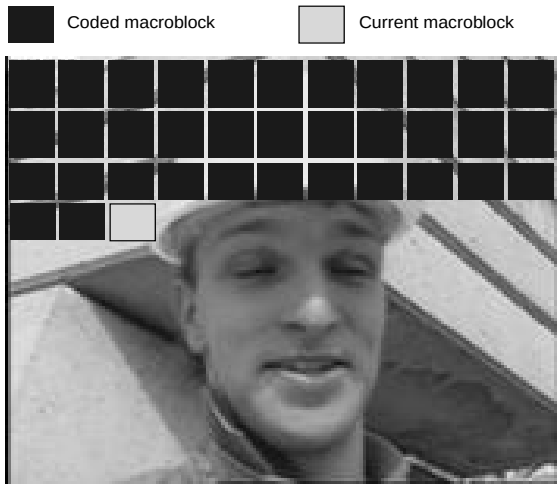

Figure 4.3: Adaptive estimation of θ and macroblock-level adaptive quantization.

4.2.2 Algorithm 2: Rate Control Based On Quantization Parameter Relaxation

Although the computational complexity of Algorithm 1 is already low when compared to other rate control algorithms reported in the literature [32, 35, 23], its complexity can be largely reduced further by the following quantization parameter relaxation method. The complexity of Algorithms 1 and 2 will be analyzed in detail in Section 4.2.3. Our second rate control algorithm is based on the following observation. After the previous $(K-1)$-th frame is coded we know its rate slope θ^{K-1}. If we use θ^{K-1} as an estimation of the rate slope of the current K-th frame, an estimation error is introduced. But we can surpress this estimation error by the quantization parameter relaxation method which is explained in the following.

As shown in Fig. 4.4, after the (K-1)-th frame is encoded, we already know R^{K-1} and ρ^{K-1}, which are defined in Table 4.1. Its corresponding rate slope is then given by,

$$\theta^{K-1} = \frac{R^{K-1}}{1-\rho^{K-1}}. \tag{4.7}$$

If the K-th frame is very similiar to the (K-1)-th frame, its rate slope θ^K should also be very close to θ^{K-1}. If we use θ^{K-1} as the estimation of θ^K, according to Eq. (3.1) the percentage of zeros to be produced by the K-th frame is given by,

$$\rho^K = 1 - \frac{R_C}{M\theta^K} = 1 - \frac{R_C}{M\theta^{K-1}}. \tag{4.8}$$

As discussed in Section 2.3.2, based on the distribution information of the DCT coefficients in the current frame, the one-to-one mapping between the quantization parameter and the percentage of zeros can be computed. Based on this mapping, the quantization parameter corresponding to ρ^K can be obtained. We denote it as $\bar{Q}^K$.

However, in practical video coding, the scene activities, especially at scene changes, cause these two neighboring frames to often have different R-D characteristics. In this

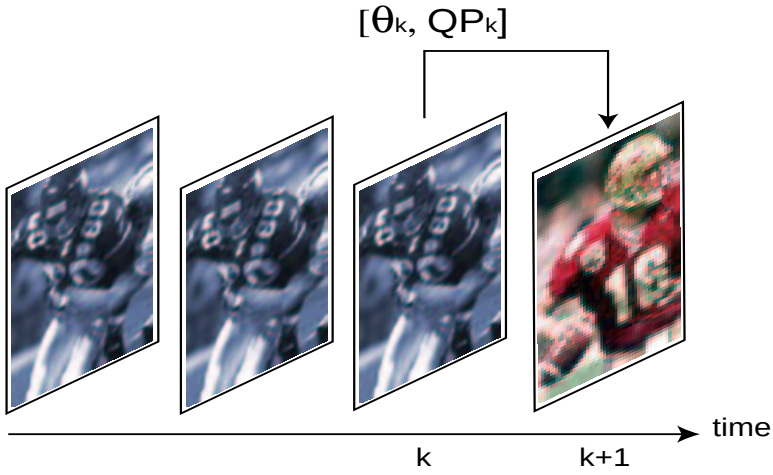

Figure 4.4: Frame-level estimation of θ with quantization parameter relaxation.

way, the actual rate slope θ^K is different from θ^{K-1}, which implies that Eq. (4.8) is not true. Therefore, the frame quantization parameter $\bar{Q}^K$ obtained above cannot achieve the target bit budget R_C. However, we observe that the desired frame quantization parameter which can achieve R_c should fall in the neighborhood of $\bar{Q}^K$. Let $\mathcal{Q} = \{\bar{Q}^K - 2, \bar{Q}^K - 1, \bar{Q}^K, \bar{Q}^K + 1, \bar{Q}^K + 2\}$ be the five candidate quantization parameters for each macroblock in the current K-th frame. Suppose we are now encoding the $(N_m + 1)$-th macroblock. If $R_m \leq 0.2 \cdot R_C$, the quantization parameter of the current macroblock QP is set to be $\bar{Q}^K$. Otherwise, we choose

$$QP = \begin{cases} \bar{Q}^K - 2 & : \quad if\ \theta_m \in [1.3\theta^K,\ \infty); \\ \bar{Q}^K - 1 & : \quad if\ \theta_m \in [1.1\theta^K,\ 1.3\theta^K); \\ \bar{Q}^K & : \quad if\ \theta_m \in [0.9\theta^K,\ 1.1\theta^K); \\ \bar{Q}^K + 1 & : \quad if\ \theta_m \in [0.7\theta^K,\ 0.9\theta^K); \\ \bar{Q}^K + 2 & : \quad if\ \theta_m \in [0,\ 0.7\theta^K). \end{cases} \tag{4.9}$$

It can be seen that in this rate control algorithm the difference between the rate slopes of two neighboring frames is compensated by the relaxation and adaptive selection of the quantization parameters. When θ_m, which is the adaptive estimation of the true θ,

is smaller than θ^K, we know $\bar{Q}^K$ is too small to achieve the target bit rate R_C. Hence, we need to select a larger quantization parameter in its neighborhood. Otherwise, we just select a smaller one. Based on the above frame-level estimation of the rate slope and the quantization parameter relaxation scheme given by Eq. (4.9), we present Algorithm 2 as follows.

Step 1. *Initialization.* Before encoding the video sequence, set $\theta^0 = 7$ and $K = 1$. Suppose that the current frame is the K-th frame. Before encoding the first macroblock of the current frame, set $N_m = R_m = \rho_m = 0$. Compute R_C with Eq. (4.2).

Step 2. *Determine the frame quantization parameter QP^K.* Generate $\mathcal{D}_0(x)$ and $\mathcal{D}_1(x)$ which are the distributions of the DCT coefficients in the intracoded and intercoded macroblocks in the current frame, respectively. Compute the one-to-one mapping between the quantization parameter and the percentage of zeros as discussed in Section 2.3.2. Compute ρ^K with Eq. (4.8) and determine its corresponding quantization parameter QP^K based on the one-to-one mapping.

Step 3. *Macroblock-level adaptive quantization.* Compute θ_m with Eq. (4.4). Determine the quantization parameter QP for the current macroblock according to Eq. (4.9).

Step 4. *Update.* Let ρ_0 and R_0 be the number of zeros and the number of bits produced by the current macroblock, respectively. Set $\rho_m = \rho_m + \rho_0$, $R_m = R_m + R_0$, and $N_m = N_m + 1$.

Step 5. *Macroblock loop.* Repeat steps 2, 3, and 4 for the next macroblock until all the macroblocks in the current frame are encoded. Compute θ^K according to Eq. (4.7).

Step 6. *Frame loop.* Repeat steps 2 to 5 for the next frame until all the frames in the video sequence are encoded.

4.2.3 Performance Analysis of the Algorithms

We next analyze the computational complexity and implementation cost of Algorithms 1 and 2, and compare their running time with other rate control algorithms reported in the literature [23, 32, 35].

In Algorithm 1, at the beginning of the rate control process, the distributions of DCT coefficients in the current frame need to be generated. After each macroblock is coded, the frequencies of its DCT coefficients need to be subtracted from $\mathcal{D}_0^{N_m}(x)$ and $\mathcal{D}_1^{N_m}(x)$ to obtain the updated distributions of the rest of the uncoded macroblocks. It can be easily seen that the overall computational complexity is approximately twice the complexity of the distribution generation. Note that the distribution generation has low computational complexity and low implementation cost. Therefore, the overall complexity of Algorithm 1 is low. In Algorithm 2, we need to generate the distributions for the whole frame only at the beginning. This is the major computation in Algorithm 2 because, when determining the quantization parameter for each macroblock, we need only two additions and one multiplication. Therefore, Algorithm 2 reduces the overall complexity of Algorithm 1 by approximately half.

Next, we compare the running time of the proposed algorithms with other standard rate control algorithms, such as the MPEG-2 TM5 and H.263 TMN8 algorithms. Note that the running time depends on the CPU speed and the configuration of the computer. To circumvent this issue, we define the measure unit to be the average TM5 rate control time in MPEG-2 coding of the "Foreman" QCIF sequence. The running time simulation results are listed in Table 4.2.3. Here, ρ-RC-1 and ρ-RC-2 refer to the proposed Algorithm 1 and Algorithm 2, respectively. We can see that the ρ-RC-1 is faster than the TM5 and TMN8 rate control algorithms, and ρ-RC-2 is about twice as

Table 4.2: Ruuning time comparison between the proposed algorithms and other standard rate control algorithms.

MPEG-2			
Video	TM5	ρ-RC-1	ρ-RC-2
Foreman	1.000	0.891	0.426
Coastguard	1.125	0.829	0.469
Tabletennis	1.021	0.974	0.486
H.263			
Video	TMN8	ρ-RC-1	ρ-RC-2
News	1.405	0.992	0.493
Salesman	1.325	0.939	0.473
Akiyo	1.220	0.814	0.401

fast as ρ-RC-1.

4.3 Simulation Results

The proposed rate control algorithms do not depend on the specific coding algorithm. Therefore, they can be applied to any typical video coding systems. In this section, we present the rate control results for MPEG-2, H.263, and MPEG-4 video coding, and compare their performance with other well-known rate control algorithms reported in the literature [23, 32, 35].

4.3.1 MPEG-2 Coding

In MPEG-2, the video sequence is coded by the unit of GOP. Each GOP consists of at least one intracoded picture (I-frame) and some intercoded pictures (P- and B-frames). In general, the MPEG-2 rate control has two major steps. In the first

step, the target bit rate for each video frame inside the GOP needs to be determined by bit allocation. In the second step, the target bit rate is achieved by adjusting the quantization parameter according to the source model. We employ the TM5 bit allocation scheme [32] to determine the number of bits assigned to each frame in the current GOP. [1] For each frame, the proposed rate control algorithms are employed to achieve the target bit rate. The test videos are "Foreman", "Salesman", "Tabeltennis" and "Coastguard". The target bit rate for each test is shown in Table 4.3. To measure rate control performance, we define the relative control error as

$$E_{rc} = \frac{R - R_T}{R_T} \times 100\%, \tag{4.10}$$

where R and R_T are the actual and target coding bits of each video frame, respectively. The relative control error for each frame in "Foreman" and "Tabletennis" is plotted in Figs. 4.5 and 4.6, respectively. It can be seen that when compared to the TM5 rate control algorithm the proposed algorithms yield a much smaller control error, which is mostly less than 2%. The peak signal-to-noise ratio (PSNR) of each frame in "Foreman" and "Tabletennis" is plotted in Figs. 4.7 and 4.8, respectively. It can be seen that with the proposed rate control algorithms the picture quality is significantly improved. Also note that in the above experiment ρ-RC-2 has almost the same performance as ρ-RC-1. The picture quality improvement for the other two test videos is listed in Table 4.3. A PSNR gain of 0.87 dB on average is achieved. For some video frames, the gain is even up to 2 dB. The improved picture quality is due to our accurate source model and robust rate control scheme.

[1] In Chapter 5, we develop a distortion model which, when combined with the linear rate model, leads to an optimum bit allocation scheme in the ρ-domain.

Table 4.3: Rate control results for MPEG-2. ρ-RC-2 represents the proposed ρ-domain rate control algorithm.

Video		Foreman	Salesman	Tabletennis	Coastguard	Coastguard
Target Rate		288000	288000	288000	288000	1024000
Bit Rate	TM5	289024	289089	288208	288895	1024273
	ρ-RC-2	288203	288170	288008	288169	1024152
Control Error	TM5	12.8%	4.9%	15.2%	6.5%	2.9%
	ρ-RC-2	0.9%	1.3%	0.7%	0.8%	0.3%
PSNR (dB)	TM5	32.96	36.58	32.24	31.38	38.37
	ρ-RC-2	33.84	37.20	33.12	32.20	39.64
	Gain	0.88	0.62	0.88	0.82	1.17

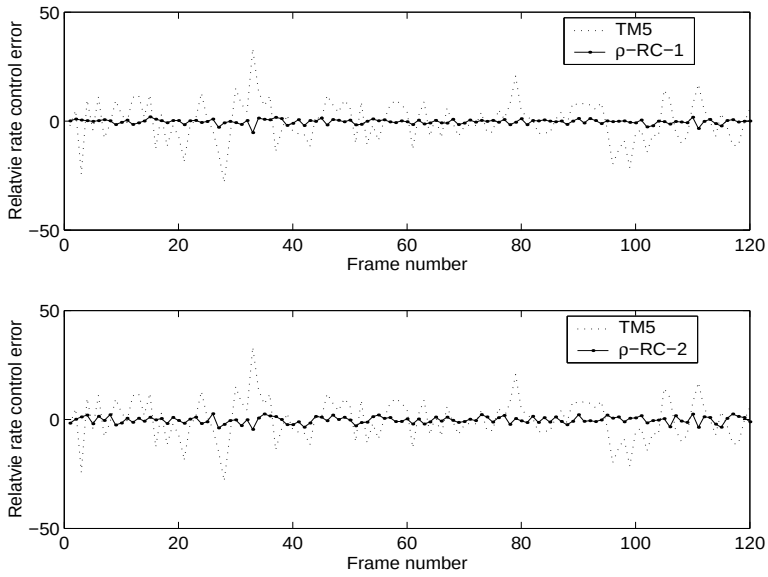

Figure 4.5: The relative bit rate control error in percentage for each frame in Forman when the proposed rate control algorithm and the TM5 algorithm are applied to the MPEG-2 coding.

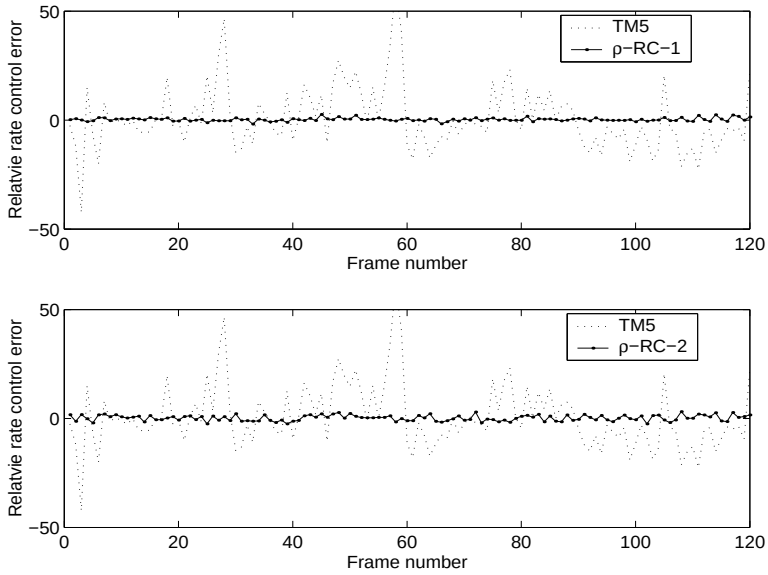

Figure 4.6: The relative bit rate control error in percentage for each frame in Tabletennis when the proposed rate control algorithm and the TM5 algorithm are applied to the MPEG-2 coding.

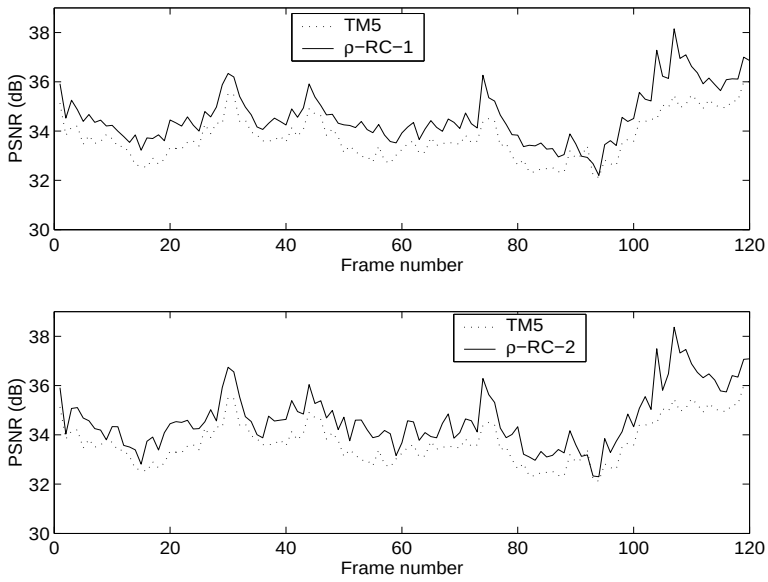

Figure 4.7: The PSNR of each frame in Forman when the proposed rate control algorithm and the TM5 algorithm are applied to the MPEG-2 coding.

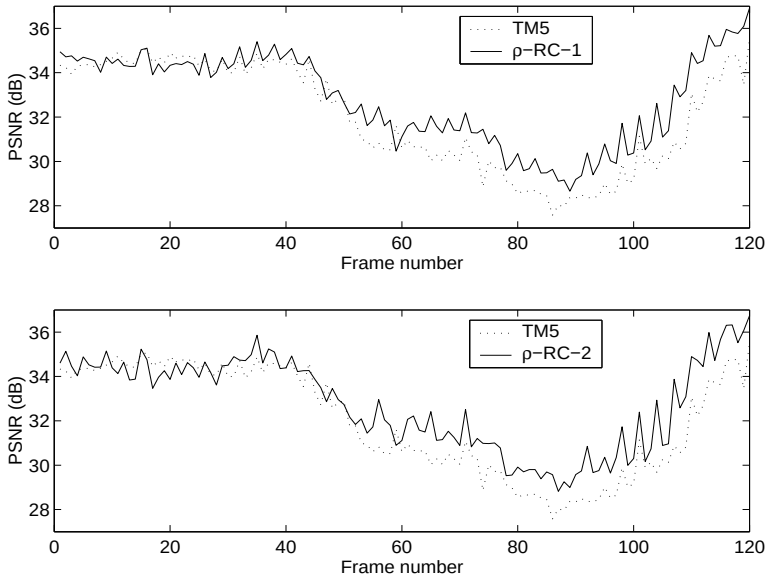

Figure 4.8: The PSNR of each frame in Tabletennis when the proposed rate control algorithm and the TM5 algorithm are applied to the MPEG-2 coding.

4.3.2 H.263 Coding

In real-time video coding with H.263, the time delay should be very small, which imposes strict requirement on the rate control accuracy. In the following experiment, we compare the proposed low-complexity ρ-RC-2 rate control algorithm with the TMN8 algorithm [23] which is one of the best available rate control algorithms for video coding. The configuration of each test is shown in Table 4.4. The frame rate is fixed at 10 frames/second (fps). We plot the number of bits in the buffer for each test in Fig. 4.9, which shows that the proposed rate control algorithm maintains a steadier buffer level than does the TMN8 algorithm, especially at lower coding bit rates. The number of bits produced by each frame is plotted in Fig. 4.10. With the proposed algorithm, the output bit rate of the video encoder is well matched to the target bit rate or the channel bandwidth. The average PSNR of the luminance components in each test is listed in Table 4.4. The proposed algorithm achieves a slightly improved picture quality because of its more robust rate control scheme.

Table 4.4: Description of the rate control tests with the H.263 codec.

Test Name	Video Name	Target Rate	PSNR TMN8	PSNR This work
fm48	Foreman	48 kbps	30.31	30.41
sm32	Salesman	32 kbps	30.92	30.71
nw24	News	24 kbps	30.28	30.63
cg48	Coastguard	48 kbps	28.43	28.55

4.3.3 MPEG-4 Coding

We use the MoMuSys MPEG-4 codec [61] with the H.263-type quantization scheme to test the proposed rate control algorithm and the VM8 algorithm. The two test videos are "Carphone" and "News". We treat the whole scene as one video object. The frame rate is 10 fps. The target bit rate is 64 kbps. The number of bits produced by each frame in "Carphone" and "News" are plotted in Figs. 4.11 and 4.12, respectively. With the proposed algorithm, an almost constant bit rate is achieved. The relative control error is less than 1%. However, in the rate control of VM8, the bit rate variation is much larger.

The proposed rate control algorithm has also been tested using other target bit rates and video sequences. The simulation results, along with the results presented in the above, show that the proposed algorithm provides a much more robust and accurate rate control than other algorithms reported in the literature.

4.4 Picture Quality Evaluation

In MPEG-2 TM5 and H.263 TMN8 rate control algorithms, the quantization parameter QP is determined at the macroblock level. There is no restriction for its range.

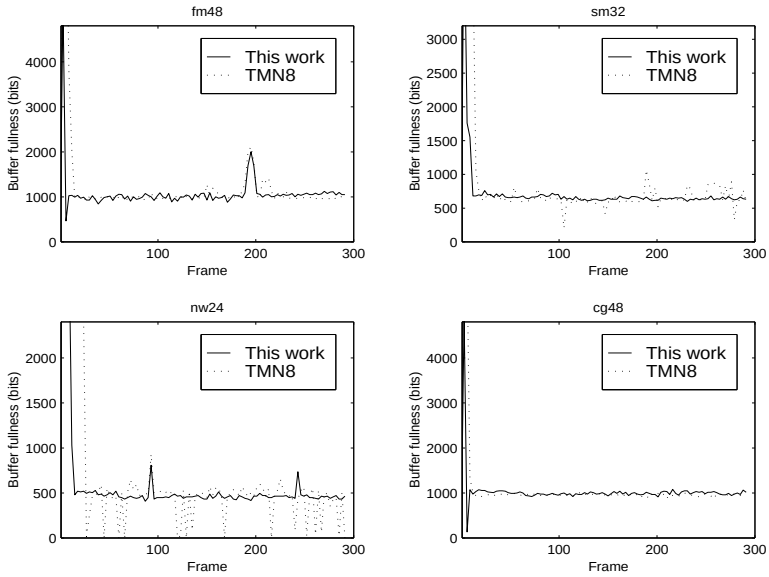

Figure 4.9: The number of bits in the buffer when the proposed algorithm (solid line) and the TMN8 rate control algorithm (dotted line) are appplied to the H.263 video coding.

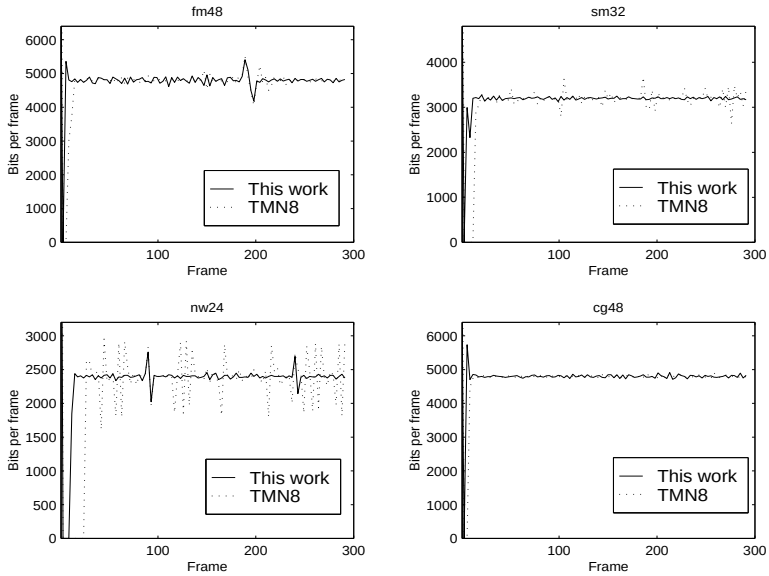

Figure 4.10: The number of bits produced by each encoded frame when the proposed algorithm (solid line) and the TMN8 rate control algorithm (dotted line) are applied to the H.263 video coding.

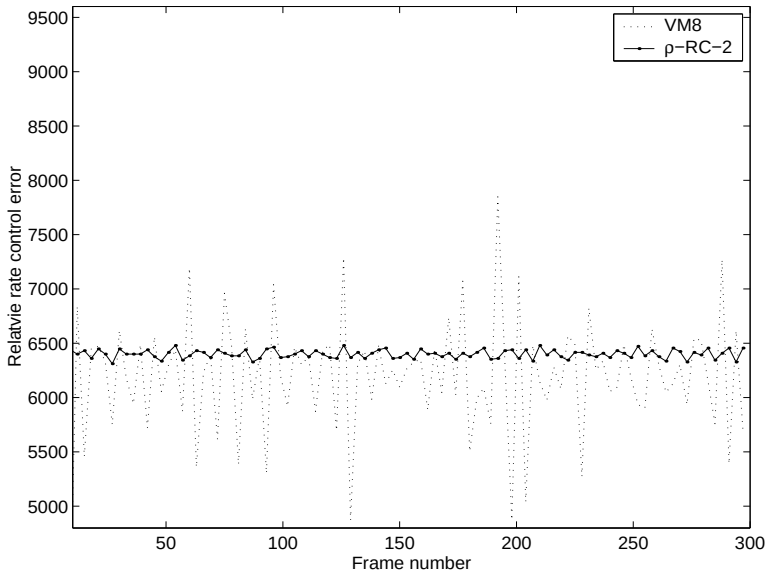

Figure 4.11: The number of bits produced by each encoded frame in Carphone when the proposed algorithm (solid line) and the VM8 rate control algorithm (dotted line) are applied to the MPEG-4 coding.

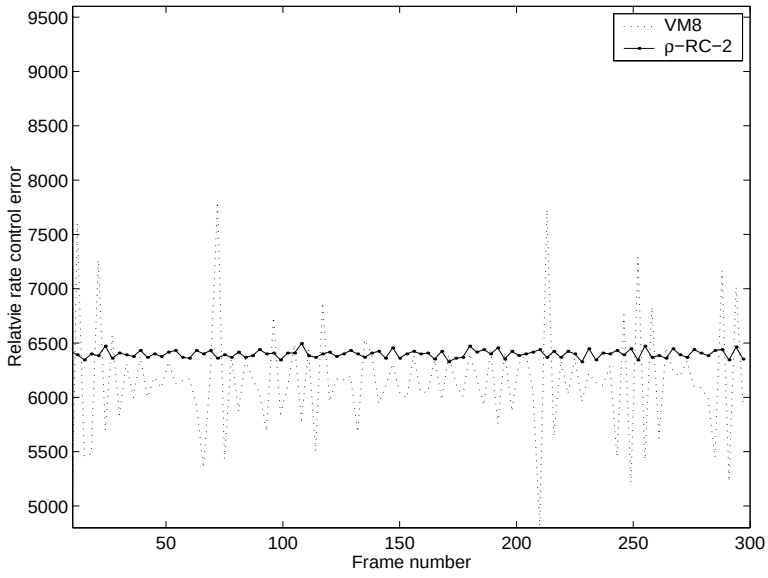

Figure 4.12: The number of bits produced by each encoded frame in News when the proposed algorithm (solid line) and the VM8 rate control algorithm (dotted line) are applied to the MPEG-4 coding.

However, in the proposed ρ-RC-2 algorithm, based on the distribution of the DCT coefficients, we first determine the mean quantization parameter $\bar{Q}^K$ of the whole frame. The quantization parameter QP of each macroblock is then chosen from its neighborhood $\mathcal{Q}$.

Let QP_{max} be the maximum and QP_{min} be the minimum macroblock quantization parameters in the current frame. The dynamic range of QP is then given by $QP_{max} - QP_{min}$. If the range is 0, then all the macroblocks in the current frame use the same quantization parameter. In Fig. 4.13 we plot the dynamic range of QP for each frame when the ρ-RC-2 algorithm and the TM5 algorithm are applied in MPEG-2 coding of the "Tabletennis" QCIF sequence. The dynamic range comparison results with the TMN8 rate control algorithm in H.263 coding of the Carphone sequence is depicted in Fig. 4.14. Note that in this experiment, according to our experience, we have enlarged the quantization parameter relaxation range where $\mathcal{Q} = \{\bar{Q}^K - 2, \bar{Q}^K - 1, \bar{Q}^K, \bar{Q}^K + 1, \bar{Q}^K + 2, \bar{Q}^K + 3\}$. This is because, in H.263 video coding, the coding bit rate is normally low. Therefore, the corresponding quantization parameter is normally large, especially at very low bit rates such as the 24 kbps rate in the above experiment. The maximum range of the proposed algorithm is 6. However, in the TMN8 rate control algorithm, the range could be very large, as large as 29. Note that in standard video coding, the maximum quantization parameter is 31. This means, in TMN8, some blocks in the current frame are quantized by an extremely small QP while others are quantized by an extremely large QP. This implies a very large quality variation inside each picture. Consequently, the overall picture quality degrades because of the lack of uniformity. Fig. 4.15 shows the 111-th frame reconstructed by the H.263 encoder when the TMN8 algorithm and the ρ-RC-2 algorithm are applied. The proposed algorithm yields a better picture quality. This improvement in picture quality is due to the inherent difference between the two rate control algorithms. In the MPEG-2 TM5 or

H.263 TMN8 rate control algorithm (since the source model is not accurate enough), the model parameters are empirically adjusted during the encoding process.

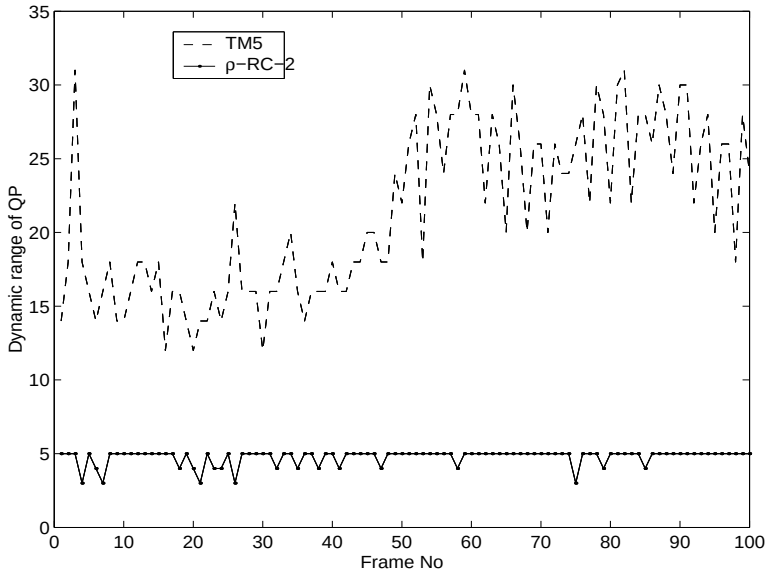

Figure 4.13: The dynamic range of the quantization parameters in each frame of Tabletennis.qcif at 576 kbps when the proposed ρ-RC-2 algorithm (solid line) and the TM5 rate control algorithm (dashed line) are applied to the MPEG-2 coding.

Let us consider the following example. Suppose the target number of bits for the current frame is 4000 and the corresponding mean quantization parameter is 10. When encoding the first several macroblocks, the TMN8 estimates $QP = 2$ because its source model is not so accurate. After some time, it suddenly realizes that there are very few bits left for the remaining macroblocks. Then the TMN8 adjusts its model parameters and dramatically increases the values of QP. At the end part of the frame, the maximum $QP = 31$ is employed to reduce the bits as much as possible. In this way, the dynamic range of QP becomes very large.

Another disadvantage of the very large dynamic range of QP is that some important information in the picture might be lost. For example, suppose that the region of interest is at the bottom half of the picture. The TMN8 rate control algorithm uses very

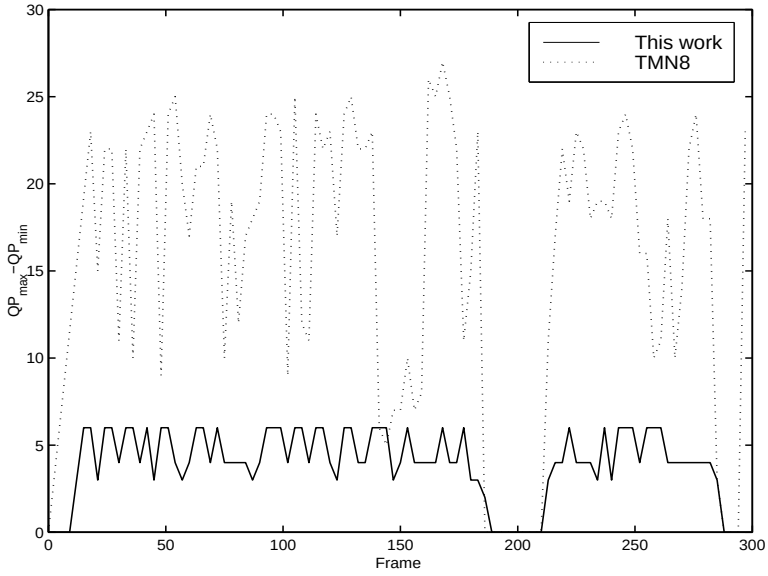

Figure 4.14: The dynamic range of the quantization parameters in each frame of Carphone.qcif at 24 kbps when the proposed ρ-RC-2 algorithm (solid line) and the TMN8 rate control algorithm (dotted line) are app lied to the H.263 coding.

Figure 4.15: The reconstructed 111th frame in Carphone.qcif when the TMN8 algorithm (left) and the proposed ρ-RC-2 algorithm (right) are applied to H.263 video coding. The channel rate is 24 kbps.

large QP to quantize this part. Its quality will be poor and the valuable information will be lost. However, in the proposed rate control algorithm, since the quantization parameter for each macroblock is restricted inside a small interval, a steady and uniform picture quality is guaranteed.

4.5 Summary

Based on the linear ρ-domain rate model developed in Chapter 3, very simple rate control algorithms were proposed. The proposed algorithms can be applied directly to any typical video coding systems. Our simulation results show that, compared to the standard rate control algorithms, the proposed rate control algorithms have lower computational complexity and implementation cost. However, the rate control is much more accurate and robust, especially in MPEG-2 video coding. The simplicity of the linear rate model and the superior performance of the proposed rate control algorithms have demonstrated the great advantage of our ρ-domain analysis. In the next Chapter, we develop a distortion model in the ρ-domain. Furthermore, when combined with the linear rate model in Chapter 3, an optimum bit allocation scheme is developed in the ρ-domain. It is then applied to practical video coding to improve the rate control and coding performance.

Chapter 5

ρ-Domain Optimum Bit Allocation

The ultimate objective in video coding and transmission is to provide the best video quality at the receiver end, given the constraint of certain network conditions. Obviously, maximizing the picture quality means the minimizing the distortion of the reconstructed video. Given a bit budget, the best picture quality or minimum distortion can be achieved by optimum bit allocation [27, 28, 29, 30].

In the previous two chapters, a linear rate model and unified rate control algorithms have been developed in the ρ-domain for all standard video coding systems. We observe that in the ρ-domain the rate function has a unqiue behavior that enables us to develop the accurate and robust rate control algorithms. This motivates us also to develop a distortion model in the ρ-domain. Once both the rate model and the distortion model are available, we can then develop an optimum bit allocation scheme in the ρ-domain, which further extends the capability of the rate control algorithms proposed in Chapter 4.

This chapter is organized as follows: First, a generic distortion model for transform coding is developed in the ρ-domain. Second, based on this distortion model and the linear rate model in Chapter 3, a ρ-domain optimum bit allocation scheme is

presented. Third, we apply this bit allocation scheme to MPEG-4 coding to distribute the bit budget among different video objects. Finally, with a macroblock classification scheme, we also apply it to H.263 video coding at the macroblock level to improve the picture quality.

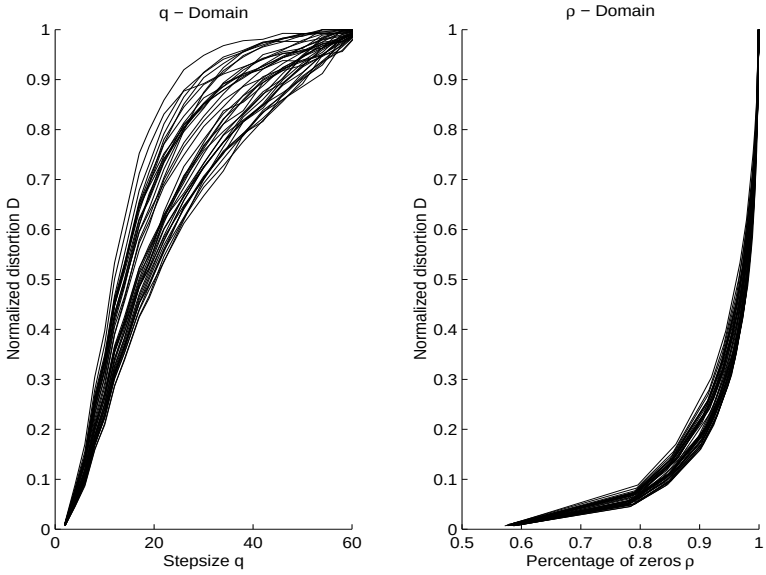

Figure 5.1: The distortion curves of each frame in Foreman.qcif in the q-domain (left) and the ρ-domain (right).

5.1 ρ-Domain Distortion Model

It is much easier to model the distortion than the coding bit rate for two reasons. First, in typical transform coding systems, all of the information loss occurs during the quantization process of the transform coefficients. This implies that to model the distortion we need to consider only the behavior of the quantizer. However, the coding bit rate is affected by every component of the transform coding system: quantization, data representation, and entropy coding. Furthermore, among these components quantization is the simplest and easiest to model. The second reason for the difficulty in rate modeling is that, in image/video coding, the transform coefficients are often grouped

to be jointly coded as one symbol. For example, in EZW coding, all of the insignificant coefficients in a spatial orientation tree are coded with one ZTR symbol. In JPEG and MPEG coding, each non-zero coefficient is jointly coded with its preceding zeros using a codedword from their Huffman tables. This type of vector-based coding scheme makes it very difficult to directly model the coding algorithm and to estimate the final coding bit rate. However, in distortion modeling, since each transform coefficient is quantized independently, the distortion of the whole picture is just a summation of the distortion at every pixel. This implies that the distortion formula can be explicitly derived from the distribution of the transform coefficients.

In the classical R-D analysis, the distortion D is considered to be a function of the quantization parameter q, denoted by $D(q)$. From the previous two chapters we know that the rate function $R(\rho)$ has a unique behavior in the ρ-domain. This motivates us to also study the distortion function in the ρ-domain. Let $\bar{D} = D/\sigma^2$ be the normalized distortion. In Fig. 5.1 we plot the normalized distortion function in the q-domain $\bar{D}(q)$ and the function in the ρ-domain $\bar{D}(\rho)$ for each video frame of Foreman coded by H.263. Two observations are made from these plots. First, in the q-domain, $\bar{D}(q)$ is defined within an infinity range $[0, +\infty)$. However, in the ρ-domain, $\bar{D}(\rho)$ is defined within a finite range [0, 1]. Note that when $\rho \to 0$, which means the quantization parameter q is very small, we have $\bar{D}(\rho) \to 0$. When $\rho \to 1$, which means the quantization parameter q is very large, we have $\bar{D}(\rho) \to 1$. Second, in the q-domain, for different video frames, the plots of $D(q)$ are quite different from each other. However, in the ρ-domain, for different video frames, the variation of $D(\rho)$ is very small. This implies that the distortion function has a more robust and regulated behavior in the ρ-domain than in the q-domain. We find that, for each video frame, $D(\rho)$ can be well approximated by

$$D(\rho) = \sigma^2 e^{-\alpha(1-\rho)}, \tag{5.1}$$

where α is a constant that normally ranges from 10 to 20.

Assume that the DCT coefficients have a Laplacian distribution [55]. For a uniform threshold quantizer with a dead zone $(0.5 + b) \cdot q$ where q is the quantization stepsize, from Section 3.2, we know that the corresponding distortion $D(\rho)$ is given by

$$D(\rho) = \frac{1 + \eta^a - 2\eta + (1 - a)(1 + \eta^a)\eta \ln \eta}{\lambda(1 + \eta^a)}, \tag{5.2}$$

where $a = 1/(1 + 2b)$ and $\eta = 1 - \rho$. Obviously, this distortion function is highly nonlinear and complex. It will be very difficult to develop a close-form formula for the optimum bit allocation based on this complex expression. However, we observe that $D(\rho)$ given by Eq. (5.2) can be approximated very closely by Eq. (5.1), which is much simpler. Consequently, we employ Eq. (5.1) instead of Eq. (5.2) as our ρ-domain distortion model.

5.2 Optimum Bit Allocation

Based on the distortion model developed in the previous section and the linear rate model presented in Chapter 3, a ρ-domain optimum bit allocation scheme is developed in this section. Before the formulation of our bit allocation scheme, we briefly review some existing bit allocation schemes in the literature.

5.2.1 Bit Allocation in Review

In transform coding of images and videos, bit allocation is employed to distribute the bits budget among different groups of transform coefficients to achieve the minimum overall quantization distortion. The problem of optimum bit allocation was first addressed by Huang and Schultheiss [62] where only an approximate solution to the problem was provided. Further improvement has been suggested in [27, 63, 64] within the context of source quantization and coding. The optimum bit allocation scheme can be applied to various image coding algorithms to improve their coding performance,

such as JPEG [65] and wavelet-based image coding [66, 67, 68]. In video coding, bit allocation can be incorporated into the rate control algorithm to further extend the capability of the control algorithm and to improve the video presentation quality [23, 32, 41].

The optimum bit allocation is carried out based on the R-D functions of the encoder. The R-D functions can be modeled theoretically and employed to derive the close-form expression for the optimum bit allocation scheme as in [23, 27, 62, 63, 64]. As we have observed in Section 1.1, the theoretical models in the conventional R-D analysis often suffer from relatively large estimation error. As a result, the optimum bit allocation based on these R-D models cannot be truly optimum [24]. For this reason, in practical image/video coding more accurate operational R-D curves are employed to perform the optimum bit allocation [38, 66, 67]. Since the generation of the operational R-D curves often has very high computational complexity, this type of operational bit allocation scheme does not work efficiently in practical video applications, especially in real-time video coding and transmission. In this dissertation, we show that the R-D functions have unique properties in the ρ-domain and develop simple and accurate models for the R-D functions. Based on these models, we can then develop an optimum bit allocation scheme in the ρ-domain.

5.2.2 ρ-Domain Bit Allocation

In transform coding of images and videos, we need to take two major steps to achieve the best picture quality. The first step is the optimum bit allocation. Specifically, we need to determine the number of bits assigned to each data source in such a way that the overall distortion is minimized. Here, "source" is a generic term. In video coding, it could be a frame, a video object, or a group of macroblocks inside one frame. In the second step, we need to select the quantization parameter accurately to meet the bit budget for each source, which is exactly the problem of rate control. In Chapter

4 we solved this problem by developing a ρ-domain rate control scheme. So, the only remaining issue is to develop a ρ-domain optimum bit allocation.

In the ρ-domain, the rate and distortion functions for each input source are given by Eqs. (3.1) and (5.1), respectively. Let $\{S_i|1 \leq i \leq L\}$ be the input sources. For each S_i, we have

$$R_i(\rho_i) = \theta_i \cdot (1 - \rho_i), \tag{5.3}$$

$$D_i(\rho_i) = \sigma_i^2 e^{-\alpha_i(1-\rho_i)}. \tag{5.4}$$

The optimum bit allocation problem can then be formulated as follows,

$$\min_{\rho_i} \quad \sum_{i=1}^{L} \sigma_i^2 e^{-\alpha_i(1-\rho_i)} \cdot N_i, \tag{5.5}$$

$$s.t. \quad \sum_{i=1}^{L} \theta_i \cdot (1 - \rho_i) \cdot N_i = R_T, \tag{5.6}$$

where N_i is the size of S_i and R_T is total number of bits available. With the Langrange multiplier, the constrained minimization problem can be converted to the following unconstrained problem,

$$\min_{\rho_i} \quad \sum_{i=1}^{L} \sigma_i^2 e^{-\alpha_i(1-\rho_i)} \cdot N_i + \lambda \cdot [\sum_{i=1}^{L} \theta_i \cdot (1 - \rho_i) \cdot N_i - R_T]. \tag{5.7}$$

By solving this minimization problem, we obtain the optimum number of bits for each input source,

$$R_i = \xi_i N_i \ln \frac{\sigma_i^2}{\xi_i} + \frac{\xi_i N_i}{\sum_{i=1}^{L} \xi_i N_i}(R_T - \sum_{i=1}^{M} \xi_i N_i \ln \frac{\sigma_i^2}{\xi_i}), \tag{5.8}$$

where $\xi_i = \theta_i/\alpha_i$. According to Eq. (5.3), the corresponding percentage of zeros to be produced is given by

$$\rho_i = 1 - \frac{B_i}{\theta_i N_i}. \tag{5.9}$$

In the following, we apply this optimum bit allocation scheme to practical video coding.

5.3 Scalable Rate Control for MPEG-4

The ISO MPEG-4 video coding supports content-based interactivity which allows the access and manipulation of video objects in the compression domain [10]. To do this, each video frame is segmented into several objects associated with some physical meaning, such as foreground people and background scene, as illustrated in Fig. 5.2. Each video object is then coded separately. The MPEG-4 output bit stream syntax also allows the separate decoding and reconstruction of each video object. In this way, scene editing, such as adding, deleting, or moving a video object can be performed directly on the bit stream. With this type of content-based coded representation of video data, the video information can used and presented much more flexibly.

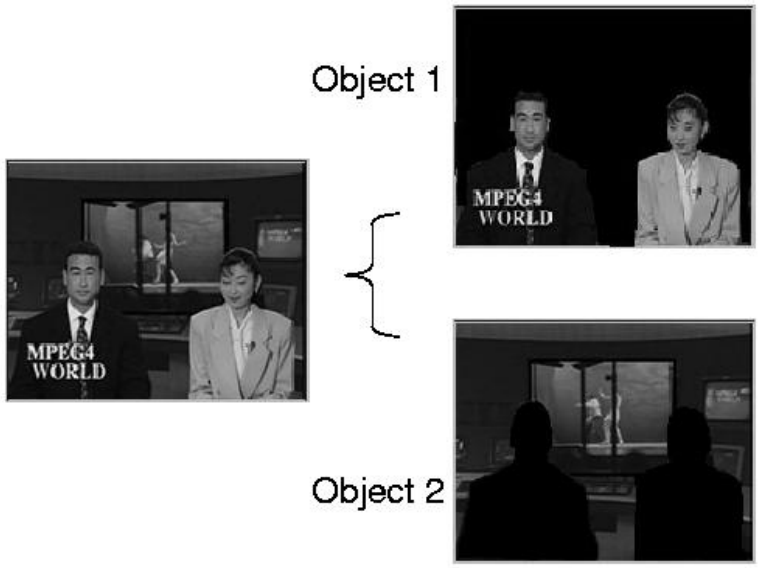

Figure 5.2: Illustration of the video objects segmentation in MPEG-4 coding.

In rate control for MPEG-4 coding, the available bit budget R_T for each video frame can be determined from the channel bandwidth and the buffer status as described in Section 4.2. The optimum bit allocation scheme developed in Section 5.2 can then be employed to efficiently distribute these bits among the video objects to maximize the

overall picture quality. Once the bits target R_i for each video object is obtained, the macroblock-level rate control algorithm ρ-RC-2, proposed in Chapter 4.2, can be then employed to control the encoder to achieve the bits target R_i.

5.3.1 Model Parameters

Note that, in the bit allocation scheme just described, there are two model parameters θ_i and α_i to be determined. In practical video coding, these two parameters of the current object can be determined by the coding statistics of the same object in the previous frame of the same type. To be more specific, after coding the k-th frame, we already know the number of bits R_i used for coding the object VO_i, the percentage of zeros ρ_i produced by VO_i, and the distortion D_i. With Eqs. 5.3 and 5.4, the values of θ_i and α_i can be determined as follows,

$$\theta_i = \frac{R_i}{1-\rho_i}, \tag{5.10}$$

$$\alpha_i = \frac{1}{1-\rho_i} \ln \frac{\sigma_i}{D_i}. \tag{5.11}$$

As depicted in Fig. 5.3, they are then used for the object VO_i in the current (k+1)-th video frame. Note that, after scene segmentation, each video object becomes more homogeneous. The temporal variation of the model parameters is significantly reduced. As a result, this type of frame-level estimation of the model parameters works quite well in practice. Based on our experience, the initial values of θ_i and α_i are set to 7 and 12, respectively.

5.3.2 Experimental Results

We incorporate the proposed bit allocation and rate control algorithm into the MoMuSys MPEG-4 codec [61] and compare it to the VM8 rate control scheme [34]. The test QCIF video is "News" at 64 kbps. The frame rate is 10 fps. Fig. 5.4 shows the number of bits produced by each video frame when the proposed algorithm and the

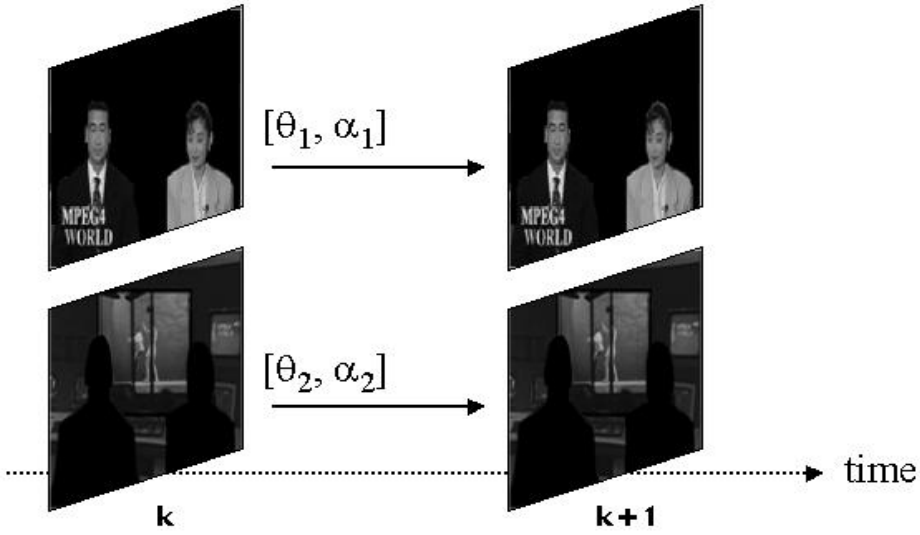

Figure 5.3: Determination of the model parameters.

VM8 algorithm are applied. The actual coding bit rate is much closer to the target bit rate when the proposed algorithm is applied. The numbers of bits assigned to each video object are depicted in Fig. 5.5. Note that Object 2 takes more bits than Object 1. This is because the major activity in the scene is from the dancers (center background) in Object 2. Fig. 5.6 shows the PSNR of each video frame. With the proposed bit allocation scheme, we see that the picture quality is significantly improved. The improvement, about 1.25 dB on average, achieved by the proposed algorithm is due to its more accurate R-D models and more robust rate control.

It is straightforward to extend the proposed bit allocation scheme to take the subjective quality into account. For example, in general, the user is more interested in the moving foreground objects in the scene. In our bit allocation and rate control scheme, we can assign more bits to these objects and code them with higher fidelity. This can be realized by introducing a distortion weight w_i for each object into the objective

function as follows,

$$\min_{\rho_i} \quad \sum_{i=1}^{L} w_i \sigma_i^2 e^{-\alpha_i(1-\rho_i)} \cdot N_i, \tag{5.12}$$

$$s.t. \quad \sum_{i=1}^{L} \theta_i \cdot (1-\rho_i) \cdot N_i = R_T. \tag{5.13}$$

In this case, the optimum number of bits assigned to each object is given by,

$$R_i = \xi_i w_i N_i \ln \frac{\sigma_i^2}{\xi_i} + \frac{\xi_i w_i N_i}{\sum_{i=1}^{M} \xi_i w_i N_i} (B - \sum_{i=1}^{M} \xi_i w_i N_i \ln \frac{\sigma_i^2}{\xi_i}), \tag{5.14}$$

Obviously, the objects of interest should have relatively larger weights to guarantee that they are coded with less distortion.

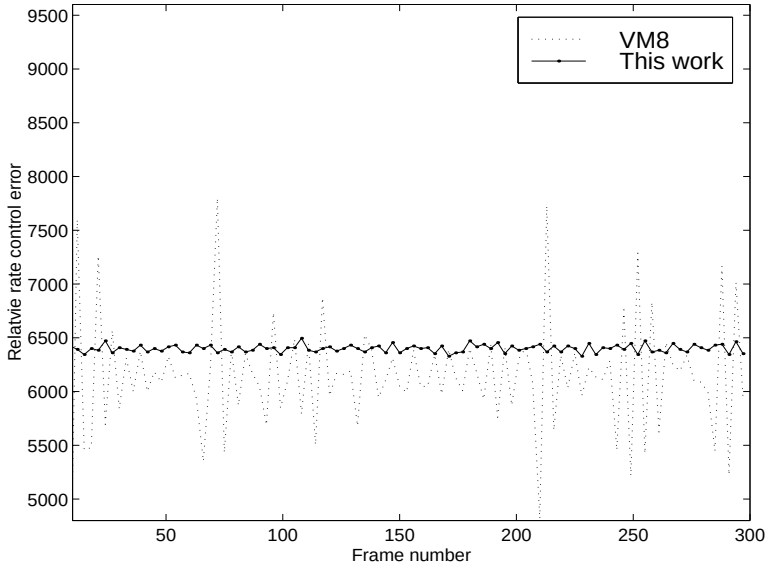

Figure 5.4: Bits per frame when the proposed algorithm (solid line) and the VM8 algorithm (dotted line) are applied to the MPEG-4 codec.

5.4 R-D Optimized Coding for H.263

In the above section, the optimum bit allocation scheme developed in Section 5.2 is applied to MPEG-4 coding at the video object level. As we can see in this section, it

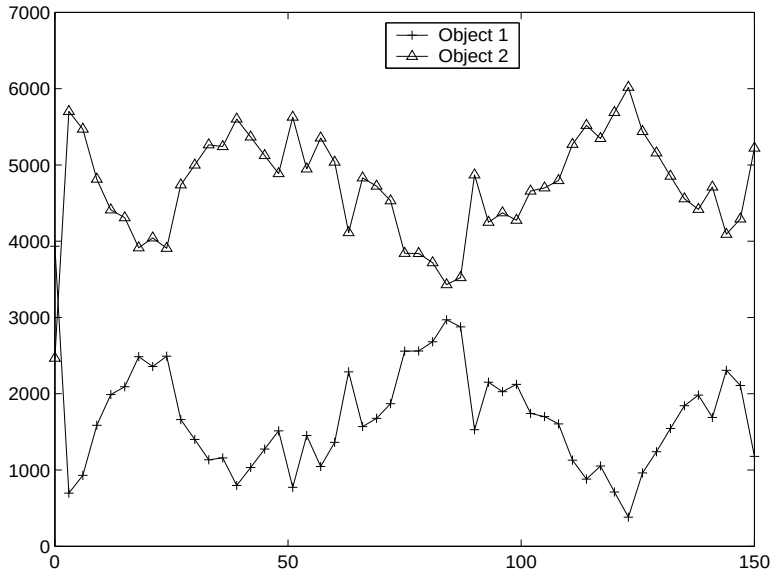

Figure 5.5: Bits assigned to each video object in the News sequence when the optimum bit allocation scheme is applied to the MPEG-4 codec.

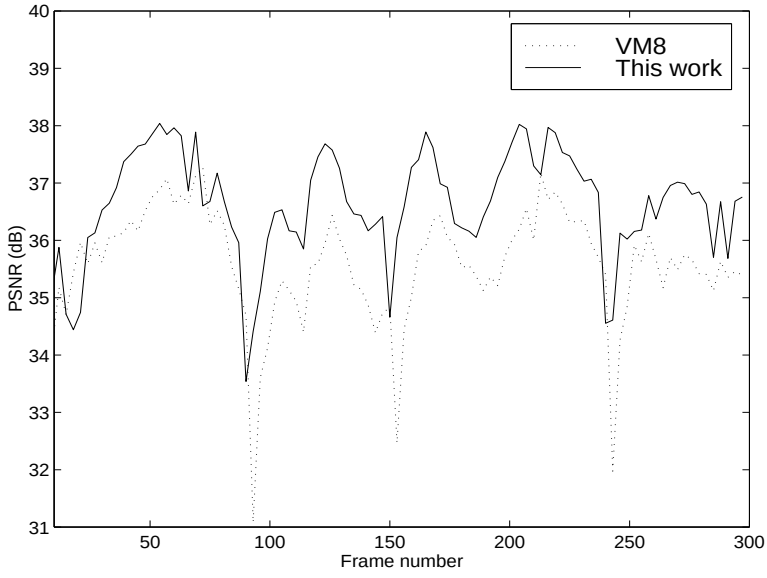

Figure 5.6: PSNR of each frame when the proposed algorithm (solid line) and the VM8 algorithm (dotted line) are applied to the MPEG-4 codec.

can also be applied to video coding at the macroblock level. In a QCIF video frame, there are 99 16×16 macroblocks. We observe that it is not efficient to apply the bit allocation scheme directly to distribute the bit budge among these 99 macroblocks for the following two reasons: First, a large number of data sources implies that the average number of bits for each source is very small. Effective distribution of a smaller number of bits requires more accurate R-D models in the bit allocation scheme. Second, these 99 macroblocks may have a wide range of R-D characteristics. For example, some macroblocks may be very active while others may be almost black. (Their coefficients are very close to zeros.) This will introduce strong singularity into the bit allocation scheme which often results in negative bits assigned to those almost black macroblocks.

To improve the efficiency and robustness of the optimum bit allocation scheme, we first classify the 99 macroblocks into three classes according to their activity. The bit allocation scheme is then employed to distribute the bit budget among these three classes instead of among the 99 macroblocks. We observe that, after classification and grouping, the singularity introduced to the allocation process is significantly reduced. According to our simulation experience, "three" is a good choice for the class number with which the optimum bit allocation operates most effectively and robustly.

5.4.1 Macroblock Classification

The activity measure we choose for macroblock classification is the variance of the macroblock, denoted by $\{\sigma_j^2 | 1 \leq j \leq N\}$, where N is the total number of macroblocks in the current video frame. In our classification scheme, we first rearrange all of the macroblocks in decreasing order according to their variances. The first class S_1 consists of the top 10 macroblocks which are the most active. The third class S_3 consists of the last 69 macroblocks which are most inactive. The remaining 20 macroblocks are placed into the second class S_2. The class sizes 10, 20, and 69 are chosen based on our experience. Certainly, they can be adaptively adjusted according to the specific picture

characteristics.

Each class is then treated as a separate input source. The parameters of the R-D models for each source can be determined by the interframe prediction presented in Section 5.3.1. To be more specific, after coding the previous frame, we already know the number of bits R_i used for coding the class S_i, the number of zeros produced by S_i and its distortion D_i. With Eqs. 5.3 and 5.4, the values of θ_i and α_i can be determined. They are then used for the class S_i in the current video frame. With the optimum bit allocation scheme, the number of bits assigned to each class is determined by Eq. (5.8). The corresponding percentage of zeros to be produced by this class is given by Eq. (5.9).

5.4.2 Determination of the Quantization Parameter

With macroblock classification and bit allocation, the target number of bits R_i for each class are obtained. Each class of macroblocks can be treated as a generalized video object. Accordingly, the linear rate model developed in Chapter 3 also applies to each class. Therefore, the rate control algorithms proposed in Chapter 4 can be employed to achieve the bits target R_i for each class during the coding process.

5.4.3 Experimental Results

We incorporate the proposed bit allocation scheme and rate control algorithm into the H.263+ codec [53] and compare it with the TMN8 rate control scheme [23]. The two test videos are "Foreman" at 48 kbps and "News" at 24 kbps. The frame rate is fixed at 10 fps. Fig. 5.7 shows the number of bits produced by each video frame when the proposed algorithm and the TMN8 algorithm are applied. The numbers of bits assigned to each macroblock class are depicted in Figs. 5.8 and 5.9. The actual coding bit rate is shown to be much closer to the target bit rate when the proposed algorithm is applied, especially at lower bit rates. Fig. 5.10 shows the PSNR value

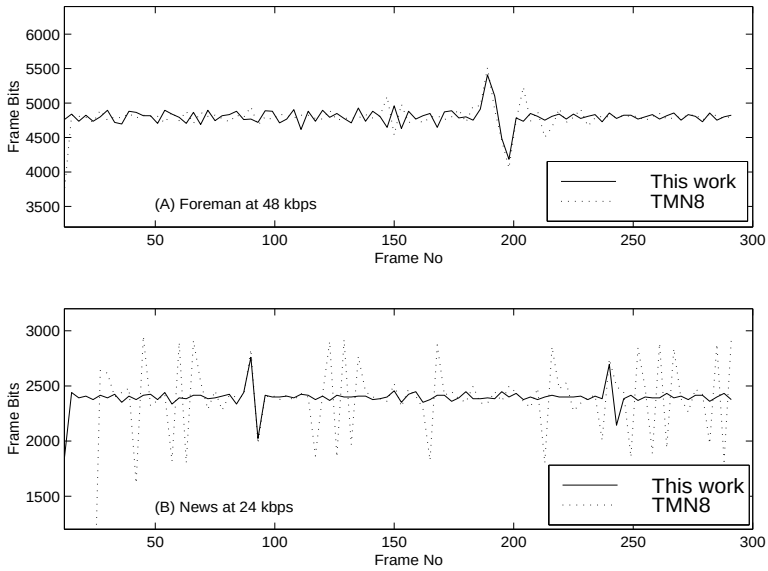

Figure 5.7: Bits per frame when the proposed algorithm and the TMN8 algorithm are applied to the H.263 codec: (A) Foreman at 48 kbps; (B) News at 24 kbps.

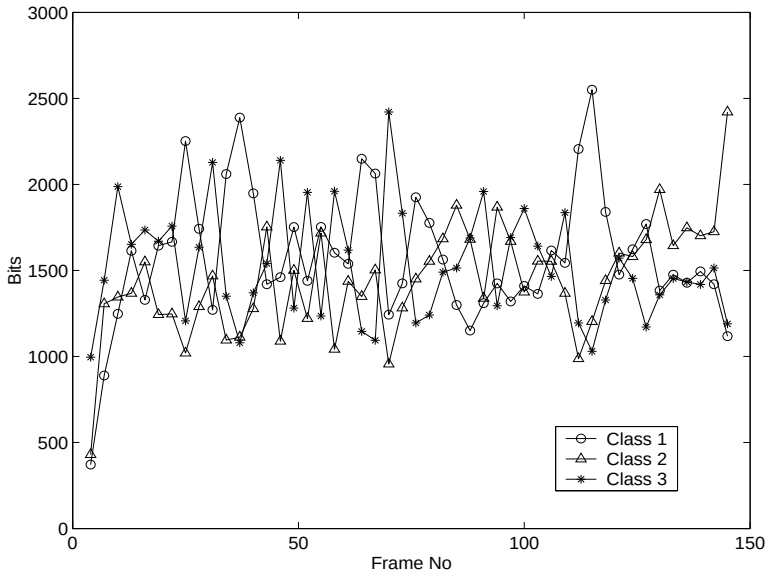

Figure 5.8: The numbers of bits assigned to each macroblock class when the proposed bit allocation scheme is applied to H.263 coding of Foreman.

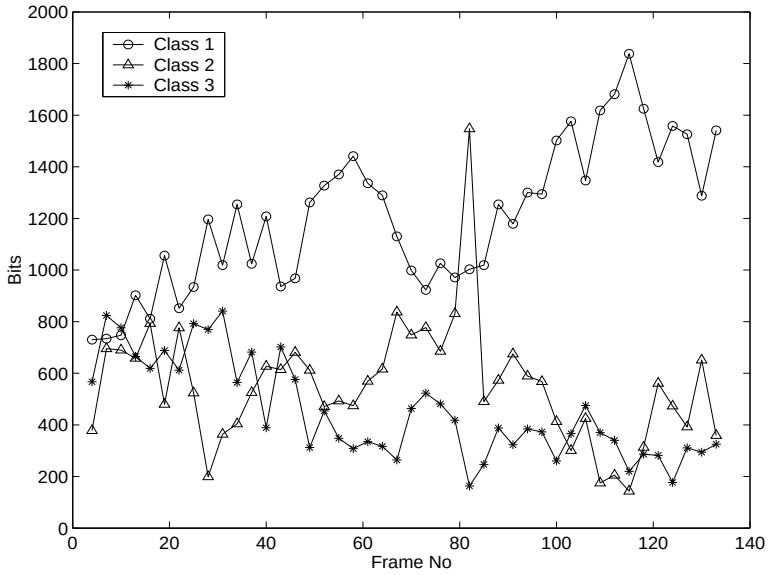

Figure 5.9: The numbers of bits assigned to each macroblock class when the proposed bit allocation scheme is applied to H.263 coding of News.

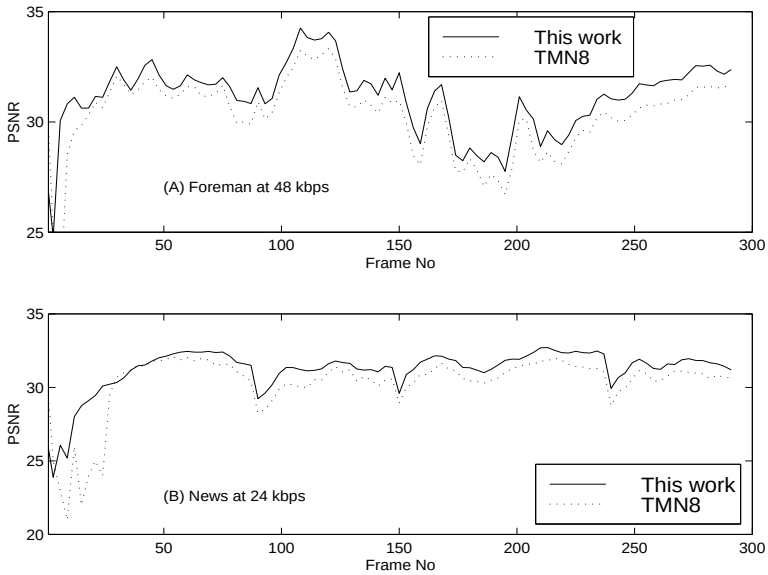

Figure 5.10: PSNR of each frame when the proposed algorithm and the TMN8 algorithm are applied to the H.263 codec: (A) Foreman at 48 kbps; (B) News at 24 kbps.

of each video frame. With the proposed bit allocation scheme, the picture quality is significantly improved. Note that the TMN8 rate control algorithm has already included an optimum bit allocation scheme. But, the TMN8 bit allocation is based on the traditional q-domain R-D formulas [21, 23]. The improvement, about 0.8–1.0 dB, achieved by the proposed algorithm is due to our more accurate rate and distortion models.

The macroblock classification can also be combined with other functionalities, such as motion tracking [69, 70] and region of interest coding [71, 72]. In this way, the macroblock classification is closely related to the user's interests and requirements. In the optimum bit allocation, we can assign different distortion weights to different regions. In this way, the region of interest can be assigned more bits and coded with higher quality.

5.5 Summary

In this chapter, a distortion model is first developed in the ρ-domain. Based on this distortion model and the linear rate model developed in Chapter 3, an optimum bit allocation scheme is developed. It is then applied to MPEG-4 video coding to distribute the bit budget among different video objects. It has also been applied to H.263 coding at the macroblock-level when coupled with macroblock classification. The proposed bit allocation scheme extends the capability of the rate control algorithm developed in Chapter 4. In addition, significant picture quality improvement is achieved due to its accurate R-D models.

0

Chapter 6

ρ-Domain Source Modeling

In the previous chapters, we developed a linear rate model and a distortion model in the ρ-domain. The linear rate model is simple and accurate since it is derived directly from the practical coding results. However, it has a model parameter θ which is the slope of the rate line. In video coding, the coding statistics of previous frames can be used to estimate the value of θ for the current frame. Typical video coding syntax, such as MPEG and H.263 standards, support macroblock-level adaptive quantization. Therefore, θ can also be adaptively estimated at the macroblock level as in the ρ-RC-1 rate control algorithm. However, typical still image coding algorithms, such as EZW, SPIHT, stack-run, and JPEG image coding, do not support adaptive quantization. In other words, the quantization of each transform coefficient is non-adaptive and does not depend on the quantization of its preceding coefficients. The quantization of all transform coefficients is controlled only by the picture quantization parameter, denoted by QP.

The lack of the coding statistics of previous data and the non-adaptive quantization scheme make it much more difficult to select the value of QP to achieve the target coding bit rate or picture quality in still image coding than in video coding. In this chapter we develop a source modeling framework which can explicitly estimate the R-D functions

before quantization and coding with very low computational complexity. Based on the accurately estimated R-D curves, as illustrated in Fig. 1.1, we can determine the value of QP to achieve the target bit rate or picture quality. This chapter is organized as follows: After a brief review of existing approaches for source modeling, a new ρ-domain framework for source modeling is presented. We show that it is a unified framework for all typical transform coding systems, not only for transform coding of still images, but all also for DCT-based video coding, such as H.263 and MPEG. The proposed framework is used to estimate the the R-D curves for typical still image coding systems such as SPIHT, stack-run, and JPEG. It is also employed for the rate control in video coding. Furthermore, based on the estimated R-D curves, an encoder-based rate smoothing algorithm is developed which can control the encoder such that the output bit stream has both a smoothed rate shape and a consistent picture quality.

6.1 Source Modeling in A Brief Review

There are two basic approaches for source modeling. The first is the analytic approach. Its objective is to derive a set of mathematical formulas for the R-D functions based on the statistical properties of the source data. In this approach, both the coding system and the image are first decomposed into components whose statistical models are already known. These models are then combined to form a complete analytic model for the whole coding system. The R-D functions for a simple quantizer have been developed for a long time [20, 21]. In the analytic source model proposed by Hang and Chen [10], a theoretical entropy formula for the quantized DCT coefficients is developed based on the R-D theory of the Gaussian source and the uniform quantizer. The mismatches between the theoretical entropy and the actual coding bit rate of the entropy encoder is, however, compensated by empirical estimation.

The second approach is the empirical approach. Here, the R-D functions are con-

structed by mathematical processing of the observed R-D data. In the R-D estimation algorithm proposed by Lin and Ortega [11], eight control points on the R-D curves are first computed by running the coding system eight times. The whole R-D curves are then constructed by cubic interpolation. In the MPEG rate control algorithm proposed by Ding and Liu [12], the R-D curves are fitted by mathematical functions with several control parameters which are estimated from the observed R-D data of the coding system. In general, this type of R-D estimation algorithms have very high complexity. In addition, such algorithms do not provide us with insights into the R-D behaviors of the transform coding systems [22].

It is well known that the coding performance of a coding systems is determined by both the characteristics of the input source data and the ability of coding algorithm to explore these characteristics. How to characterize the input source data and how to model the coding algorithm with low complexity still remain challenging. Here, by "low complexity" we mean the overall complexity of the R-D curves estimation should be relative small compared to the complexity of the whole coding algorithm. In other words, the R-D curves estimation should take much less time than the actual coding process. In this sense, obviously, the empirical approach has high computational complexity since it needs to run the coding process several times. In this chapter, based on the ρ-domain analysis presented in Chapter 1, we develop a very efficient source modeling framework which can accurately estimate the R-D curves before quantization and coding with very low computational complexity and implementation cost.

6.2 A Unified Source Modeling Framework

In Fourier analysis, which is a powerful tool for digital signal processing, to study the behavior of a function $f(x)$, we first represent $f(x)$ by a linear combination of the basis

functions $\{\sin(nx), \cos(nx)\}$ which have well-known properties, as follows,

$$f(x) = a_0 + \sum_{n=1}^{\infty} a_n \cos(nx) + b_n \sin(nx), \tag{6.1}$$

where $\{a_n, b_n\}$ are called Fourier coefficients. By studying these Fourier coefficients, we can then analyze the behavior of $f(x)$. This method is referred to as *signal decomposition and spectrum analysis*. In this chapter, we apply this methodology to analyze and estimate the rate function of an image/video encoder.

In the previous chapters, we have shown the great advantage of the ρ-domain analysis. When a rate function in the q-domain is mapped into the ρ-domain, the image-dependent variation and highly nonlinear behavior of the rate function is significantly removed. Therefore, in this chapter we still analyze and estimate the rate function in the ρ-domain. In Chapter 3, both experimentally and theoretically, we show that, for typical transform coding systems, the rate function in the ρ-domain, denoted as $R(\rho)$, is approximately linear. Although the slope of $R(\rho)$ remains unknown, the linearity itself provides very useful information for the rate estimation carried out in this chapter. Based on this observation, our rate estimation has two major steps: In the first step, we employ the decomposition and analysis methodology described above to estimate the rate function $R(\rho)$ in the ρ-domain. In the second step, we utilize the linearity constraint to improve the estimation accuracy and robustness.

To estimate $R(\rho)$ using the decomposition scheme, we first define two basis functions $Q_{nz}(\rho)$ and $Q_z(\rho)$, called characteristic rate curves, to characterize the input source data. Here, the source data can be a still image or a video frame. We then show that, in the ρ-domain, $Q_{nz}(\rho)$ and $Q_n(\rho)$ have unique behaviors that enable us to estimate them with very low computational complexity. In our decomposition scheme, the actual rate function is represented by a linear combination of $Q_{nz}(\rho)$ and $Q_n(\rho)$,

$$R(\rho) = A(\rho) \cdot Q_{nz}(\rho) + B(\rho) \cdot Q_z(\rho) + C(\rho), \tag{6.2}$$

where $A(\rho)$, $B(\rho)$ and $C(\rho)$ are the rate decomposition coefficients. For a given input image, $Q_{nz}(\rho)$ and $Q_z(\rho)$ are fixed. If we use different coding algorithms to encode this image, we should obtain different $R(\rho)$. According to Eq. (6.2), we know the corresponding decomposition coefficients should also be different. In other words, different coding algorithms correspond to different decomposition coefficients. Therefore, we can say that $\{A(\rho),\ B(\rho),\ C(\rho)\}$ model the coding algorithm, while $\{Q_{nz}(\rho),\ Q_z(\rho)\}$ characterize the input source data. As mentioned above, the R-D performance of a coding system is determined by these two components. We see that both of these components have been integrated by linear combination into Eq. (6.2), which serves as the framework for our ρ-domain source modeling. In the following section, we define $Q_{nz}(\rho)$ and $Q_z(\rho)$, analyze their properties, and discuss the rate decomposition scheme in detail.

6.3 Characteristic Rate Curves

We first define the characteristic rate curves $Q_{nz}(\rho)$ and $Q_z(\rho)$. Based on our extensive simulation results, we then discuss their statistical properties. With these properties, a fast algorithm is proposed to estimate these two rate curves.

6.3.1 Definition

The characteristic rate curves $Q_{nz}(\rho)$ and $Q_z(\rho)$ are employed to characterize the transform coefficients to be quantized and coded by the image/video encoder. Their definitions are based on the following two observations: First, as mentioned in Section 1.4, zeros play a key role in transform coding of images and videos. Therefore, it is necessary and beneficial to model these zeros separately. In other words, we need to classify the transform coefficients into two parts—zero and non-zero coefficients— and model them separately. Consequently, we need to introduce two characteristic rate

curves $Q_z(\rho)$ and $Q_{nz}(\rho)$ to characterize the zero and non-zero coefficients, respectively. Second, we observe that, in any typical transform coding systems, coding the non-zero coefficients is comparable to the binary representation, and coding the zeros is comparable to the binary representation of their run-length. For example, in EZW and SPIHT coding the coefficients are coded bit-plane by bit-plane [16, 18]. In stack-run, explicit binary representation is employed [19]. In JPEG, for each non-zero coefficient, the length of its binary representation is jointly coded with its preceding zeros while its residue bits are sent out directly to the decoder [2]. In MPEG and H.263, the non-zero coefficients are Huffman coded. Note that, in their Huffman tables, the length of the codeword increases with the size of the non-zero coefficient [6, 9]. For the zero coefficients, in JPEG, MPEG and H.263, run-length coding of zeros is directly employed. Therefore, we believe that the binary representation of the non-zero coefficients and the zeros run-length numbers will give us the most valuable information about their coding characteristics in different coding systems. Accordingly, our definitions of the two characteristic rate curves are based on the binary representation scheme. The expressions $Q_z(\rho)$ and $Q_{nz}(\rho)$ are defined by the following pseudo coding process.

Step 1. *Conversion to 1-D array.* After transform and quantization with a quantization parameter q, the transform coefficients are rearranged into a 1-D array $\mathcal{L}$. If the wavelet transform is used, the subband coefficients are rearranged into $\mathcal{L}$ in a raster scan order. If DCT is used, all the DCT coefficients are rearranged into $\mathcal{L}$ in a zig-zag scan order inside each block and a blockwise raster scan order at the block level.

Step 2. *Pseudo coding.* For any non-zero number x, its size $S(x)$ is defined as

$$S(x) = \lfloor \log_2 |x| \rfloor + 2, \tag{6.3}$$

which is exactly the number of bits for its sign-magnitude representation. Note that, according to the above definition, $S(\pm 1)$ is 2 instead of 1. For each continuous string of zeros in $\mathcal{L}$, we count their run length. Let Q'_z be the sum of the sizes of all the run length numbers. For all the non-zero transform coefficients in $\mathcal{L}$, we define

$$Q'_{nz} = \sum_{x \in \mathcal{L},\, x \neq 0} S(x), \tag{6.4}$$

which is the sum of their sizes. Let

$$Q_{nz} = \frac{1}{M} Q'_{nz}, \quad Q_z = \frac{1}{M} Q'_z, \tag{6.5}$$

where M is the number of coefficients inside the picture. Respectively, Q_{nz} and Q_z can be regarded as the pseudo coding bit rate for the non-zeros and zero coefficients. Obviously, they are functions of q. Let ρ be the percentage of zeros among the quantized transform coefficients. From Chapter 1, we know there is a one-to-one mapping between q and ρ. Therefore, mathematically, Q_{nz} and Q_z are also functions of ρ, denoted by $Q_{nz}(\rho)$ and $Q_z(\rho)$, respectively. These two functions are called the characteristic rate curves.

We would like to point out one implementation detail about the above definition. As we see from Section 2.3.1, in MPEG-2, H.263 and MPEG-4 video coding, the DC coefficients from the intracoded macroblocks are quantized with a fixed quantization parameter 8 and encoded with a fixed number of bits which is also 8. This implies that the coding bit rate of these DC coefficients is fixed and does not depend on the quantization parameter. Therefore, when we scan the picture to form the 1-D array, the DC coefficients from the intracoded macroblocks are all skipped. Their coding bit rate is compensated by our rate decomposition scheme.

6.3.2 Statistical Properties

Next we show that in the ρ-domain, the two characteristic rate curves $Q_{nz}(\rho)$ and $Q_z(\rho)$ have unique properties. For each sample image in Fig. 3.1, we first decompose it with a 9-7 Debauchies wavelet [52, 13]. According to their definitions, we generate the rate curves $Q_{nz}(\rho)$ and $Q_z(\rho)$ and plot them in Fig. 6.1. Two observations can be made from these plots: First, although the sample images are quite different from each other, their characteristic rate curves have almost the same pattern. The second observation is that Q_{nz} is very close to a straight line. Note that when ρ is 1.0 which means all the coefficients are quantized to zeros, by definition $Q_{nz} = 0$. That is to say, $Q_{nz}(\rho)$ must pass through the point $[1.0, 0.0]$. Hence, it has the following expression,

$$Q_{nz}(\rho) = \kappa \cdot (1 - \rho), \tag{6.6}$$

where κ is a constant. It can be seen that $Q_{nz}(\rho)$ has a very simple behavior. Besides this, for each sample image, $Q_z(\rho)$ also has a rather simple behavior. In Fig. 6.2, we plot these two characteristic rate curves in the q-domain. It can be seen that in the q-domain, they have large image-dependent variations and highly nonlinear behaviors.

The unique behaviors of $Q_{nz}(\rho)$ and $Q_z(\rho)$ exist not only for wavelet coding, but also for the DCT-based image coding. For each sample image in Fig. 3.1, according to their definitions, we generate $Q_{nz}(\rho)$ and $Q_z(\rho)$ with DCT and JPEG-style quantization, and plot them in Fig. 6.3. The same phenomenon as in the above is observed. Next, we show that unique properties of $Q_{nz}(\rho)$ and $Q_z(\rho)$ exist not only for still images, but also for motion compensated pictures which are the major type of source data in video coding. Let's take two QCIF video sequences "Foreman" and "Salesman" as examples. First, we run the H.263 coder on these two videos at a fixed quantization parameter 8. For each video, we output 30 sample motion-compensated difference pictures. Each sample picture is taken at every ninth frame. Note that the first one is an I-frame without motion compensation; all of the others are P-frames. For each sample picture,

according to their definitions, we generate the two rate curves $Q_{nz}(\rho)$ and $Q_z(\rho)$ with DCT and H.263 style quantization. These two curves for "Foreman" and "Salesman" are plotted in Figs. 6.4 and 6.5, respectively. We can see that the unique properties of $Q_{nz}(\rho)$ and $Q_z(\rho)$ also exist in transform coding of videos.

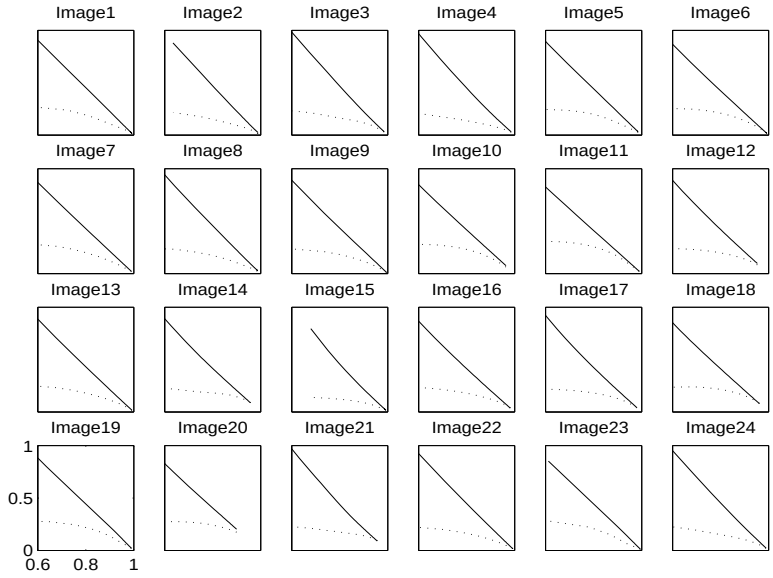

Figure 6.1: Plots of $Q_{nz}(\rho)$ (solid) and $Q_z(\rho)$ (dotted) for the 24 sample images with wavelet transform and uniform threshold quantization. The x-axis represents the percentage of zeros ρ while the y-axis represents the pseudo coding bit rate. All the plots have the same coordinate system.

6.3.3 Justification of the Linearity of $Q_{nz}(\rho)$

The definition of $Q_{nz}(\rho)$ is based on the pseudo coding of the non-zero transform coefficients, which is actually the sign-magnitude representation given by Eq. (6.3). From the simulation results given above, we observe that it has a very interesting linear behavior. In the following, we provide a theoretical justification for the linearity of $Q_{nz}(\rho)$. As discussed in Section 2.3.1, the quantization schemes employed in the typical transform coding systems are all essentially uniform threshold quantizers. Therefore, in the following, we take the uniform threshold quantizer as an example to show the

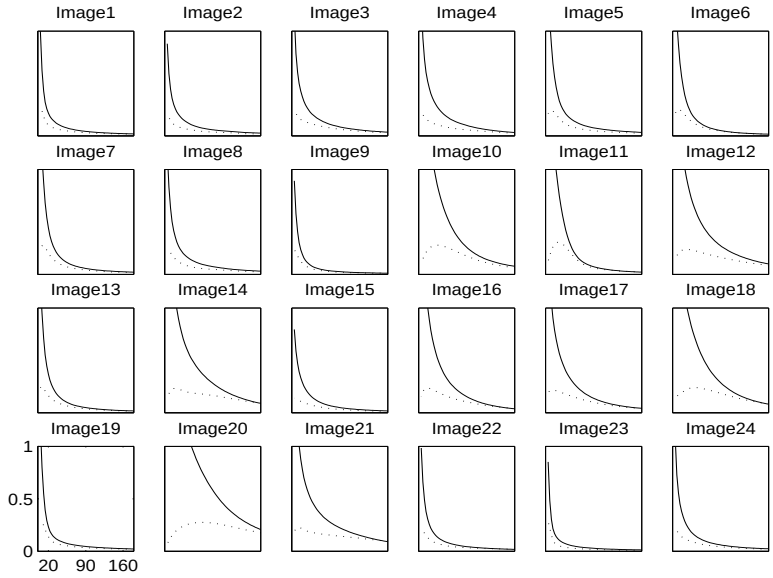

Figure 6.2: Plots of $Q_{nz}(q)$ (solid) and $Q_z(q)$ (dotted) for the 24 sample images with wavelet transform and uniform threshold quantization. The x-axis represents the quantization parameter q while the y-axis represents the pseudo coding bit rate. All the plots have the same coordinate system.

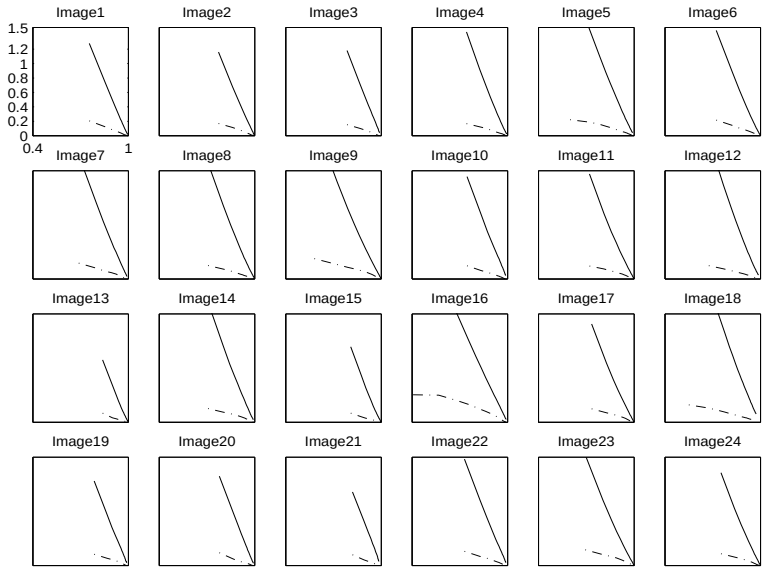

Figure 6.3: Plots of $Q_{nz}(\rho)$ (solid) and $Q_z(\rho)$ (dotted) for the 24 sample images with DCT and JPEG quantization. The x-axis represents the quantization parameter q while the y-axis represents the pseudo coding bit rate. All the plots have the same coordinate system.

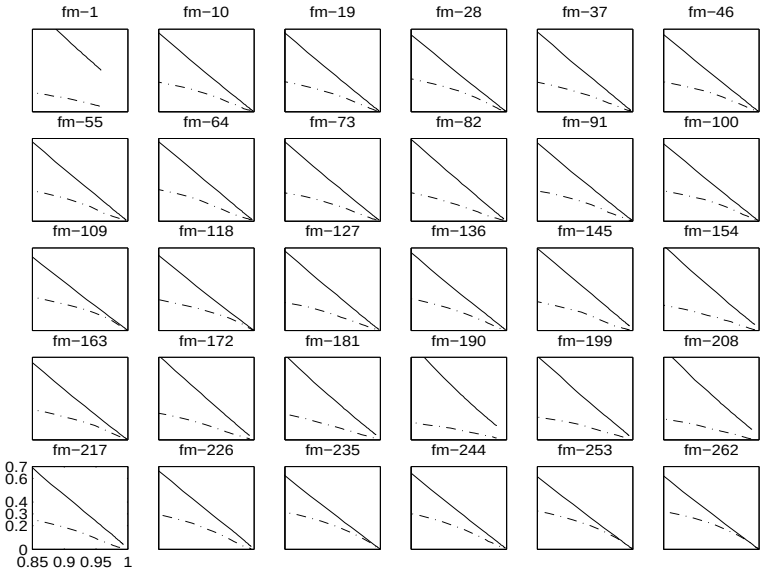

Figure 6.4: The plots of $Q_{nz}(\rho)$ (solid line) and $Q_z(\rho)$ (dash-dot line) for the 30 sample difference pictures from Foreman. The x-axis represents the percentage of zeros ρ. All the subplots have the same coordinate system as the one at the bottom-left corner.

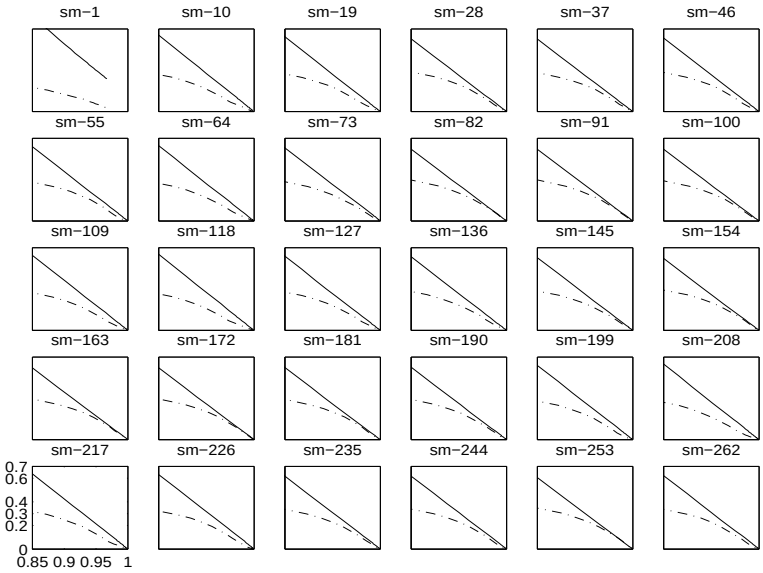

Figure 6.5: The plots of $Q_{nz}(\rho)$ (solid line) and $Q_z(\rho)$ (dash-dot line) for the 30 sample difference pictures from Salesman. The x-axis represents the percentage of zeros ρ. All the subplots have the same coordinate system as the one at the bottom-left corner.

linearity of $Q_{nz}(\rho)$. It is well known that the transform coefficients have a generalized Gaussian distribution given by Eq. (3.4). According to Eq. (6.4), for any given quantization step size q, we have

$$Q_{nz}(q) = \frac{2}{M} \cdot \int_{\Delta}^{+\infty} p_{gg}(x)(\lfloor \log_2 |I(x)| \rfloor + 2)\, dx \tag{6.7}$$

where $I(x)$

$$I(x) = \lceil \frac{x - \Delta}{q} \rceil \tag{6.8}$$

is the quantization index of x. The corresponding percentage of zeros is given by

$$\rho(q) = \frac{1}{M} \int_{-\Delta}^{+\Delta} p_{gg}(x)\, dx. \tag{6.9}$$

It is very difficult to derive a close-form expression for $Q_{nz}(\rho)$ directly from Eqs. (6.7) and (6.9). However, we can evaluate them numerically and compute a few points on $Q_{nz}(\rho)$. In Fig. 6.6, we plot them for different ν which is the shape control parameter of $p_{gg}(x)$. It can be seen that these points almost lie on a straight line. This implies that, if we assume the transform coefficients have a generalized Gaussian distribution, $Q_{nz}(\rho)$ must be an approximately linear function.

6.3.4 Fast Estimation of $Q_{nz}(\rho)$

Since $Q_{nz}(\rho)$ is modeled as a straight line passing through the point $[1.0, 0.0]$, we need to compute only one point on it in order to estimate the whole rate function. In the following, we discuss the estimation procedure in detail for different transform coding systems.

As mentioned in Section 2.3.1, typical wavelet-based image coding, such SPIHT, EZW, and stack-run employ the uniform threshold quantizer. After transform, we scan the subband image and generate the distribution of the transform coefficients, denoted by $\mathcal{D}(x)$. We then choose one quantization parameter q_0, and compute the corresponding value of $Q_{nz}(q_0)$ and $\rho(q_0)$ using Eqs. (6.7) and (6.9) with $p_{gg}(x)$ replaced

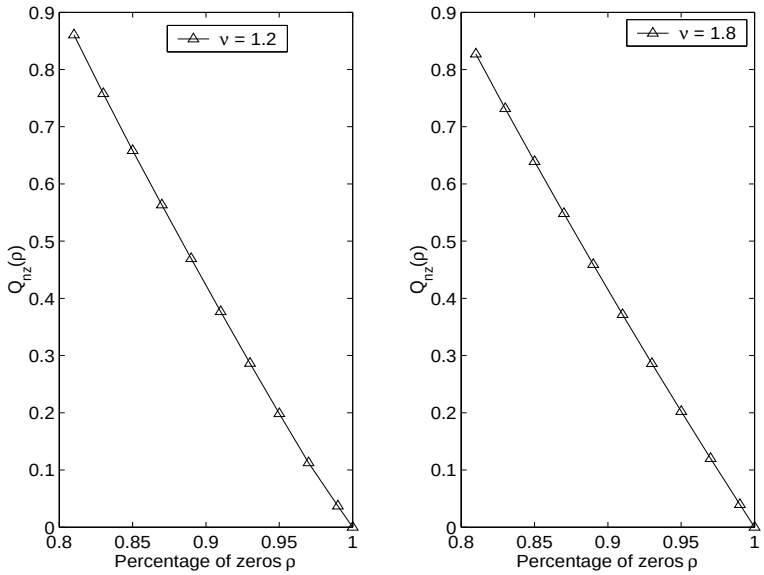

Figure 6.6: Plots of the theoretically computed $Q_{nz}(\rho)$ for the generalized Gaussian distribution with different shape control parameters ν. Here, we set $\Delta = q$.

by the actual distribution $\mathcal{D}(x)$. With the two points of $[Q_{nz}(q_0), \rho(q_0)]$ and $[1.0, 0.0]$, we can construct the whole rate curve $Q_{nz}(\rho)$ with Eq. (6.6) where

$$\kappa = \frac{Q_{nz}(q_0)}{1 - \rho(q_0)}. \tag{6.10}$$

For H.263 video coding, from Eq. (2.6) we know the intracoded and intercoded macroblocks have different quantization schemes. Let the distributions of the DCT coefficients in the intracoded and intercoded macroblocks be $\mathcal{D}_0(x)$ and $\mathcal{D}_1(x)$, respectively. (As pointed out in Section 6.3.1, $\mathcal{D}_0(x)$ should not include the DC coefficients from the intracoded macroblocks.) With Eqs. (2.5), (2.6) and (6.4), we have

$$Q_{nz} = \sum_{|x| \geq 2q} S(\mathrm{UTQ}[2q, 2q; x]) \cdot \mathcal{D}_0(x) + \sum_{|x| \geq 2.5q} S(\mathrm{UTQ}[2q, 2.5q; x]) \cdot \mathcal{D}_1(x) \tag{6.11}$$

We can then compute the slope κ with Eq. (6.10) and estimate the whole rate function $Q_{nz}(\rho)$. For JPEG or MPEG coding, we can first convert its perceptual quantization scheme into a uniform or H.263-style quantization scheme with prescaling as discussed in Section 2.3.2, and then follow the same procedure to estimate $Q_{nz}(\rho)$.

6.3.5 Fast Estimation of $Q_z(\rho)$

In Section 6.3.4, we developed a fast algorithm to estimate the rate function $Q_{nz}(\rho)$. In the following, by exploring the correlation between $Q_z(\rho)$ and $Q_{nz}(\rho)$, we develop a fast estimation scheme for $Q_z(\rho)$. To study the correlation between two curves, we first define feature variables for each curve, then study the correlation between these feature variables. The feature variable for $Q_{nz}(\rho)$ is its slope κ. The feature variables for $Q_z(\rho)$ are its function values at $\rho_i = 0.70$, 0.75, 0.80, 0.85, 0.90, and 0.95. Consider the characteristic rate curves plotted in Fig. 6.1. For each ρ_i, $1 \leq i \leq 6$, and for each of the 24 sample images, we know κ which is the slope of $Q_{nz}(\rho)$, and $Q_z(\rho_i)$ which is the value of $Q_z(\rho)$ at ρ_i. Therefore, for each ρ_i, we have a total of 24 points of $\{[\kappa, Q_z(\rho_i)]\}$ which are depicted in Fig. 6.7.

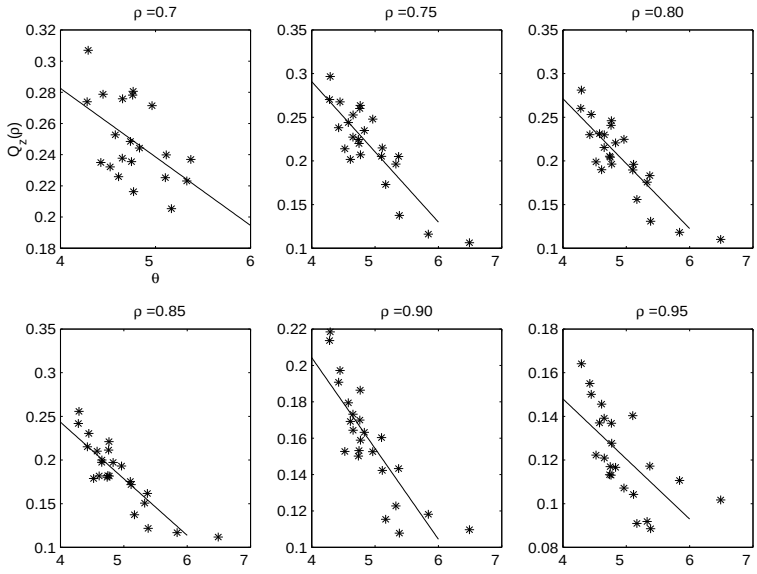

Figure 6.7: The linear correlation between κ and the values $Q_z(\rho_i)$ at 0.70, 0.75, 0.80, 0.85, 0.90 and 0.95 with wavelet transform.

Fig. 6.8 illustrates the correlation between κ and $Q_z(\rho_i)$ for the characteristic rate curves in Fig. 6.3. Note that there is a strong correlation between κ and $Q_z(\rho_i)$. In our extensive simulation with a wide range of images, this strong correlation has been

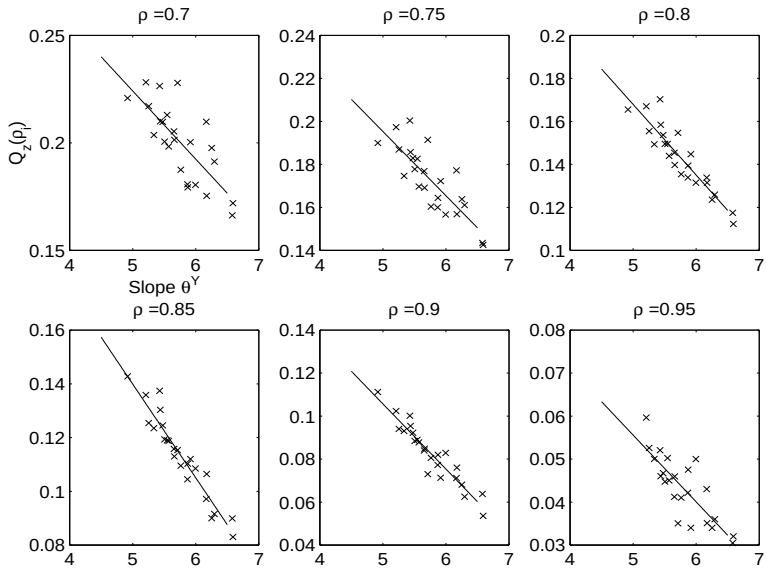

Figure 6.8: The linear correlation between κ and the values $Q_z(\rho_i)$ at 0.70, 0.75, 0.80, 0.85, 0.90 and 0.95 with DCT.

found to exist. Therefore, we have the following linear correlation model,

$$Q_z(\rho_i) = A_i\theta + B_i \tag{6.12}$$

which can be employed to estimate $Q_z(\rho_i)$. The coefficients A_i and B_i are obtained by statistical regression and the corresponding linear model is also plotted in Figs. 6.7 and 6.8. Based on Eq. (6.12) , we can estimate six points on $Q_z(\rho)$. If necessary, the whole rate curve can be constructed by linear interpolation.

6.4 Rate Curve Decomposition

In the previous section, we defined two rate curves, $Q_{nz}(\rho)$ and $Q_z(\rho)$, to characterize the input source data. According to our decomposition and analysis scheme (presented in Section 6.2), the actual rate curve in the ρ-domain $R(\rho)$ is represented by a linear combination of $Q_{nz}(\rho)$ and $Q_z(\rho)$, as shown in Eq. (6.2), where the coding algorithm is modeled by the decomposition coefficients $\{A(\rho),\ B(\rho),\ C(\rho)\}$. In the following,

we take the JPEG coding algorithm as an example to explain how to determine the decomposition coefficients $\{A(\rho),\ B(\rho),\ C(\rho)\}$ for a specific coding algorithm.

6.4.1 Decomposition Coefficients

For the 24 sample images shown in Fig. 3.1, with the fast algorithms developed in the previous section, we can estimate the values of $Q_{nz}(\rho_i)$ and $Q_z(\rho_i)$ where $\rho_i = 0.65 + i \cdot 0.05,\ 1 \leq i \leq 6$. By running the JPEG coding algorithm at different quantization parameters, we can generate some points on its actual rate curve $R(\rho)$. With interpolation, we can obtain $\{R(\rho_i)|1 \leq i \leq 6\}$. According to Eq. (6.2), we should have

$$R(\rho_i) = A(\rho_i) \cdot Q_{nz}(\rho_i) + B(\rho_i) \cdot Q_z(\rho_i) + C(\rho_i). \tag{6.13}$$

The values of $\{A(\rho_i),\ B(\rho_i),\ C(\rho_i)\}$ are obtained by linear regression over the sample data $\{Q_{nz}(\rho_i), Q_z(\rho_i), R(\rho_i)\}$ for the 24 sample images. The decomposition coefficients for the JPEG coding algorithm are listed in Table 6.1. Following the same procedure, we can also obtain the decomposition coefficients for other coding algorithms, such as EZW, SPIHT, stack-run, MPEG, and H.263. Once they are obtained, they are fixed during the actual rate curve estimation process.

6.4.2 Linear Regulation

With Eq. (6.13), six points on the actual rate curve $R(\rho)$ are estimated. From Chapter 3 we know that in any typical transform coding systems, $R(\rho)$ is a linear function given by Eq. (3.1). Therefore, theoretically, $\{R(\rho_i)|1 \leq i \leq 6\}$ should lie on a straight line. But, because of the source modeling error, this is often not the case. However, we can utilize the linear constraint to significantly remove the modeling error. To be more specific, from $\{R(\rho_i)|1 \leq i \leq 6\}$ we first estimate the slope θ and then construct $R(\rho)$ using the linear rate model in Eq. (3.1). According to the linear regression theorem,

Table 6.1: The values of $A(\rho)$, $B(\rho)$ and $C(\rho)$ at ρ_i for the JPEG coding algorithm.

ρ_i	$A(\rho_i)$	$B(\rho_i)$	$C(\rho_i)$
0.70	1.2151	-0.4438	0.9005
0.75	0.8089	-0.5030	0.9201
0.80	0.6480	-0.3831	0.8449
0.85	0.5763	-0.3449	0.7856
0.90	0.5531	-0.2241	0.6808
0.95	0.4043	-0.1489	0.5845

the optimum estimation of θ is given by

$$\theta = \frac{\sum_{i=1}^{6} \rho_i \sum_{i=1}^{6} R(\rho_i) - 6 \sum_{i=1}^{6} \rho_i R(\rho_i)}{6 \sum_{i=1}^{6} \rho_i^2 - (\sum_{i=1}^{6} \rho_i)^2} \tag{6.14}$$

According to the linear rate model, the estimated rate curve $R(\rho)$ after linear regulation is then given by

$$R(\rho) = \frac{1}{6} \sum_{i=1}^{6} R(\rho_i) + \theta \cdot (\rho - \frac{1}{6} \sum_{i=1}^{6} \rho_i). \tag{6.15}$$

6.5 A Unified R-D Curve Estimation Algorithm

Based on the fast estimation of $Q_{nz}(\rho)$ and $Q_z(\rho)$ and the rate curve decomposition scheme, a unified R-D curve estimation algorithm for all typical transform coding systems is proposed as follows:

Step 1. *Generation of the distribution.* After transform, either DWT or DCT, generate the distribution of the transform coefficients. Note that the transform coefficients are real numbers. We can approximate their distribution by the histogram of their integer parts. In the implementation of standard video coding such as

MPEG and H.263, the DCT coefficients automatically have integer values. In this case, no approximation is needed. Depending on the specific quantization scheme, the distribution generation varies. For example, in H.263, as discussed in Section 6.3.4, the distributions of the intracoded and intercoded macroblocks need to be stored separately. In JPEG and MPEG coding, we need to generate the distribution after prescaling of the DCT coefficients. Based on the distributions of the transform coefficients, the one-to-one mapping lookup table between q and ρ is obtained as discussed in Section 2.3.2.

Step 2. *Estimate $Q_{nz}(\rho)$ and $Q_z(\rho)$.* Choose one quantization parameter q_0 and compute the corresponding $Q_{nz}(q_0)$ and $\rho(q_0)$ as discussed in Section 6.3.4. The slope of $Q_{nz}(\rho)$ is obtained by Eq. (6.10). The value of $Q_{nz}(\rho)$ at ρ_i is given by

$$Q_{nz}(\rho_i) = \kappa(1 - \rho_i). \tag{6.16}$$

With the linear correlation model in Eq. (6.12), the value of $Q_z(\rho_i)$ is determined.

Step 3. *Rate Curve Estimation.* With Eq. (6.13), compute $R(\rho_i)$. After linear regulation, the estimated rate curve $R(\rho)$ is given by Eq. (6.15). With the q-ρ mapping lookup table, $R(\rho)$ is mapped into the q-domain to obtain $R(q)$.

Step 4. *Compute distortion curve.* The D-Q curve $D(q)$ can be directly computed from the distribution as discussed in Section 5.1.

We can see that, in the proposed algorithm, the major computation is to generate the distribution of the transform coefficients. The rest of the computation involves only addition and multiplication operations that are carried out on the distribution. Compared to the complexity of the whole coding process, the overall complexity of the proposed estimation algorithm is very low. In addition, the R-D curves are estimated before quantization and coding. The estimation accuracy is evaluated in the following sections.

6.6 Application in Still Image Coding

We apply the algorithm presented in Section 6.5 to estimate the R-D curves for transform coding of still images.

6.6.1 Wavelet-Based Image Coding

The proposed estimation algorithm has been applied to the SPIHT and the stack-run (SR) encoders. We arbitrarily choose six test images as shown in Fig. 6.9. The estimated R-D curve and the actual one for SR and SPIHT coding are shown in Figs. 6.10 and 6.11, respectively. It can be seen that the estimated R-D curves are very close to the actual ones curves.

Figure 6.9: The six test images for the evaluation of the proposed R-D estimation algorithm when applied to the SPIHT and stack-run encoders.

6.6.2 DCT-Based Image Coding

The R-D estimation algorithm proposed in Section 6.5 is also applied to JPEG coding of still images. The six test images with a wide range of R-D characteristics are shown

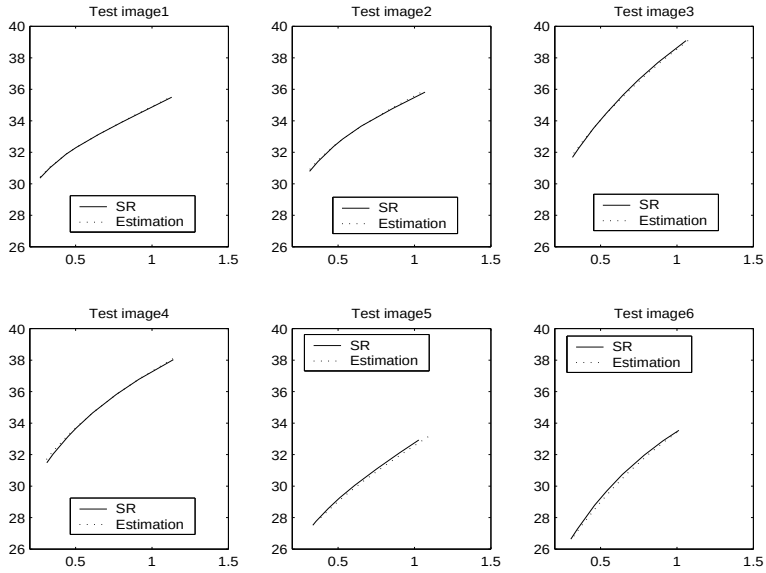

Figure 6.10: The R-D curve estimation results for the stack-run coding system.

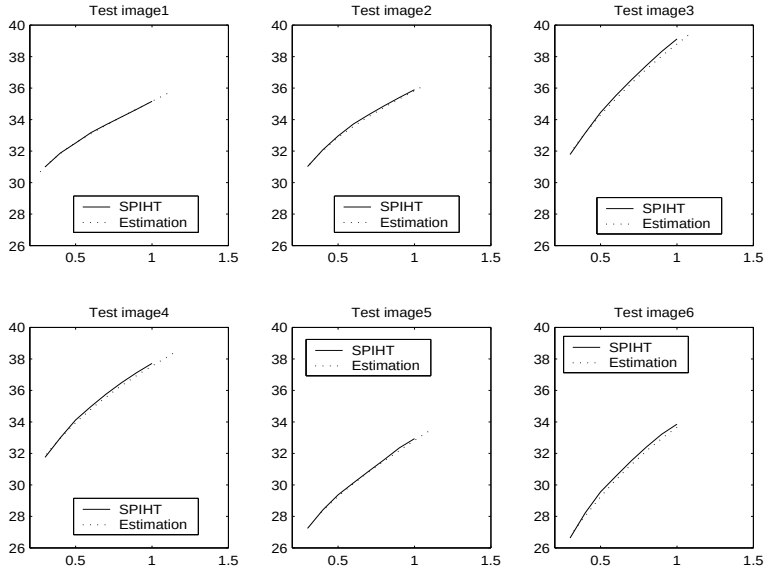

Figure 6.11: The R-D curve estimation results for the SPIHT coding system.

Table 6.2: The relative estimation error for JPEG coding of the test images shown in Fig. 6.12.

q_i	Rate estimation error $\mathcal{E}(\lambda_i)$ for testing image					
	No. 1	No. 2	No. 3	No. 4	No. 5	No. 6
0.5	/	-1.4%	+1.1%	+2.3%	/	+1.4%
0.8	1.4%	-2.2%	-0.3%	-0.1%	-0.1%	-0.1%
1.2	0.8%	-2.7%	-0.9%	-1.8%	0.4%	-1.3%
2.0	1.0%	-3.9%	-2.8%	-4.0%	-0.4%	-3.1%
2.8	2.7%	-3.7%	-2.9%	-3.5%	-0.7%	-2.8%
3.2	3.8%	-3.4%	-2.1%	-2.7%	-0.1%	-2.3%
4.5	3.5%	-3.8%	-4.3%	-3.6%	-1.2%	-1.8%
5.5	4.1%	-2.8%	-2.8%	-3.1%	-0.2%	-0.2%

in Fig. 6.12. We apply the proposed estimation algorithm to estimate their rate curves. The estimated rate curves and the actual curves are plotted in Fig. 6.13. The relative estimation errors of the rate curve $R(q)$ at $q_i = 0.5$, 0.8, 1.2, 2.0, 2.8, 3.2, 4.5 and 5.5 for each test image are listed in Table 6.2. The estimation errors are very small, mostly less than 3% in their absolute values.

6.7 Frame-Level Rate Control for Video Coding

The algorithm proposed in Section 6.5 is a unified R-D estimation algorithm for any typical transform coding systems. It can also be applied to estimate the rate or distortion curve, or both, for transform coding of video sequences. In the following, we apply the proposed algorithm to rate control for H.263 video coding.

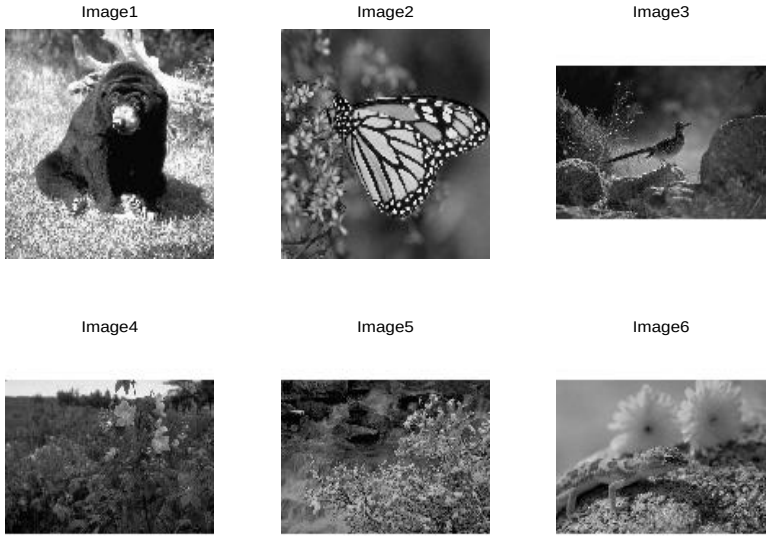

Figure 6.12: The six images for testing the performance of the proposed algorithm.

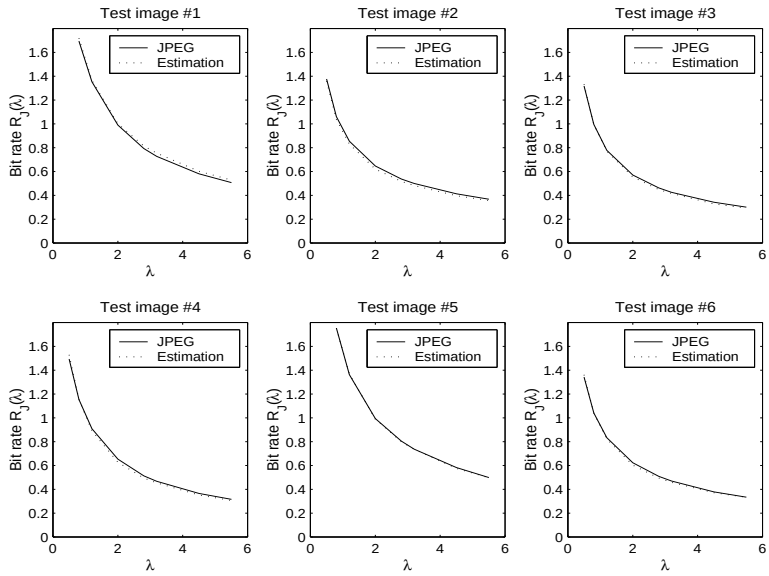

Figure 6.13: The estimated rate curves and the real JPEG rate curves of the six test images. The x-axis represents the quantization parameter q.

6.7.1 Frame-level Rate Control Algorithm

The rate control here consists of the following major steps: In the first step, as discussed in Section 4.2, the target bit rate R_T of the current video frame is determined according to the channel bandwidth and buffer status. In the second step, using the estimation algorithm in Section 6.5, we estimate the rate curve $R(q)$ for the current frame. Finally, based on $R(q)$, the frame quantization parameter QP_0 can be determined to achieve the target bit rate R_T. Obviously, QP_0 satisfies $R_T = R(QP_0)$. We see that this rate control algorithm operates at the frame level.

6.7.2 The Quantization Parameter of Each Macroblock

It should be noted that, in any typical video encoder, the quantization parameter should have an integer value between 1 and 31 [6, 8, 10]. However, the frame quantization parameter QP_0 obtained in the frame-level rate control algorithm is a real number. For example, QP_0 could be 5.30. If we round QP_0 to its nearest integer 5 and use it for the quantization parameter for each macroblock in the current frame, the actual coding bit R will be quite different from the target bit rate R_T, which is actually achieved by $QP_0 = 5.30$. In the following, we propose a very simple approach to solve this problem. Let

$$QP_+ = \lceil QP_0 \rceil, \quad QP_- = \lfloor QP_0 \rfloor, \tag{6.17}$$

which are the two closest integers to QP_0. Let $\gamma = QP_0 - QP_-$. Let ϕ be a random variable with a uniform distribution on [0, 1]. Each time when we determine the quantization parameter QP for a macroblock, we produce a sample value for ρ. If its sample value is less than γ, we set $QP = QP_+$. Otherwise, we set $QP = QP_-$. In this way, approximately $\gamma \cdot 100$ percent of macroblocks in the current video frame use the quantization parameter QP_+ while all the rest use QP_-. Consequently, the average frame quantization parameter is very close to QP_0. In this way, the actual coding

Table 6.3: Comparison of the number of frames skipped and average PSNR for TMN8 and the proposed algorithm in H.263. The unit of the channel rate is kbits per second.

Test Video	Channel Rate	Frame Skipped		PSNR(dB)		
		This Work	TMN8	This Work	TMN8	Gain
Foreman	64	0	1	31.65	31.60	+0.05
Foreman	48	0	2	30.40	30.28	+0.12
M.&D.	16	2	6	31.13	31.09	+0.04
Salesman	32	1	3	32.89	32.84	+0.05
Carphone	32	1	3	31.02	30.09	+0.03
Coastguard	32	0	3	27.54	27.37	+0.17
News	48	0	2	33.62	33.66	–0.04
Container	32	1	4	33.23	33.05	+0.18

bit rate will be very close to the target bit rate R_T. Let S_+ and S_- be the groups of macroblocks which use QP_+ and QP_-, respectively. The overall activity of S_+ is almost the same as that of S_- because the macroblocks from each set are chosen by a uniform random variable. This is another advantage of this approach.

6.7.3 Rate Control Results

The proposed rate control algorithm is implemented in the H.263 codec (Version 2.0) [53] and tested for various video sequences and applications. The frame rate is fixed at 10 fps. All the test video sequences are in the QCIF picture format. In each test described in the following, the name of the test video sequence, the channel rate C, and the frame rate F are indicated in the title of the respective plot. In the following experiments on rate control, we compare the proposed algorithm with the TMN8 rate control algorithm [23].

Buffer Regulation Performance

In Figs. 6.14(a) – (d), we plot the number of bits in the encoder buffer for each coded frame when the proposed rate control algorithm and the TMN8 algorithm are applied in the H.263 codec. The horizontal dash-dot line shows the buffer size B_T. This is the threshold for frame skipping. Note that the first frame in each video is an I-frame. It produces a lot of bits. Therefore, when the channel rate C is relatively small, the encoder skips several frames. It can be seen that the proposed frame-level rate control algorithm maintains a even steadier buffer level than does the macroblock-level TMN8 algorithm.

The proposed rate control algorithm estimates the rate curve for each frame and controls its coding bit rate totally independently from other frames. Therefore, it does not suffer from any performance degradation at scene changes. To show this, we first generate a combination video by concatenating the first 30 frames of the following five videos together: "Foreman", "Carphone", "Salesman", "Miss-America" and "Coast-guard". In this way, a scene change occurs between any two neighboring video clips. In Fig. 6.15, we plot the buffer fullness when the proposed rate control algorithm and the TMN8 algorithm are applied. Again, we see that the proposed algorithm works very well at scene changes.

PSNR Performance and Frame Skip

In Table 6.3, we compare the average PSNR and the number of the skipped frames for the proposed algorithm and the TMN8 algorithm. It should be noted that the "PSNR" here refers to the PSNR of the luminance component. For each frame skipped, we reconstruct it with a copy of its previous frame and then compute the corresponding PSNR value. From Table 6.3 we can see that the proposed algorithm has fewer skipped frames and slightly better PSNR performance.

Adaptivity to the Channel Bandwidth

In live video applications, the channel bandwidth varies over time. Normally, the video encoder can detect the available channel bandwidth from the feedback information [73]. To simulate the rate control for live video in this situation, we suppose the channel bandwidth is changing as plotted in Fig. 6.16 (top). The actual coding bit rates of News are also plotted in Fig. 6.16 (bottom) when the proposed algorithm and the TMN8 rate control algorithm are applied. With the proposed rate control algorithm, the encoder can better match the coding bit rate to the channel bandwidth.

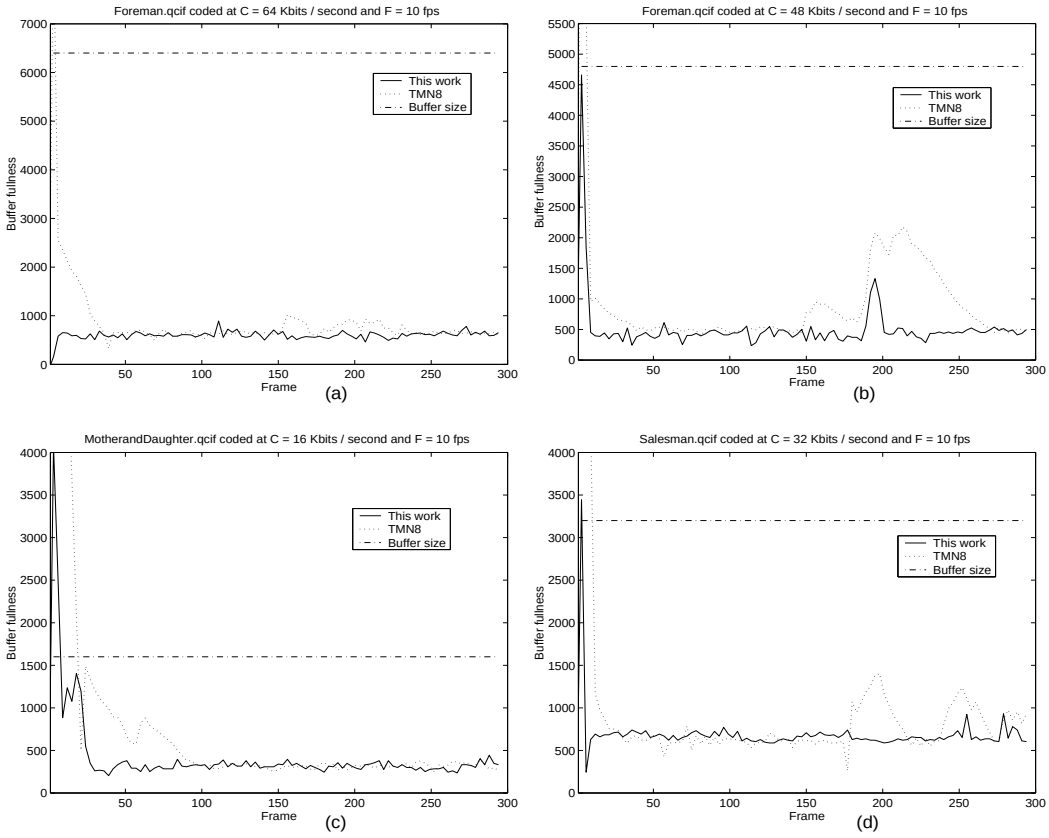

Figure 6.14: (a) – (d) Comparison of the number of bits in the encoder buffer when the proposed rate control algorithm (solid line) and the TMN8 (dashed line) are employed in the H.263 video coder. The name of respective video sequence, the channel rate C and the frame rate F are indicated in the title of each plot. The horizontal dash-dot line shows the buffer size B_T. The first frame is I-frame which needs more coding bits than the P-frames.

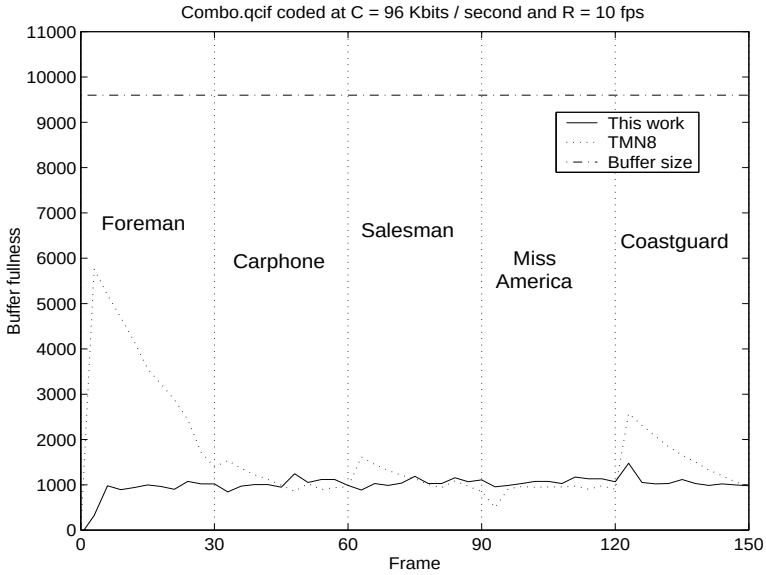

Figure 6.15: Comparison of the number of bits in the encoder buffer when the proposed rate control algorithm (solid line) and the TMN8 (dashed line) are applied to a video sequence with scene changes. The test video is formed by concatenating the following 5 QCIF videos together: Foreman, Carphone, Salesman, Miss-America and Coastguard.

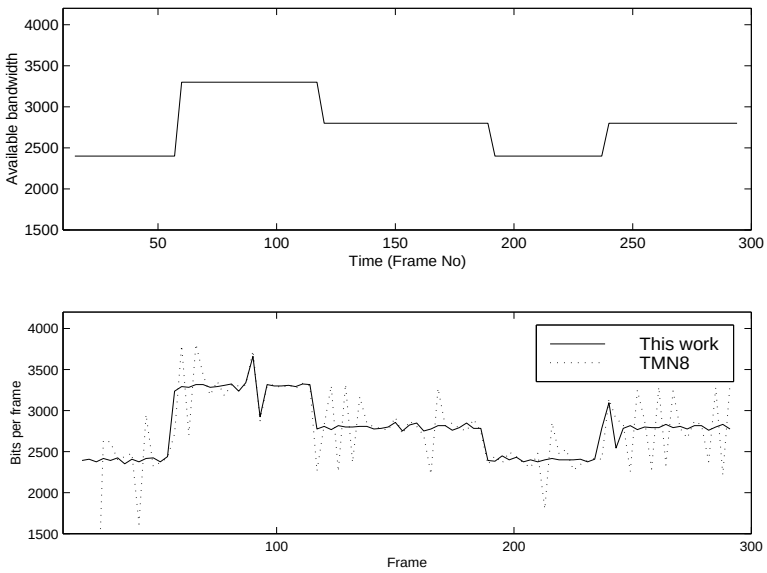

Figure 6.16: Comparison of the rate control performance with varying channel bandwidth for News.qcif at 24 kbps.

6.8 Compression-Based Rate Shape Smoothing

In video coding, the output bit rate of the encoder has a variable bit rate (VBR) due to the scene activities. For active videos, the output bits stream will be bursty with frequent and large rate fluctuations, which lead to large packet loss ratio and delay variation [74, 75] when the coded video data is transmitted. From the standpoint of transmission efficiency, network traffic management and resource allocation, VBR streaming video is much harder to handle than the video stream with a constant bit rate (CBR) [75, 76, 77]. However, CBR coding often has a large variation in picture quality. One way to smooth the frame bit rates without severe degradation of the picture quality is to introduce buffers on the packet delivery path [75]. The buffer operates like a lowpass filter. To achieve good smoothing effect, the buffer size should be relatively large, for example 10 to 30 frames. This obviously will increase the implementation and traffic management cost. In addition, large buffer size results in large playback start-up latency.

With the estimation algorithm proposed in Section 6.5, we can estimate the R-D curves for the input video frame. Based on the estimated R-D curves, we can control the quantization setting of the video encoder to achieve the target bit rate or picture quality. In the following, based on the R-D estimation algorithm, we develop an encoder-based rate shape smoothing algorithm. With this algorithm, the encoder can be controlled such that the output bit stream has both a smoothed rate shape and a consistent picture quality, which are highly desirable in practical video applications.

In real-time playback of coded video data, a constant presentation quality throughout the whole video is highly desirable. Let the target picture quality be D_T (in dB). Let the bit rate of each video frame be R_n where n is the frame number. Obviously, R_n varies over time or is even bursty. Note that in practice we do not have to keep the picture quality constant. Normally, the user cannot tell the difference between the

picture quality of D_T dB and $D_T \pm 1$ dB when the D_T is relative high, such as 35 or more. Based on this observation, we can let the picture quality of each frame, denoted by D_n, vary within a small range $[D_T - \delta, D_T + \delta]$ around the target quality D_T. Here δ is a small constant, such as 1 or 2. Obviously, this guarantees that no severe video quality degradation occurs over time. With this relaxation of the picture quality D_n, we can now smooth the rate shape $\{R_n\}$ of the video stream as follows:

Step 1. *Initialization.* When $n = 0$, based on the estimated R-D curves, determine the bit rate R_0 such that the corresponding picture quality D_0 is D_T. With the rate control algorithm in Section 6.7.1, the quantization parameter QP for each macroblock is determined to meet the target rate R_0.

Step 2. *Smoothing.* Suppose the current frame number is $n\,(n \geq 1)$. Based on the estimated R-D curves, determine the coding bit rates R_+ and R_- such that the corresponding picture quality D_n is $D_T+\delta$ and $D_T-\delta$, respectively. Obviously, we have $R_+ > R_-$. If $R_{n-1} \leq R_-$, we set $R_n = R_-$. If $R_{n-1} \geq R_+$, we set $R_n = R_+$. Otherwise, we set $R_n = R_{n-1}$. This implies that we are trying to make R_n as close to R_{n-1} as possible under the constraint $D_T - \delta \leq D_n \leq D_T + \delta$. With the rate control algorithm in Section 6.7.1, the quantization parameter QP for each macroblock is determined to meet the target rate R_n.

This rate smoothing algorithm imposes very little additional complexity to the proposed rate control algorithm. To demonstrate its efficiency, we perform the following experiment: The test video is Foreman in the QCIF format. If we set the target picture quality for each frame as $D_T = 34$ dB, the corresponding bit rate for each frame R_n is plotted in Fig. 6.17. Note that the rate varies dramatically. After we apply the above rate shape smoothing algorithm, the smoothed rate shape is shown in Fig. 6.17 in solid line. It can be seen that after smoothing, the burtiness of the rate is significantly reduced. The corresponding picture quality of each frame D_n is plotted in Fig. 6.18.

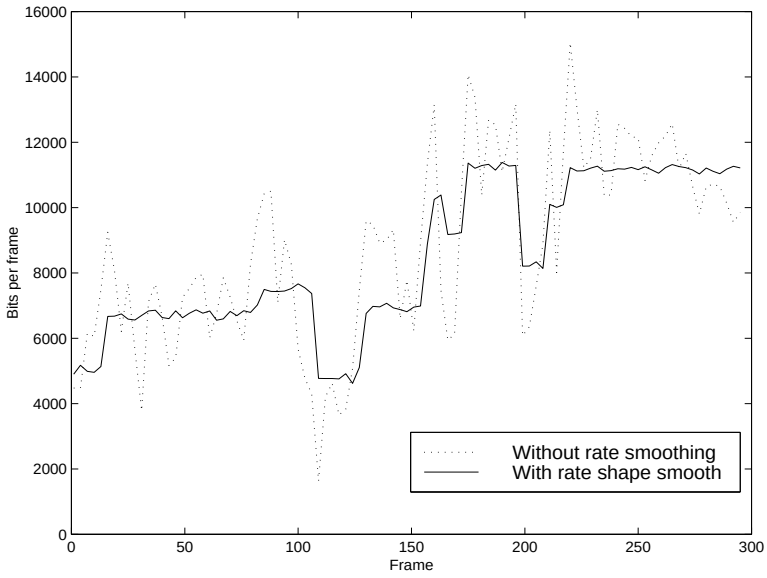

Figure 6.17: The rate shapes with and without smoothing for Foreman.qcif when the picture quality variation range is 1 dB.

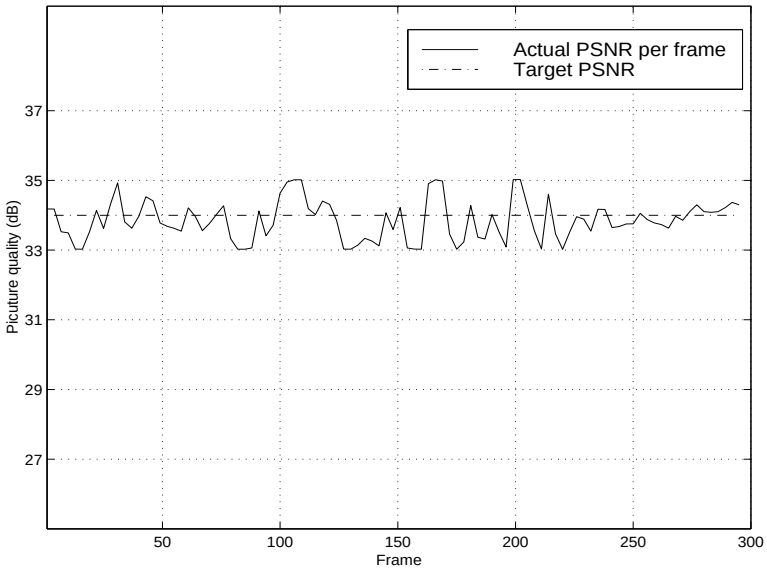

Figure 6.18: The video quality after rate shape smoothing for Foreman.qcif.

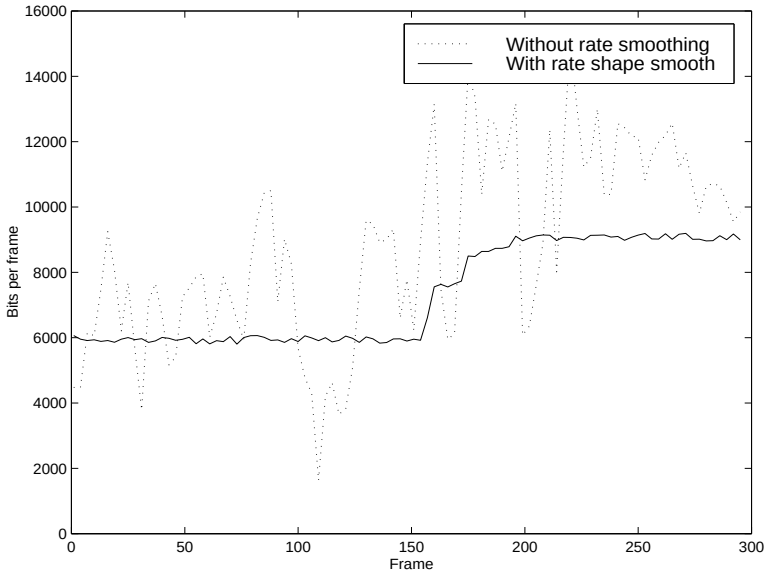

Figure 6.19: The rate shapes with and without smoothing for Foreman.qcif when the picture quality variation range is 2 dB.

It can be seen that D_n always falls in a small decibel range [33, 35], which implies the video quality is consistent over the playback time. If we change the value of δ from 1 dB to 2 dB, then rate shape is even smoother, as shown in Fig. 6.19. Our simulation results show that with the proposed smoothing algorithm, the output bit stream of the video encoder has both a smoothed rate shape and a consistent picture quality.

6.9 Summary

By introducing the concepts of characteristic rate curve and rate curve decomposition, a unified framework for source modeling has been developed in this chapter. With this framework, the R-D curves have been estimated before quantization and coding for any typical transform coding of still images. The relative estimation error is less than 5%. In addition, it has very low computational complexity. The source modeling framework has also been applied to rate control for video coding. Because of its more accurate source model, the proposed frame-level rate control algorithm is able to control the

output bit rate of the video encoder according to the channel condition accurately and robustly. Based on the estimated R-D curves, an encoder-based rate shape smoothing algorithm has been developed. With this smoothing algorithm, an optimum trade-off is made between the VBR and CBR video coding. The output bit stream of the encoder has both a smoothed rate shape and a consistent video quality, which is highly desirable in practical video coding and communications.

Chapter 7

Conclusions

In this chapter, we summarize the principal contributions of this dissertation. The proposed ρ-domain R-D analysis framework has more potential applications in practical video applications, such as video transcoding, wireless video, and video transmission over Internet. Future research directions are also discussed in this chapter.

7.1 Principal Contributions

The objective of this dissertation has been set to develop a unified source modeling framework and a unified rate control algorithm for visual coding and transmission. The source model and the rate control algorithm need to be accurate, robust, fast, and cost-effective. To accomplish this objective, we have developed the ρ-domain R-D analysis where the R-D functions are studied in the ρ-domain instead of the conventional q-domain. As we have observed throughout this dissertation, the R-D functions have unique behaviors in the ρ-domain which enables us to analyze, model and estimate them accurately and robustly. Based on our extensive simulation results with various coding algorithms and a wide range of image/video data, we have shown that the rate function in any typical transform coding system is a linear function in the ρ-domain. With Shannon's source coding theorem, we have provided a theoretical justification

for this linear rate model. Based on this model, a unified rate control algorithm was developed for all standard video coding systems, such as MPEG-2, H.263, and MPEG-4. Our experimental results show that the proposed rate control algorithm outperforms other algorithms reported in the literature by providing much more accurate and robust rate control. Within the framework of ρ-domain analysis, we have also developed a generic distortion model. Based on this distortion model and the linear rate model, an optimum bit allocation scheme has been developed in the ρ-domain. We have applied the proposed bit allocation scheme to MPEG-4 coding at the object-level and to H.263 coding at the macroblock level. Our experimental results have shown that the picture quality has been significantly improved by the proposed optimum bit allocation scheme.

In video coding, the parameters of the rate and distortion models are estimated from the context information such as the coding statistics of previous frames or macroblocks. To estimate the R-D functions without context information, we have developed a unified ρ-domain source modeling framework for transform coding of images and videos. In this framework, the characteristics of the input source data and the model of the coding algorithm are integrated by the rate curve decomposition scheme. This scheme allows us to estimate the R-D functions with very low complexity before quantization and coding. Our extensive simulation results have shown that the estimation accuracy is very high. Based on this R-D estimation algorithm, we have developed a frame-level rate control algorithm which is able to maintain a even steadier buffer level than the macroblock-level TMN8 rate control algorithm. With the estimated R-D functions, we also developed an encoder-based rate shape smoothing algorithm. We have shown that, with this smoothing algorithm, the output bit stream of the video encoder has both a smoothed rate shape and a consistent picture quality, which is highly desirable in practical video coding applications.

7.2 Future Directions

Some ideas for future extensions to the ρ-domain R-D analysis and rate control developed in this dissertation are listed in the following:

- **Further reduction of rate control complexity**

  In Chapter 4, we presented two rate control algorithms: ρ-RC-1 and ρ-RC-2. From Section 4.2.3, we know ρ-RC-2 has only about half the complexity of ρ-RC-1. However, from the simulation results presented in Section 4.3, we can see that they have almost the same rate control performance. This implies that further reduction of the algorithm complexity without significant performance degradation is still possible. In Eq. (4.2), to obtain the target number of bits R_C for the transform coefficients, we need to precompute the motion vectors and header information bits. To do this, we need to complete the motion compensation for all of the macroblocks before starting the rate control and coding process. However, in the software reference models for MPEG-2, H.263, and MPEG-4, the motion compensation and coding of the current macroblock must be completed before switching to the next macroblock. As a result, our rate control algorithm does not fit the macroblock-wise coding framework in a natural way. Obviously, this increases the implementation cost. For this reason, it is worthwhile to investigate an adaptive estimation scheme for the motion vectors and header information bits, and to develop a macroblock-wise rate control algorithm which can fit the actual coding process most naturally.

- **Adaptive ρ-domain source modeling**

  In Chapter 6, using the rate curve decomposition methodology, we developed a ρ-domain source modeling framework. In this framework, the coding algorithm is modeled by the decomposition coefficients which are obtained by linear regression

of the training data. During the R-D estimation process, these decomposition coefficients are fixed. To improve the estimation accuracy and robustness, we can adaptively update the decomposition coefficients after a certain number of pictures are processed by the estimation algorithm. We expect that the decomposition coefficients will model the coding algorithm more accurately and robustly as more pictures are processed.

- **Rate control for storage video coding and transcoding**

  The rate control algorithms developed in Chapter 4 have superior performance for video coding, especially for live video coding and conversational video technology. The storage video coding, which is another type of video application, is different from the live video coding in that its coding could be offline with multiple passes [24]. In this case, the R-D functions of every video frame inside a GOP can be modeled with the R-D models developed in Chapters 3 and 5, or estimated with the R-D estimation algorithm developed in Chapter 6. Based on the R-D functions, optimum bit allocation can be employed to maximize the video presentation quality at the GOP level [38, 78].

  Because of the heterogeneous nature of the network, different users may connect to the network with different bandwidth. In addition, even for the same user, the available channel bandwidth varies dramatically over time. Therefore, in real-time transport of the precoded video data, such as video on demand (VOD) [81, 82], video transcoding is necessary in order to adjust the coding bit rate according to the available channel bandwidth [79, 80]. The proposed rate control algorithm, R-D estimation, and optimum bit allocation can be applied to control the output bit rate of the video transcoder and to improve the picture quality.

- **Rate control for joint source and channel coding**

The coded video data is transmitted through the communication channel. A real-world communication channel often suffers from transmission errors, especially in wireless communication [83]. In this case, channel coding [84] is necessary in order to protect the video data and guarantee the video presentation quality at the receiver end [85, 86]. In this way, the actual coding bit rate or bandwidth requirement of the video data should be the coding bit rate after channel coding. Therefore, it is necessary to model the R-D behavior of the joint source-channel coding instead of the source coding only. Based on these R-D models, we can then develop an optimum bit allocation scheme and a rate control algorithm to maximize the video presentation quality at the receiver end.

- **Rate control for network video transmission**

  In this dissertation, we employ a simplified network model where only the channel bandwidth is considered. In real-world networks for video transmission, we need also to consider other factors, such as transmission delay, cell-delay variation, and pack-loss ratio [87]. Potentially, all of these factors have an effect on the video presentation quality at the receiver. If the transmission delay is considered, the coding bit rate should satisfy the buffering constraints presented in [87]. Therefore, the rate control algorithm can be employed to maximize the picture quality under these buffering constraints. Practical video coding should also consider the pack loss. Given a network pack-loss ratio, the encoder setting can be optimized to acheive the best video presentation quality at the receiver. For example, the selection of the intra/inter coding mode of each macroblock can be R-D optimized [88, 89] to improve the quality of the video transmitted over the network. Based on the source modeling framework developed in this dissertation, it is worthwhile to investigate an R-D optimized encoder control algorithm for video coding and end-to-end network transmission.

Bibliography

[1] Advanced Television Systems Committee, "ATSC digital television standard," September 1995.

[2] G. K. Wallace, "The JPEG still picture compression standard," *Commun. ACM*, vol. 34, pp. 30–44, April 1991.

[3] W. B. Pennebaker and J. L. Mitchell, *JPEG Still Image Data Compression Standard*, Van Nostrand Reinhold, New York, 1993.

[4] "JPEG-2000 verification model version 5.2," ISO/IEC JTC1/SC29/WG01 Ad hoc Group on JPEG-2000 Verification Model, WG1 N1422, August 1999.

[5] M.W. Marcellin, M. J. Gormish, A. Bilgin, and M. P. Boliek, "An overview of JPEG-2000," *Proceedings DCC 2000*, Snowbird, UT, USA, pp. 523–541, March 2000.

[6] D. LeGall, "MPEG: A video compression standard for multimedia application," *Commun. ACM*, vol. 34, pp. 46–58, April 1991.

[7] B. G. Haskell, A. Puri, and A. N. Netravali, *Digital Video: An Introduction to MPEG-2,* Chapman and Hall, 1997.

[8] ITU-T, "Video coding for low bit rate communications," *ITU-T Recommendation H.263*, version 1, version 2, January 1998.

[9] G. Cote, 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, November 1998.

[10] T. Sikora, "The MPEG-4 video standard verification model," *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 7, pp. 19–31, February 1997.

[11] U. Reimers, "Digital video broadcasting," *IEEE Communications Magazine* pp. 104–110, June 1998.

[12] N. Ahmed, T. Natarajan, and K. R. Rao, "Discrete cosine transform," *IEEE Transactions of Computers*, vol. C-23, pp. 90–93, January 1974.

[13] M. Antonini, M. Barlaud, P. Mathieu, and I. Daubechies, "Image coding using wavelet transform," *IEEE Trans. Image Processing,* vol. 1, pp. 205–220, April 1992.

[14] H. -J. Kim and C. C. Li, "Lossless and lossy image compression using biorthogonal wavelet transforms with multiplierless operations," *IEEE Trans. Circuits and systems-II: Analog and Digital Signal Processing*, vol. 45, no. 8, August 1998.

[15] A. N. Netravali and B. Haskell, *Digital Picture: Representation and Compression*, Plenum Press, New York, 1988.

[16] J. M. Shapiro, "Embedded image coding using zero-trees of wavelet coefficients," *IEEE Trans. Signal Processing*, vol. 41, pp. 3445–3462, December 1993.

[17] I. H. Witten, R.M. Neal, and J.G. Cleary, "Arithmetic coding for data compression," *Commun. ACM*, vol. 30, pp. 520–540, June 1996.

[18] A. Said and W. A. Pearlman, "A new fast and efficient image codec based on set partitioning in hierarchical trees," *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 6, pp. 243–250, June 1996.

[19] M. J. Tsai, J. D. Villasenor, and F. Chen, "Stack-run image coding," *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 6, pp. 519–521, October 1996.

[20] H. Gish and J. N. Pierce, "Asymptotically efficient quantizing," *IEEE. Trans. on Inform. Theory*, vol. IT-14, pp. 676–683, September 1968.

[21] T. Berger, *Rate Distortion Theory,* Prentice Hall, Englewood Cliffs, NJ, 1984.

[22] H. -M. Hang and J. -J. Chen, "Source model for transform video coder and its application – Part I: Fundamental theory," *IEEE Trans. Circuits and Systems for Video Technology*, vol. 7, pp. 287–298, April 1997.

[23] J. Ribas-Corbera and S. Lei, "Rate control in DCT video coding for low-delay communications," *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 9, pp. 172–185, February 1999.

[24] W. Ding and B. Liu, "Rate control of MPEG video coding and recording by rate-quantization modeling," *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 6, pp. 12–20, Feb. 1996.

[25] C. -W. Lin, T. -J. Liou, and Y. -C. Chen, "Dynamic rate control in multipoint video transcoding," *2000 IEEE International Symposium on Circuits and Systems,* Geneva, Switzerland, pp. 28–31 May 2000.

[26] L. Wang, "Rate control for MPEG video coding," *Signal Processing: Image Communication*, vol. 15, pp. 493–511, March 2000.

[27] Y. Shoham and A. Gersho, "Effcient bit allocation for an arbitrary set of quantizers," *IEEE Transaction on Acoustic, Speech, Signal Processing,* vol. 36, pp. 1445–1453, September 1988.

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

[29] T. Wiegand, M. Lightstone, D. Mukherjee, T. G. Campbell, and S. K. Mitra, "Rate-distortion optimized mode selection for very low bit rate video coding and the emerging H.263 standard," *IEEE Transactions on Circuits and Systems for Video Technology,* vol. 6, pp. 182–190, April 1996.

[30] J. Shen, W. -Y. Chen, "Fast rate-distortion optimization algorithm for motion-compensated transform coding of video," *IEE Electronics Letters*, vol. 36, pp. 305–306, Feb. 2000.

[31] A. Vetro, H. Sun, and Y. Wang, "MPEG-4 rate control for multiple video objects," *IEEE Transactions on Circuits and Systems for Video Technology,* vol. 9, pp. 186–199, February 1999.

[32] "MPEG-2 Video Test Model 5," *ISO/IEC JTC1/SC29/WG11 MPEG93/457*, April 1993.

[33] J. Ribas-Corbera and S. Lei, "Contribution to the rate control Q2 experiment: A quantizer control tool for achieving target bit rates accurately," *Coding of Moving Pictures and Associated Audio MPEG96/M1812 ISO/IEC JTC/SC29/WG11*, Sevilla, Spain, February 1997.

[34] Video Group, "Text of ISO/IEC 14496-2 MPEG4 video VM - Version 8.0," *ISO/IEC JTC1/SC29/WG11 Coding of Moving Pictures and Associated Audio MPEG 97/W1796,* Stockholm, Sweden, July 1997.

[35] T. Chiang and Y. -Q. Zhang, "A new rate control scheme using quadratic rate distortion model," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 7, pp. 246–250, Feb. 1997.

[36] B. Tao, H. A. Peterson, and B. W. Dickinson, "A rate-quantization model for MPEG Encoders," *Proceedings of 1997 International Conference on Image Processing*, pp. 338–341, October 1997.

[37] K. H. Yang, A. Jacquin, and N. S. Jayant, "A normalized rate-distortion model for H.263-compatible codecs and its application to quantizer selection," *Proceedings of 1997 International Conference on Image Processing*, pp. 41–44, October 1997.

[38] L.-J. Lin and A. Ortega, "Bit-rate control using piecewise approximated rate-distortion characteristics," *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 38, pp. 82–93, January 1990.

[39] E. D. Frimout, J. Biemond, and R. L. Lagendik, "Forward rate control for MPEG recording," *Proceedings of SPIE Visual Commun. Image Processing,* Cambridge, MA, pp. 184–194, November 1993.

[40] A. Y. K. Yan and M. L. Liou, "Adaptive predictive rate control algorithm for MPEG videos by rate quantization method," *Proceedings of Picture Coding Symposium,* Berlin, Germany, pp. 619–624, September 1997.

[41] H. -J. Lee, T. Chiang, and Y.-Q. Zhang, "Scalable rate control for MPEG-4 video," *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 10, pp. 878–894, September 2000.

[42] Z. He, T. -H. Yu, and S. K. Mitra, "Blockwise zeros mapping image coding," *Proceedings of International Conference on Image Processing*, Vancouver, Canada, September 2000.

[43] T. -H. Yu, Z. He, and S. K. Mitra, "A novel coding scheme for wavelet image compression," *Proceedings of 34th Asilomar Conference on Signals, Systems, and Computers*, Pacific Grove, CA, October 2000.

[44] S. Patel and S. Srinivasan, "Modified embedded zero-tree wavelet algorithm for fast implementation of wavelet image codec," *IEE Electronics Letters*, vol. 36, pp. 1713–1714, September 2000.

[45] B. A. Banister and T. R. Fischer, "Quantization performance in SPIHT and related wavelet image compression algorithms," *IEEE Signal Processing Letters,* vol. 6, pp. 97–99, May 1999.

[46] S. Battista, F. Casalino, and C. Lande, "MPEG-4: a multimedia standard for the third millennium," *IEEE Multimedia*, vol. 7, pp. 76–84, March 2000.

[47] N. Brady, "MPEG-4 standardized methods for the compression of arbitrarily shaped video objects," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 9, pp. 1170–1189, December 1999.

[48] J. -W. Lee and Y. -S. Ho, "Target bit matching for MPEG-2 video rate control," *Proceedings of Tencon 1998*, New Delhi, India, vol. 1, pp. 66–69, December 1998.

[49] H. -J. Lee, T. Chiang, and Y. -Q. Zhang, "Scalable rate control for MPEG-4 video," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 10, pp. 878–894, September 2000.

[50] P. P. Vaidyanathan, "Filter banks with maximum coding gain and energy compaction," *Conference Record of The Twenty-Ninth Asilomar Conference on Signals, Systems and Computers*, Los Alamitos, CA, USA, vol. 1, pp. 36–40, November, 1996.

[51] Z. Xiong, K. Ramchandran, M. T. Orchard, and Y. -Q. Zhang, "A comparative study of DCT- and wavelet-based image coding," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 9, pp. 692–695, August 1999.

[52] I. Daubechies, *Ten lectures on wavelets*, SIAM, Philadelphia, PA, 1992.

[53] Telenor codec, "ITU-T/SG-15, video codec test model, TMN5," Telenor Research, January 1995.

[54] S. M. LoPresto, K. Ramchandran, and M. T. Orchard, "Image coding based on mixture modeling of wavelet coefficients and a fast estimation-quantization framework," *Proceedings 1997 Data Compression Conference*, Snowbird, UT, USA, pp. 221–230. March, 1997.

[55] E. Y. Lam and J. W. Goodman, "A mathematical analysis of the DCT coefficient distributions for images," *IEEE Transactions on Image Processing*, vol. 9, pp. 1661–1666, October 2000.

[56] R. W. Buccigrossi and E. P. Simoncelli, "Image compression via joint statistical characterization in the wavelet domain," *IEEE Transactions on Image Processing*, vol. 8, pp. 1688–1701, December 1999.

[57] S. M. LoPresto, K. Ramchandran, and M. T. Orchard, "Image coding based on mixture modeling of wavelet coefficients and a fast estimation-quantization framework," *Proceedings of Data Compression Conference*, Snowbird, UT, March 1997.

[58] A. J. Viterbi and J. K. Omura, *Principles of Digital Communication and Coding*, McGraw-Hill, New York, 1979.

[59] T. G. Cover and J. A. Thomas, *Elements of Information Theory*, John Wiley and Sons, Inc., 1991.

[60] T. Chang and C. -C. J. Kuo, "Texture analysis and classification with tree-structured wavelet transform," *IEEE Trans. on Image Processing*, vol. 2, pp. 432–435, October 1993.

[61] MoMuSys codec, "MPEG4 verification model version 7.0," *ISO / IEC JTC1 / SC29 / WG11 Coding of Moving Pictures and Associated Audio MPEG97*, Bristol, U.K., March 1997.

[62] J. J. Huang and P. M. Schultheiss, "Block quantization of correlated Gaussian random variables," *IEEE Transactions on Communication System*, vol. CS-11, pp. 289–296, September, 1963.

[63] A. Segall, "Bit allocation and encoding for vector sources," *IEEE Trans. on Information Theory*, vol. IT-22, pp. 162–169, March 1976.

[64] B. Fox, "Discrete optimization via marginal analysis," *Management Science*, vol. 13, pp. 210–216, November 1966.

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

[66] R. L. Joshi, T. R. Fischer, and R. H. Bamberger, "Optimum classification in subband coding of images," *Proceedings of International Conference on Image Processing*, vol. 2, pp. 883–887, November 1994.

[67] R. L. Joshi, V. J. Crump and T. R. Fischer "Image subband coding using arithmetic coded trellis coded quantization," *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 5, pp. 515–523, December 1995.

[68] Z. Xiong and X. Wu, "Wavelet image coding using trellis coded space-frequency quantization," *IEEE Signal Processing Letters*, vol. 6, pp. 158–161, July 1999.

[69] D. Decarlo, and D. Metaxas "Optical flow constraints on deformable models with applications to face tracking," *International Journal of Computer Vision*, vol. 38, pp. 99–127, August 2000.

[70] L. Zhang, "Tracking a face for knowledge-based coding of videophone sequences," *Signal Processing: Image Communication*, vol. 10, pp. 93–114, July 1997.

[71] M. Kunt, A. Ikonomopoulos, and M. Kocher, "Second generation image coding techniques," *Proceedings of IEEE*, vol. 73, pp. 549–575, April 1985.

[72] J. Benois-Pineau, D. Barba, N. Sarris, and M. G. Strintzis, "Video coding for wireless varying bit-rate communications based on area of interest and region representation," *Proceedings. International Conference on Image Processing*, Santa Barbara, CA, USA, vol. 3, pp. 555–558, October 1997.

[73] D. Wu, Y. T. Hou, and Y. -Q. Zhang, "Transporting real-time video over the Internet: challenges and approaches," *Proceedings of the IEEE*, vol. 88, December 2000.

[74] Y. T. Hou, D. Wu, W. Zhu, H. -J. Lee, T. Chiang, and Y. -Q. Zhang, "An end-to-end architecture for MPEG-4 video streaming over the Internet," *Proceedings of 1999 International Conference on Image Processing*, vol. 1, pp. 254–257, October 1999.

[75] S. Sen, J. L. Rexford, J. K. Dey, J. F. Kurose, and D. F. Towsley, "Online smoothing of variable-bit-rate streaming video," *IEEE Transactions on Multimedia*, vol. 2, pp. 37–48, March 2000.

[76] S. Sen, J. K. Dey, J. F. Kurose, J. A. Stankovic and D. Towsley, "Streaming CBR transmission of VBR stored video," *Proceedings of the SPIE*, pp. 26–36, November, 1997.

[77] H. -C. Yang and H. -M. Hang, "Dynamic resource allocation for VBR video transmission," *Proceedings of the SPIE Visual Communications and Image Processing '99*, pp. 821–832, January, 1999.

[78] T. V. Lakshman, A. Ortega and A. R. Reibman, "VBR video: tradeoffs and potentials," *Proceedings of the IEEE*, vol. 86, pp. 952–973, May 1998.

[79] P. A. A. Assuncao and M. Ghanbari, "Transcoding of single-layer MPEG video into lower rates," *IEE Proceedings-Vision, Image and Signal Processing*, vol. 144, pp. 377–383, December 1997.

[80] K. -S. Kan, K. -C. Fan, and Y. -H. Huang, "Low-complexity and low-delay video transcoding for compressed MPEG-2 bitstream," *Proceedings of 1997 IEEE International Symposium on Consumer Electronics*, New York, NY, USA, pp. 99–102, December 1997.

[81] J. Zamora, S. Jacobs, A. Eleftheriadis, S. -F. Chang, and D. Anastassiou, "A practical methodology for guaranteeing quality of service for video-on-demand," *IEEE Transactions on Circuits and Systems for Video Technology,* vol. 10, pp. 168–178, February 2000.

[82] Y. -H. Chang, D. Coggins, D. Pitt, D. Skellern, M. Thapar, and C. Venkatraman, "An open-systems approach to video on demand," *IEEE Commununication Magazine*, vol. 32, pp. 68–80, May 1994.

[83] L. Hanzo, C. H. Wong, and P. Cherriman, "Channel-adaptive wideband wireless video telephony," *IEEE Signal Processing Magazine*, vol. 17, pp. 10–30, July 2000.

[84] J. Hagenauer, "Rate-compatible punctured convolutional codes (RCPC codes) and their applications," *IEEE Trans. Commun.,* vol. 36, pp. 389–399, April 1998.

[85] M. Bystrom and J. W. Modestino, "Combined source-channel coding schemes for video transmission over an additive white Gaussian noise channel," *IEEE Journal on Selected Areas in Communications,* vol. 18, pp. 880–890, June 2000.

[86] M. Srinivasan and R. Chellappa, "Adaptive source-channel subband video coding for wireless channels," *IEEE J. Select. Areas Commun.,* vol. 16, December 1998.

[87] A. R. Reibman and B. G. Haskell, "Constraints on variable bit-rate video for ATM networks," *IEEE Trans. Circuits Syst. Video Technol.,* vol. 2, pp. 361–372, December 1992.

[88] R. Zhang, S. L. Regunathan, and K. Rose, "Video coding with optimal inter/intra-mode switching for packet loss resilience," *IEEE Journal on Selected Areas in Communications,* vol. 18, pp. 966–976, June 2000.

[89] D. Wu, Y. T. Hou, B. Li, W. Zhu, Y. -Q. Zhang, and H. J. Chao, "An end-to-end approach for optimal mode selection in Internet video communication: theory and application," *IEEE Journal on Selected Areas in Communications,* vol. 18, pp. 977–995, June 2000.

[90] C.-Y. Hsu, A. Ortega, and A. R. Reibman, "Joint selection of source and channel rate for VBR video transmission under ATM policing constraints," *IEEE J. Select. Areas Commun.,* vol. 15, pp. 1016–1028, August 1997.

[91] M. Balakrishnan and R. Cohen, "Global optimization of multiplexed video encoders," *Proc. Int. Conf. Image Processing* Santa Barbara, CA, October 1997.

[92] Z. He, T. -H. Yu, and S. K. Mitra, "A fast, accurate and forward rate prediction and control algorithm for wavelet image coders," *Proceedings of 34th Asilomar Conference on Signals, Systems and Computers,* Pacific Grove, CA, October 2000.

[93] G. de Veciana, G. Kesidis, and J. Walrand, "Resource management in wide-area ATM networks using effective bandwidths," *IEEE Journal on Selected Areas in Communications,* vol. 13, pp. 1081–1090, August 1995.