Rate-distortion based optimal bit allocation for video streaming applications

Wu, Jianhua

2007

Wu, J. (2007). Rate-distortion based optimal bit allocation for video streaming applications. Doctoral thesis, Nanyang Technological University, Singapore.

<https://hdl.handle.net/10356/2535>

<https://doi.org/10.32657/10356/2535>

Nanyang Technological University

Downloaded on 26 Apr 2023 23:27:49 SGT

NANYANG TECHNOLOGICAL UNIVERSITY

Rate-Distortion Based Optimal Bit Allocation for Video Streaming Applications

A Thesis

Submitted to the School of Computer Engineering
of the Nanyang Technological University

by **Wu Jianhua**

in Fulfillment of the Requirement
for the Degree of Doctor of Philosophy

August 2007

Abstract

Over the last decade, with the great advances in video compression and network technologies, significant progress has been made in video streaming over packet-switched networks such as Internet. However, challenges arise with growing demand on guaranteeing end-to-end quality of service (QoS), and delivering video over resource-poor wireless networks, power-limited handheld devices and heterogeneous environment. In this research, we focus on the bit allocation issues for various video streaming scenarios including stored video adaptation, real-time video encoding and joint source channel coding.

In particular, we first study the constant quality constrained bit allocation problem for leaky prediction based FGS (L-FGS). We propose an accurate rate-distortion (R-D) model, which considers not only the distortion introduced in the current frame and the propagated distortion from the reference frame due to rate adaptation, but also the correlation between them. Based on our proposed R-D model, a sliding window technique is further developed to solve the problem of constant quality constrained bit allocation. Experimental results show that the proposed R-D model is very accurate and the corresponding bit allocation algorithm can achieve much smoother video quality than the traditional uniform bit allocation under both CBR (constant bit rate) and VBR (variable bit rate) channels.

Second, we look into the bit allocation problem in real-time video encoding. Existing bit allocation algorithms for real-time video encoding mainly target for quality smoothing. The rate fluctuation problem has not been well considered. Specifically, the VBR bitstreams of the quality-smoothed encoding schemes exhibit significant burstiness, where the high peak and the large variation in bit rates can substantially increase bandwidth requirement for the continuous playback at the client site. Therefore, we propose a rate-smoothed encoding scheme for real-time video streaming applications, which can

reduce the traffic rate burstiness. In the case that the network resource is sufficiently large and the video quality has reached the target quality, the quality smoothing is not necessary and the rate smoothing can be performed to avoid overusing the unnecessary network resource but also achieve a smoothed traffic rate. Thus, we further propose a joint quality and rate smoothed encoding scheme, which can provide adaptive smoothing according to different situations.

Third, we study joint source channel coding for bitplane based video coding over wireless channels. We consider using frame-level intra mode to stop error propagation and using UEP (unequal error protection) to combat channel errors. The question is how to optimally allocate bits between source coding and channel coding. In other words, our focus is on how to optimally select coding modes and find UEP solutions for bitplane based video coding. In particular, we propose an overall end-to-end R-D function, which considers not only the source distortion and the channel distortion introduced in the current frame but also the propagated channel distortion from the previous frames. Based on this end-to-end R-D function, we are able to find the optimal solutions for both mode selection and UEP so that an optimal tradeoff between efficiency and robustness can be achieved.

Our work throughout this dissertation is based on the accurate rate distortion analysis and modelling of the entire video transmission systems, and the experimental results show that better performance can be achieved in terms of efficiency, robustness and adaptivity.

Acknowledgments

I am especially grateful to my supervisor, Dr. Cai Jianfei, for his excellent guidance, constant encouragement throughout my PhD study in NTU. Without his help and patience, this thesis cannot reach current stage. I have learnt a lot from him, which will be greatly beneficial for my future career.

I would like to thank Associate Professor Chia Liang Tien, and all the faculty members and technicians in Center for Multimedia and Network Technology (CeMNet), for their kind help and generous encouragement.

I would like to thank all my friends and colleagues in CeMNet, especially Li Xiangjun, Zhan Wenning, Huang Haijie, Ni Zefeng, Tao Dayong, Yang Wenxian, Xu Min, Gao Deyun, and Xie Jun, who are the students and research staffs of Visual Signal Processing and Communications Group, for their precious discussions and friendship.

I would like to thank my parents for their constant support, understanding and encouragement.

Contents

Abstract	i
Acknowledgments	iii
List of Figures	vii
List of Tables	xi
Abbreviation	xii
1 Introduction	1
1.1 Research Background	1
1.2 Problem Statement and Research Objective	4
1.3 Summary of Contributions	6
1.4 Dissertation Outline	7
2 Video Coding and Transmission Techniques	8
2.1 Video Coding Standards	8
2.1.1 H.261	9
2.1.2 H.263	9
2.1.3 MPEG-1, MPEG-2 and MPEG-4	11
2.1.4 H.264	12
2.2 Scalable Video Coding Techniques	13
2.2.1 Scalable Video Coding in MPEG-2	14
2.2.2 FGS Video Coding Scheme	16
2.2.3 Scalable Extension of H.264/MPEG-4 AVC	18

2.3	Rate-Distortion Theory	19
2.4	Rate Control Techniques	20
2.4.1	TM5 Rate Control Algorithm	21
2.4.2	TMN8 Rate Control Algorithm	22
2.4.3	MPEG-4 Rate Control Algorithm	23
2.4.4	H.264 Rate Control Algorithm	24
2.5	Error Control Techniques	25
2.5.1	FEC Coding	26
2.5.2	Automatic Repeat Request	26
2.5.3	Motion-Compensated Error Concealment	27
2.5.4	Layered Video Coding with Transport Prioritization	27
2.5.5	Multiple Description Coding	28
2.5.6	Joint Source and Channel Coding	29
3	Rate-Distortion Analysis of Leaky Prediction Based FGS Video for Constant Quality Constrained Rate Adaptation	32
3.1	Related Work	33
3.2	Overview of Leaky Prediction Based FGS	35
3.3	Rate-Distortion Analysis of L-FGS	37
3.3.1	Proposed R-D Model	37
3.3.2	Experimental Results	42
3.4	Constant Quality Constrained Rate Adaptation	43
3.4.1	Proposed Bit Allocation Algorithm	43
3.4.2	Experimental Results	45
3.5	Summary	46
4	Joint Quality and Rate Smoothed Encoding for Real-Time Video Stream- ing Applications	59
4.1	Problem Statement	60
4.2	Overview of ρ -Domain Rate Control Algorithm	61
4.3	Quality-Smoothed Encoding	63
4.3.1	Quality Smoothing Algorithm	63

4.3.2	Results of Quality Smoothing	64
4.4	Rate-Smoothed Encoding	67
4.4.1	Rate Smoothing Algorithm	67
4.4.2	Results of Rate Smoothing	70
4.5	Joint Distortion and Rate Smoothed Encoding	71
4.5.1	Joint Smoothing Algorithm	71
4.5.2	Results of Joint Smoothing	73
4.6	Summary	74
5	Joint Mode Selection and Unequal Error Protection for Bitplane Coded Video Transmission over Wireless Channels	80
5.1	Related Work	81
5.2	Problem Statement	83
5.2.1	Mode-Adaptive FGS System Architecture	83
5.2.2	Unequal Error Protection	85
5.2.3	Problem Formulation	86
5.3	Analysis of End-to-End Distortion	87
5.4	Joint Source Channel Optimization	91
5.4.1	R-D Extraction of FGS Source Coding	91
5.4.2	Channel Distortion Estimation	93
5.4.3	Dynamic Programming for UEP	94
5.5	Experimental Results	96
5.6	Summary	98
6	Conclusions and Future Directions	101
6.1	Conclusions	101
6.2	Future Directions	102
	References	105
	Publications	115

List of Figures

2.1	SNR scalable video encoder.	15
2.2	FGS encoder structure.	17
2.3	The hierarchical coding structure with four temporal levels.	19
2.4	An example of R-D curve.	20
3.1	Comparison between the overall distortion $D(n)$, the solid line with “+” marks, and $D_I(n) + \alpha_{n-1}^2 \rho_{n-1} D_P(n-1)$, the solid line with “*” marks. Left: QCIF Akiyo at 48 kbps. Right: QCIF Akiyo at 80 kbps.	39
3.2	The estimation error of using Eq. (3.12) to calculate $D(n)$ of the 50-th frame of QCIF Foreman.	40
3.3	The R-D relationship within each bitplane of the 20-th enhancement layer frame of the QCIF Foreman video sequence at 10 fps and 32 kbps for the base layer. The R-D curves for all the bitplanes are placed in an increasing order (starting with the first bitplane) from left to right and from top to bottom. The x-axis represents number of bits and the y-axis represents MSE.	47
3.4	The comparison between the actual values of $D_P(n)$ and the estimated values using Eq. (3.14) for the 50-th frames of different video sequences. Left: QCIF Foreman. Right: QCIF Akiyo.	48
3.5	Distortion estimation results for adapting the L-FGS coded QCIF Foreman video sequence to CBR channels. Left: QCIF Foreman at 64 kbps. Right: QCIF Foreman at 128 kbps.	48
3.6	Distortion estimation results for adapting the L-FGS coded QCIF Akiyo video sequence to CBR channels. Left: QCIF Akiyo at 48 kbps. Right: QCIF Akiyo at 80 kbps.	49

3.7	Distortion estimation results for adapting the L-FGS coded QCIF Mother & Daughter video sequence to CBR channels. Left: QCIF Mother & Daughter at 48 kbps. Right: QCIF Mother & Daughter at 80 kbps. . . .	49
3.8	Distortion estimation results for adapting the L-FGS coded QCIF News video sequence to CBR channels. Left: QCIF News at 64 kbps. Right: QCIF News at 128 kbps.	50
3.9	Distortion estimation results for adapting the L-FGS coded QCIF Coastguard video sequence to CBR channels. Left: QCIF Coastguard at 64 kbps. Right: QCIF Coastguard at 128 kbps.	50
3.10	Distortion estimation results for adapting the L-FGS coded QCIF Foreman video sequence to the VBR channel. Left: Available bandwidth. Right: Distortion estimation results.	51
3.11	Distortion estimation results for adapting the L-FGS coded QCIF Akiyo video sequence to the VBR channel. Left: Available bandwidth. Right: Distortion estimation results.	51
3.12	Distortion estimation results for adapting the L-FGS coded QCIF Mother & Daughter video sequence to the VBR channel. Left: Available bandwidth. Right: Distortion estimation results.	52
3.13	Distortion estimation results for adapting the L-FGS coded QCIF News video sequence to the VBR channel. Left: Available bandwidth. Right: Distortion estimation results.	52
3.14	Distortion estimation results for adapting the L-FGS coded QCIF Coastguard video sequence to the VBR channel. Left: Available bandwidth. Right: Distortion estimation results.	53
3.15	Distortion for adapting the L-FGS coded QCIF Foreman sequence to CBR channels. Left: QCIF Foreman at 64 kbps. Right: QCIF Foreman at 128 kbps.	53
3.16	Distortion for adapting the L-FGS coded QCIF Akiyo sequence to CBR channels. Left: QCIF Akiyo at 48 kbps. Right: QCIF Akiyo at 80 kbps.	54
3.17	The simulation results for adapting the L-FGS coded Foreman sequence to the VBR channel. Left: Available bandwidth. Right: Distortion performance with window size=80.	57

3.18	The simulation results for adapting the L-FGS coded Akiyo sequence to the VBR channel. Left: Available bandwidth. Right: Distortion performance with window size=80.	58
4.1	The PSNR results of individual video frames for QCIF Foreman. Left: QCIF Foreman at 128 kbps. Right: QCIF Foreman at 256 kbps.	75
4.2	The PSNR results of individual video frames for QCIF News. Left: QCIF News at 64 kbps. Right: QCIF News at 96 kbps.	76
4.3	The PSNR results of individual video frames for QCIF Coastguard. Left: QCIF Coastguard at 256 kbps. Right: QCIF Coastguard at 400 kbps. . .	76
4.4	The rate-distortion performance of individual video frames for QCIF Foreman under different bit allocation schemes with the target average PSNR of 38 dB. Left: the rate results. Right: the PSNR results.	77
4.5	The rate-distortion performance of individual video frames for QCIF News under different bit allocation schemes with the target average PSNR of 38 dB. Left: the rate results. Right: the PSNR results.	77
4.6	The rate-distortion performance of individual video frames for QCIF Coastguard under different bit allocation schemes with the target average PSNR of 38 dB. Left: the rate results. Right: the PSNR results.	78
4.7	The rate-distortion performance of individual video frames for QCIF Foreman coded at 420 kbps. Left: the rate results. Right: the PSNR results.	78
4.8	The rate-distortion performance of individual video frames for QCIF News coded at 160 kbps. Left: the rate results. Right: the PSNR results. . . .	79
4.9	The rate-distortion performance of individual video frames for QCIF Coastguard coded at 560 kbps. Left: the rate results. Right: the PSNR results.	79
5.1	The MA-FGS architecture.	83
5.2	The PSNR performance of encoding the test video sequences at different bit rates. Left: QCIF Foreman with 10 fps and 32 kbps for the base layer. Right: QCIF Akiyo with 10 fps and 16 kbps for the base layer.	85
5.3	The adopted UEP structure.	86

5.4	The R-D relationship within each bitplane of the third frame of the QCIF Foreman video sequence at 10 fps and 32 kbps for the base layer. The previous two enhancement frames are coded at 6400 bits. The R-D curves for all the bitplanes are placed in an increasing order (starting with the first bitplane) from left to right and from top to bottom. The x-axis represents number of bits while the y-axis represents MSE.	92
5.5	The results of the channel distortion estimation for the QCIF Foreman video sequence coded at 10 fps and 96 kbps with 32 kbps for the base layer, no FEC and 5% packet loss rate which only occurs in the latter 30% part of each frame. Top: $\Delta = 1$. Bottom Left: $\Delta = 5$. Bottom Right: $\Delta = 10$	95
5.6	The PSNR performance of transmitting the video sequences with different total bandwidth over a BSC channel with $\epsilon = 0.1$. Left: QCIF Foreman. Right: QCIF Akiyo.	99
5.7	The PSNR performance of transmitting the test video sequences with a fixed total bandwidth over BSC channels with different SERs. Left: QCIF Foreman with 128 kbps. Right: QCIF Akiyo with 64 kbps.	99
5.8	The average PSNR performance of all the video frames for transmitting the test video sequences with a fixed total bandwidth over a BSC channel with $\epsilon = 0.1$. Left: QCIF Foreman with 128 kbps. Right: QCIF Akiyo with 64 kbps.	100
5.9	An example of the channel distortion and the corresponding coding mode for each frame over one simulation in the case of transmitting the QCIF Foreman at 128 kbps over a BSC channel with $\epsilon = 0.1$. Left: Channel distortion. Right: Coding Mode.	100

List of Tables

3.1	The symbol definitions.	36
3.2	Experiment configuration for each video sequence.	42
3.3	The performance comparison between the uniform bit allocation and our proposed scheme for QCIF Foreman.	55
3.4	The performance comparison between the uniform bit allocation and our proposed scheme for QCIF Akiyo.	55
3.5	The performance comparison between the uniform bit allocation and our proposed scheme for QCIF Mother & Daughter.	56
3.6	The performance comparison between the uniform bit allocation and our proposed scheme for QCIF News.	56
3.7	The performance comparison between the uniform bit allocation and our proposed scheme for QCIF Coastguard.	57
4.1	The performance comparison of different bit allocation schemes.	66
4.2	The performance comparison of different bit allocation schemes for a target average PSNR of 35 dB.	71
4.3	The performance comparison of different bit allocation schemes for a target average PSNR of 38 dB.	72
4.4	The performance comparison of different encoding schemes.	75
5.1	The symbol definitions.	84

Abbreviation

AMC-FGS	Adaptive Motion Compensation FGS
ARQ	Automatic Repeat Request
BSC	Binary Symmetric Channel
CBR	Constant Bit Rate
CIF	Common Interchange Format
DCT	Discrete Cosine Transform
DSC	Distributed Source Coding
EEP	Equal Error Protection
FEC	Forward Error Correction
FGS	Fine Granularity Scalability
FGST	FGS Temporal Scalability
GOB	Group of Blocks
GOP	Group of Pictures
HDTV	High-Definition Television
ISDN	Integrated Services Digital Network
ITU-T	ITU Telecommunication Standardization Sector
JSCC	Joint Source and Channel Coding
JSVM	Joint Scalable Video Model
L-FGS	Leaky Prediction based FGS
MAD	Mean Absolute Difference
MA-FGS	Mode Adaptive FGS
MB	Macroblock
MC-FGS	Motion Compensation FGS
MCTF	Motion-Compensated Temporal Filtering

MDC		Multiple Description Coding
MPEG		Moving Picture Experts Group
MSE		Mean-Squared Error
NAL		Network Abstraction Layer
NTSC		National Television System Committee
PFGS		Progressive FGS
PSNR		Peak Signal-to-Noise Ratio
QCIF		Quarter Common Interchange Format
QoS		Quality of Service
QP		Quantization Parameter
R-D		Rate-Distortion
RDO		Rate-Distortion Optimization
RFGS		Robust FGS
ROPE		Recursive Optimal Per-pixel Estimate
RS		Reed-Solomon
SER		Symbol Error Rate
SNR		Signal-to-Noise Ratio
SVC		Scalable Video Coding
UEP/ULP		Unequal Error/Loss Protection
VBR		Variable Bit Rate
VHS		Video Home System

Chapter 1

Introduction

1.1 Research Background

Over the last decade, with the great advances in video compression and network technologies, significant progress has been made in video streaming over packet-switched networks such as Internet. However, difficulties arise with growing demand on guaranteeing end-to-end quality of service (QoS), and delivering video over resource-poor wireless networks, power-limited handheld devices and heterogeneous environment. In other words, to realize good-quality video streaming at anytime, anywhere and with any device, there are still many challenges remaining.

In particular, one challenge is how to further improve the video coding efficiency. Unlike text or image, video sequences typically have huge volume in data size. For example, without compression, it requires 248 Mbps to transmit an NTSC (National Television System Committee) RGB color video, with 720×480 pixels, 24 bits/pixel and 30 frames/sec. Therefore, in order to store or transmit the huge volume of video data with limited storage capacity or bandwidth, the effective video coding algorithm is necessary and indispensable. Many compression algorithms and video coding standards such as MPEG-4 [1] and H.264 [2] have been developed. Although the latest H.264 technology can averagely achieve a bit-rate gain of 40-50% over the previous low bit-rate

CHAPTER 1. INTRODUCTION

coding technologies, it has seemingly pushed the block-based motion compensation video coding to the limit in the rate-distortion performance.

The second challenge for video streaming is how to deal with the heterogeneity problem. For today's and tomorrow's Internet video streaming systems, heterogeneity is an unavoidable and very challenging problem which multimedia processing community has to face. A source of the heterogeneity is the rapidly growing demands for personalized video delivery, i.e. different users have different requirements. Another source is the severe heterogeneity existing in Internet itself. Although Internet uses the same network protocol, IP, to connect different networks together, it cannot change the fundamental difference when users access multimedia through different access networks. In addition, due to lack of QoS (quality of service) guarantee, services provided by Internet highly depend on network traffic load and essentially Internet can only provide time-varying VBR (variable bit rate) transportation channels at present. Other sources of the heterogeneity include device capabilities, natural environment characteristics and etc [3].

Video adaptation techniques are commonly used to adapt video to various scenarios. For real-time video streaming applications, where video signals are captured, encoded and delivered in real time, e.g. video conferencing, video adaptation can be easily implemented via matching the coding rate with the current available bandwidth. However, for stored video streaming applications, where video signals are pre-encoded and stored in the server, it is not a trivial task to perform video adaptation. In general, video adaptation can be implemented in three ways. The first approach is to store many non-scalable bitstreams for each video sequence. Each bitstream is coded at different formats or different spatial/temporal/SNR (signal-to-noise ratio) resolutions. When a user requests to access the video sequence, the server can send the bitstream which is the closest to the user's requirement. Although this method usually costs more storage spaces in the video server and the chosen bitstream may not satisfy the user's requirement exactly, it

CHAPTER 1. INTRODUCTION

is widely used in practical systems due to its simplicity. Recent research works such as [4] focus on how to efficiently switch among multiple non-scalable bitstreams in order to dynamically adapt to the time-varying network conditions. The second approach is to perform video transcoding [3] including decreasing the spatial resolution, reducing the SNR, or downsampling the temporal frame rate through re-encoding, re-quantization, frame dropping, etc. Although transcoding is very flexible and does not require extra storage space, it needs complex extra processes and is not suitable for large-scale diverse users. The third approach is to use scalable video coding (SVC) [5], which has inherent ability to be adapted to different requirements. Although SVC is the most elegant approach, the high complexity limits its adoption in practical video streaming systems.

The third challenge is the packet loss problem along the path of video delivery. The packet loss could come from traffic congestion at network routers, physical layer channel impairment such as wireless channel fading, connection broken due to the handoff of mobile nodes or leaving nodes in peer-to-peer networks, etc. In addition, the packet delay problem is eventually a problem of packet loss for delay constrained video streaming applications since a packet arriving at the destination beyond the display deadline becomes useless. The packet loss problem may result in severe damage to video quality since the compressed video bitstream is very sensitive to the channel impairment. A single error in a bitstream may cause all the subsequent data become undecodable. Even with some error-resilient techniques such as those in MPEG-4, the effects of channel impairment may still extend both spatially and temporally due to the use of predictive coding. In addition, joint source and channel coding (JSCC) techniques are often used for robust video streaming under middle to heavy packet loss conditions. The difficulty is how to optimally trade off between source coding and channel coding.

Another challenge of video streaming is the low complexity requirement arising from handheld mobile devices. With the rapid growth of wireless communications, wireless

video streaming is expected to be widely deployed in the near future, where handheld mobile devices are the major wireless terminals. Handheld devices such as PDA, Palm, and smart phone have small display screens, limited computational capability and battery power. Thus, it requires a low-complexity video decoder for the case of streaming video to mobile devices. On the other hand, with more and more handheld devices equipped with cameras, it becomes quite common to encode and stream video from mobile devices, which requires a low-complexity video encoder as well. Recently, a new technology called distributed source coding (DSC) [6] has been proposed to transfer the complexity from the video encoder to the decoder while still exploiting the same correlations among video frames. The DSC technology is still in the infant stage and there are many problems remaining to be solved.

1.2 Problem Statement and Research Objective

In this research, we focus on the bit allocation issues for various video streaming scenarios including stored video adaptation, real-time video encoding and joint source channel coding.

Bit allocation in video coding can be generally classified into video rate control, which is very important for achieving good video quality, especially at low bit rates. Current rate control schemes in video coding standards do not have efficient frame-level bit allocation due to the limitation of real-time encoding. Considering off-line video coding or video pre-buffering, it is possible to analyze the importance of a group of video frames in advance so that bits can be optimally allocated among video frames instead of equal bit allocation. However, it is very challenging to perform the frame-level bit allocation due to the interframe dependency. Some works have been conducted to model the relationship between the distortion of current frame and the combination of quantization choices of the previous frame and the current frame [7, 8]. Since the number of combinations for

CHAPTER 1. INTRODUCTION

the successive quantization choices grows exponentially for a group of video frames, it is not possible to get the exact solution for the practical video coding system due to the huge computational complexity.

For real-time video encoding, it is impossible to perform optimal bit allocation among frames since future frames are not available and there is no global statistical information for the entire video sequence. In order to maintain video quality at a certain level, a simple scheme is to encode each video frame with a constant distortion. However, the bit rates of a constant-quality encoded video sequence will often exhibit significant burstiness, which can substantially increase bandwidth requirement.

For video streaming over wireless channels, error correction techniques such as FEC are often combined with error-resilient techniques such as video mode selection to combat channel errors. However, for such a typical joint source channel coding scheme, bit allocation between source coding and channel coding is a very challenging task due to the complexity of the overall system. The bit allocation problem becomes even more difficult when unequal error protection is applied, where different portions of the video data have different protections. In order to obtain the optimal solution for source and channel bit allocation, an overall end-to-end rate-distortion (R-D) function is needed. However, it is non-trivial to model the R-D relationship since the distortion is not only from the source distortion and the channel distortion introduced in the current frame but also the propagated channel distortion from the previous frames.

In summary, the objective of our research is to use the R-D analysis approach to study the bit allocation problems in various video streaming scenarios in order to achieve better performance in efficiency, adaptivity and robustness.

1.3 Summary of Contributions

In this thesis, both pre-encoded video and real-time video streaming applications are studied. We propose a constant quality constrained rate adaption scheme for leaky prediction based FGS (L-FGS) coded video sequences, and a joint quality and rate smoothed encoding scheme for real-time video streaming applications. Furthermore, we also propose a joint source channel coding scheme for bitplane based video coding over wireless channels, where an overall end-to-end R-D function is developed to achieve an optimal tradeoff between coding efficiency and robustness. The detailed contributions of this thesis are as follows:

- Development of an accurate R-D model for L-FGS, which considers not only the distortion introduced in the current frame and the propagated distortion from the reference frame due to rate adaptation, but also the correlation between them. Based on the proposed R-D model, a sliding window technique is further developed to solve the problem of constant quality constrained bit allocation.
- Development of a rate-smoothed encoding scheme for real-time video streaming applications, which can significantly reduce the traffic rate burstiness. By combining the proposed rate-smoothed encoding scheme with the quality-smoothed encoding scheme, a joint encoding scheme has been further proposed, which can provide adaptive smoothing according to different situations.
- Development of joint source and channel coding for bitplane based video coding over wireless channels. We use frame-level intra-mode to stop error propagation and unequal error protection (UEP) to combat channel errors. We propose an overall end-to-end R-D function, which considers not only the source distortion and the channel distortion introduced in the current frame but also the propagated channel

distortion from the previous frames. Based on this end-to-end R-D function, we are able to find the optimal solutions for both mode selection and UEP so that an optimal tradeoff between efficiency and robustness can be achieved.

1.4 Dissertation Outline

The remainder of this dissertation is organized as follows. Chapter 2 reviews some basic video coding and transmission techniques. Chapter 3 presents an accurate R-D model for leaky prediction based FGS and further develops constant quality constrained bit allocation scheme based on the proposed R-D model. Chapter 4 reviews the existing quality-smoothed encoding schemes, and proposes a rate-smoothed video encoding scheme for real-time video streaming applications. By combining the proposed rate smoothed encoding scheme with the quality smoothed encoding scheme, a joint encoding is further proposed, which can provide adaptive smoothing according to different situations. Chapter 5 studies FEC-based joint source channel coding (JSCC) for dependent bitplane based video coding and transmission over wireless channels. Chapter 6 summarizes the studies presented in this dissertation with conclusion remarks and future directions.

Chapter 2

Video Coding and Transmission Techniques

Visual communications have received more and more attention by both industry and academia in recent years. Many techniques have been developed to deal with the problems, such as the limited bandwidth, high error rate, and limited battery power, which are encountered during video streaming. In this chapter, we review some basic video coding and transmission techniques, which include video coding standards, scalable video coding techniques, rate-distortion theory, rate control algorithms and error control techniques.

2.1 Video Coding Standards

Due to the limited bandwidth and storage capacity, highly efficient video coding schemes are needed to compress digital video signals for the purpose of storage or transmission. In the past decade, many video coding standards have been developed, which can be classified into two categories. One is the H.26x series video coding standards developed by the ITU Telecommunication Standardization Sector (ITU-T). The other is the MPEG-x series video coding standards developed by Moving Picture Experts Group (MPEG).

2.1.1 H.261

H.261 [9] is the first video coding standard published by ITU-T in 1990, which is designed for videoconferencing and videophone services over the integrated services digital network (ISDN) at $p \times 64$ Kbits/s, $p = 1, \dots, 30$. H.261 specifies a maximum coding delay of 150 msec since it is mainly intended to support bidirectional video communication.

H.261 supports two input picture formats, which are quarter common interchange format (QCIF) with 176×144 pixels and common interchange format (CIF) with 352×288 pixels. There are two frame types: Intraframe (I-frame) and Interframe (P-frame). I-frame provides an accessing point while P-frame is coded with temporal prediction from the nearest previously coded frame. A macroblock (MB) is the basis for performing the motion compensation (MC), which consists of four luminance blocks (Y_1, Y_2, Y_3, Y_4) and two chrominance blocks (U, V). H.261 supports two MB coding modes: intra mode and inter mode. For the inter mode, motion compensated prediction from the previous frame is performed. The motion compensation is with integer-pel accuracy and there is only one motion vector for one macroblock. After motion compensation, each 8×8 block contained in the MB is discrete cosine transform (DCT) encoded followed by quantization and entropy encoding.

2.1.2 H.263

The development of H.263 [10, 11] standard was started in 1993, which initially targeted at very low bit rate video communication, i.e., bit rate less than 64 kbps. However, this limitation has been removed later on and H.263 can be used for a wide range of bit rates. The version 1 of H.263 standard was finalized in 1995. In order to improve compression performance, error-resilience and scalability, H.263 has been further extended with more features in additional annexes, which leads to H.263 Version 2 (H.263+) and H.263 Version 3 (H.263++).

CHAPTER 2. VIDEO CODING AND TRANSMISSION TECHNIQUES

H.263 can support five standardized picture formats: sub-QCIF (128×96), QCIF (176×144), CIF (352×288), 4CIF (704×576), and 16CIF (1408×1152). Similar to H.261, H.263 utilizes DCT transform to remove spatial redundancies and motion compensation to remove temporal redundancies. Although H.263 uses the similar coding algorithm to that used in H.261, H.263 provides better picture quality at low bit rates with little additional complexity. One major difference between H.263 and H.261 is that motion compensation in H.263 is with half-pel precision while that in H.261 is with full-pel precision. H.263 includes four advanced coding modes: unrestricted motion vectors, syntax-based arithmetic coding, advanced prediction and PB frames. In baseline H.263, only one motion vector per macroblock is allowed for motion compensation. However, the advanced prediction mode allows to use four motion vectors per macroblock, one for each of the four 8×8 luminance blocks. When the unrestricted motion vector mode is used, the range of motion vectors' values is $[-31.5, 31.5]$, instead of $[-16, 15.5]$ in the baseline H.263, and motion vectors are allowed to point outside the picture boundaries.

H.263+ is backward compatible with H.263, which is approved as a standard in January 1998. Besides five standardized picture formats supported by H.263, H.263+ can also support custom picture formats, which makes H.263+ suitable for a wide range of video applications. H.263+ provides 12 new optional coding modes to improve the performance of H.263. One major improvement of H.263+ is that it provides temporal, SNR, and spatial scalability. Scalability is very desirable in heterogeneous environments, which can easily adjust the frame rates, bit rates, and resolutions of the encoder's output bitstream to meet the requirements of the decoder. H.263+ also contains four error-resilience-oriented modes, such as forward error-correction mode, slice-structured mode, etc., to improve the quality of video transmission in error-prone environments [12]. H.263++ adds three more optimal modes to H.263, which was approved in November of 2000. The data partitioned slice mode provides enhanced error resilience by separating header and motion vector data from DCT coefficients data in the bitstream.

2.1.3 MPEG-1, MPEG-2 and MPEG-4

The ISO SC29 WG11 “Moving Picture Experts Group” (MPEG) was established in 1988. MPEG-1 [13, 14] is the first video coding standard developed by MPEG committee, which was published in 1993. MPEG-1 was developed for storage of moving pictures and its associated audio at about 1.5 Mbps on digital storage media such as CD-ROM. With MPEG-1, one CD-ROM disk can store up to 72 minutes of video and audio with VHS (Video Home System) quality. The basic coding algorithm of MPEG-1 is very similar to H.261, including 16×16 macroblock structure, motion compensation, 8×8 DCT transform, etc. Different from H.261, B-pictures are introduced in MPEG-1 to improve the compression efficiency by utilizing both past and future frames as reference frames to perform the motion compensation. In addition, MPEG-1 introduces a quantization scale parameter to provide spatial-adaptive weighting quantization.

MPEG-2 [15] is so far the most extensively used video coding standard. One of the main applications for MPEG-2 is digital TV including standard TV and high-definition TV (HDTV). The target bit rate of MPEG-2 is about 2-30 Mbps. MPEG-2 is designed to be backward compatible to MPEG-1, i.e., every MPEG-2 compatible decoder can decode MPEG-1 bitstreams. MPEG-2 defines many profiles and levels to support various applications. Basically, a profile defines a set of coding algorithms that can be used to generate conforming bitstreams, while a level defines the range of the key parameters. Compared with MPEG-1, MPEG-2 improves the coding efficiency and incorporates more functionalities. Moreover, MPEG-2 supports not only progressive videos but also interlaced videos. In MPEG-2, a video picture in an interlaced video can be encoded as either a field picture or a frame picture with either frame-based DCT or field-based DCT, which significantly improves the coding efficiency of interlaced video sequences. Another main difference between MPEG-1 and MPEG-2 is that MPEG-2 provides three scalable video coding schemes including SNR scalability, spatial scalability, and temporal scalability.

In addition to frame-based functionalities of the MPEG-1 and MPEG-2, MPEG-4 [1, 16] supports object-based coding and manipulation of “objects” within video scenes. MPEG-4 can support variable block size motion compensation and the motion compensation can be with the quarter-pel accuracy. The elements of a video scene can come from three fields: digital television, interactive graphics, and the world wide web. MPEG-4 can offer efficient coding at bit rates between 5 kbps and 15 Mbps or higher. In addition, MPEG-4 can provide content-based interactivity, error resilient tools, scalability and etc.

2.1.4 H.264

H.264 [2, 17, 18] is the latest international video coding standard. The original name of H.264 is H.26L, which was started by ITU-T Video Coding Experts Group (VCEG) in 1998. In the late 2001, VCEG and MPEG formed the Joint Video Team (JVT) to create a single video coding standard for the new ITU-T Recommendation H.264 and a new part of MPEG-4 (MPEG-4 part-10 or MPEG-4 AVC).

Compared with the previous video coding standards, H.264 introduces many new features. In particular, H.264 supports variable block-size motion compensation, where the minimum block size can be as small as 4×4 . It also allows the encoder to select multiple reference frames for better motion estimation. Furthermore, instead of using DCT, an integer transform is applied to each 4×4 blocks, which avoids the common inverse-transform mismatches caused by floating point computation [19]. In addition, an adaptive in-loop deblocking filter [20] is introduced in the motion-compensated prediction loop, which greatly reduces the “blocking” artifact and improves the inter-picture prediction efficiency.

Another important component of H.264 is the advanced entropy coding. There are three entropy coding methods provided in H.264: universal variable length codes (UVLC), context adaptive variable length coding (CAVLC), and context adaptive binary arithmetic coding (CABAC) [21]. All the syntax elements are encoded with UVLC while the

quantized transform coefficients can be encoded with either CAVLC or CABAC. The basic idea of CAVLC is to use multiple VLC tables and adaptively select the optimal VLC table according to the context. On the other hand, CABAC is similar to CAVLC but based on adaptive arithmetic coding. Compared with CAVLC, CABAC can achieve 5% – 15% bit rate reduction.

With all these advanced features, H.264 has achieved significant improvement in rate-distortion efficiency. Compared with H.263 or MPEG-4 Simple Profile, H.264 can provide up to 50% bit rate savings for equivalent perceptual quality. Besides the improvement in coding efficiency, H.264 also introduces a network abstraction layer (NAL) to adapt H.264 bitstreams to a variety of networks.

2.2 Scalable Video Coding Techniques

In today's video communication systems, video sequences will be transmitted through different networks with different channel capacity. The devices used to receive and decode the video sequences may have different computational ability, ranging from wireless handheld devices to high performance computers. Therefore, for a video sequence, only one bitstream coded at a certain bit rate and stored at the servers cannot meet diverse requirements. Scalable video coding [5] is a common approach to provide video adaptation to heterogeneous scenarios. With scalable video coding, a video signal only needs to be encoded once with highest resolution, and can be decoded into different resolutions to meet different application requirements. Due to the advantages of flexibility and low storage space cost, scalable video coding has become more and more popular in video streaming systems. In the past few years, several scalable video coding schemes [22, 23, 24, 25] have been proposed for video transmission. In these scalable video coding schemes, typically, a video sequence is coded into a base layer and several enhancement

layers. A better video quality can be obtained only when an enhancement layer is completely received and decoded.

2.2.1 Scalable Video Coding in MPEG-2

MPEG-2 is the first video coding standard that provides the scalability mechanisms [25]. MPEG-2 utilizes layered coding technique to generate one base layer and several enhancement layer bitstreams. The base layer bitstream can be independently decoded to provide the coarse video quality. However, the enhancement layer bitstreams can only be decoded to provide enhanced video quality when the base layer bitstream is available. MPEG-2 provides four methods to generate a layered bit stream: data partitioning, signal-to-noise ratio (SNR) scalability, spatial scalability, and temporal scalability.

Data partitioning operates at the system level and it is a frequency domain method. Based on the priority breakpoint (PBP) value, which indicates the maximum number of block transform coefficients contained in the base layer bitstream, data partitioning separates a non-layered bitstream into two layered bitstreams. Since data partitioning does not change the low level coding decisions, the overhead introduced by data partitioning is negligible. When only the partial information is used to reconstruct the reference frame in the decoder, drift problem will appear since the reference frame is not identical to that used in the encoder. However, drift can be reduced by using frequent intra-coded frames.

SNR scalability is a technique that encodes a video sequence at the same temporal resolution and spatial resolution, but with different video quality. A SNR scalable video encoder is shown in Fig. 2.1. In the figure, the base layer is encoded in the same way as the non-layered encoder. The resulting error, which is formed by subtracting the de-quantized DCT coefficients of the base layer from the original input DCT block, is re-quantized with an enhancement layer quantizer to form the enhancement layer bitstream. The base layer bitstream contains all the motion vectors information and no motion vectors are included in the enhancement layer bitstream.

CHAPTER 2. VIDEO CODING AND TRANSMISSION TECHNIQUES

Figure 2.1: SNR scalable video encoder.

Spatial scalability is a technique to code a video sequence into layers at different spatial resolution, but the same temporal resolution. The base layer is coded at lower spatial resolution, and the reconstructed base layer frame is up-sampled to form the prediction for the encoding of the enhancement layer. In MPEG-2, the enhancement layer encodes the difference between the original video frame and the prediction, which is a weighted average of the previous enhancement layer frame and the up-sampled base layer frame.

For temporal scalability, the base layer is encoded in the normal way but with lower frame rate. The enhancement layer provides the missing frames to form the higher frame rate. Different from the base layer, the enhancement layer can perform the motion compensation by utilizing either a previously decoded picture in the enhancement layer or the reconstructed frames in the base layer as the reference frames. The spatial resolution of each layer is the same as that of the input video sequence.

In all these MPEG-2 layered scalable video coding methods, in order to provide enhanced video quality, an entire enhancement layer needs to be received and decoded. If only partial of an enhancement layer is correctly received, there is no improvement for the video quality.

2.2.2 FGS Video Coding Scheme

FGS (Fine Granularity Scalability) [26] is a new scalable coding scheme adopted in MPEG-4. The basic idea of FGS is to encode a video sequence into two layers: a base layer and an enhancement layer. The base layer is coded without scalability while the enhancement layer is coded bitplane by bitplane. By adopting the bitplane coding method in the enhancement layer, the FGS enhancement layer bitstream can be truncated in any position, which provides fine granularity scalability. There is no prediction in the FGS enhancement layer, which limits any channel impairment within one frame and prevents successive frames from error propagation at the cost of decreasing coding efficiency. Fig. 2.2 shows the FGS encoder structure. Compared with traditional scalable coding schemes, FGS provides an enhancement layer which is scalable with fine granularity. In other words, the reconstructed video quality is proportional to the decoded number of bits in the enhancement layer. In addition, an extension of FGS, called FGST (FGS temporal scalability), was proposed to provide temporal scalability [27]. It has been demonstrated that FGS can provide flexibility and good performance for Internet video streaming [26, 28]. However, FGS is not suitable for wireless video transmission. This is mainly because the low coding efficiency of FGS does not fit the limited bandwidth of wireless networks.

To improve the coding efficiency of FGS, many schemes [29, 30, 31, 32, 33] have been proposed. In [30], a motion compensation FGS (MC-FGS) was developed, which includes two classes, two-loop MC-FGS and single-loop MC-FGS, according to the number of MC loops introduced. The two-loop MC-FGS introduces a MC loop in the enhancement layer to remove the temporal redundancy. While the MC prediction is restricted to the enhancement layer of B frames. In the single-loop MC-FGS, both the base layer and the enhancement layer are used for the base layer prediction. The single-loop MC-FGS achieves higher coding efficiency since high-quality reference frames are employed in the

CHAPTER 2. VIDEO CODING AND TRANSMISSION TECHNIQUES

Figure 2.2: FGS encoder structure.

base layer, but has lower robustness since errors occurring in the enhancement layer affect the performance of the base layer. Schaar et al. further proposed an adaptive motion compensation FGS scheme (AMC-FGS) [31], which can adaptively choose the most suitable FGS structure based on the device capability and channel condition. Similarly, Peng introduced a mode-adaptive FGS scheme in [32], which adaptively uses the previous reconstructed enhancement layer macroblock, the current reconstructed base layer macroblock or the average of them to form the reference macroblock for the prediction of the current enhancement layer.

The progressive FGS (PFGS) scheme was proposed in [29], which encodes video frames into multiple layers and uses several high-quality references for the predictions in the enhancement layer encoding. In addition, the PFGS keeps a prediction path from the base layer to the the highest layers across several layers to have the property of graceful error and loss recovery. Since the original PFGS only chooses its reference at the frame level, Sun et al. proposed an effective scheme to control the PFGS coding at the macroblock level, which can achieve better trade-off between coding efficiency and drift errors than the original PFGS [34] .

Another novel technique, the robust FGS (RFGS), was proposed in [33] to further

improve the temporal prediction at both the enhancement layer and the base layer by utilizing two parameters, the number of bitplanes and the amount of predictive leak, to control the construction of reference frames. As the number of bitplanes increases, the coding efficiency will be improved, but the drift errors will occur when the bitplanes used for the reference in the encoder cannot be received in the decoder. The leaky predication factor can be used to speed up the attenuation of the accumulated errors. Therefore the corresponding selection of these two parameters can be used to provide the tradeoff between the coding efficiency and drift errors. In this dissertation, we denote such leaky prediction [33, 35, 36] based FGS coding as L-FGS.

2.2.3 Scalable Extension of H.264/MPEG-4 AVC

The scalable extension of H.264/MPEG-4 AVC was selected as the first working draft for the MPEG's scalable video coding standardization project in January 2005 [37]. The corresponding reference encoding process of this working draft is described in the Joint Scalable Video Model (JSVM). The basic idea of JSVM is to extend the H.264/AVC hybrid video coding towards motion-compensated temporal filtering (MCTF) by utilizing the lifting framework [38]. Thus, the motion compensation features of H.264, which is one of the major reasons that H.264 achieves much better coding efficiency than previous coding standards, can be re-used in the MCTF prediction and update steps. To be compatible with H.264/AVC, hierarchical B pictures can also be utilized to provide the temporal scalability with hierarchical temporal decomposition [39]. The difference between hierarchical B and MCTF is that MCTF has the extra update step. A typical hierarchical prediction structure with four dyadic hierarchy stages is depicted in Fig. 2.3. In order to support several temporal scalability levels, it is required that only pictures of a coarser or the same temporal level can be utilized as reference frames for motion prediction. Since pictures that are used as reference frames have to be encoded first, the coding order is different from the display order as shown in Fig. 2.3.

CHAPTER 2. VIDEO CODING AND TRANSMISSION TECHNIQUES

Figure 2.3: The hierarchical coding structure with four temporal levels.

The temporal scalability can be achieved by removing those pictures that are not reference pictures for the remaining pictures. The spatial scalability is achieved by using a pyramid of spatial resolutions and each spatial resolution is coded using a new layer. Considering the dependencies between the spatial resolution layers, three inter-layer prediction mechanisms are exploited to improve the coding efficiency of the enhancement layers, which include inter-layer intra texture prediction, inter-layer motion prediction and inter-layer residual prediction [40]. For SNR scalability, fine grain SNR scalability is achieved by encoding successive refinements of the transform coefficients [40].

2.3 Rate-Distortion Theory

Rate distortion theory is a branch of information theory, which gives theoretical performance bounds for lossy compression schemes. The basic problem in rate distortion theory is that: what is the minimum bit rate required to encode a source in order to achieve a certain distortion? The rate distortion theory provides a rate-distortion function $R(D)$ which has the following properties [41]:

Figure 2.4: An example of R-D curve.

- For any given distortion D , it is possible to design a coding scheme which can obtain average distortion arbitrarily close to D with rate arbitrarily close to $R(D)$.
- There is no practical compression system that can achieve distortion D or below D at a bit rate below $R(D)$.

Fig. 2.4 shows the curve of a theoretic R-D function, which is convex, continuous and strictly decreasing. However, a practical R-D curve may not be strictly decreasing.

2.4 Rate Control Techniques

For video coding and transmission, the two most important factors are coding bit rate R and picture quality. The widely used measure for picture quality is the mean-squared error (MSE) between the original video and the reconstructed one, which is denoted as

D. Since the channel bandwidth is often limited and time-varying, optimal rate control is very important in order to achieve good video quality.

Generally, there are two kinds of approaches for the R-D modeling: an analytical model-based approach and an operational rate-distortion based approach [42, 43]. In the analytical model-based approach, the analytical model for the whole coding system is derived by first decomposing the coding system into components whose statistical models are known and then combining these models together. In an operational rate-distortion based approach, R-D functions are constructed by processing the current available R-D data. The bit allocation strategy is then determined by pre-analyzing the R-D characteristics of the future frames. Generally, this kind of approach is of high computational complexity.

In typical transform coding, both R and D are controlled by the quantization parameter (QP). Technically speaking, the major issue in rate control is how to achieve the target coding bit rate accurately by properly setting the quantization parameters. This requires accurate rate-quantization (R-Q) functions and distortion-quantization (D-Q) functions. In the past years, many rate control schemes have been proposed in existing standards, such as TM5 in MPEG2 [15], VM8 in MPEG-4 [1] and TMN8 in H.263+ [44].

2.4.1 TM5 Rate Control Algorithm

The TM5 rate control scheme is a GOP (Group of Pictures) based bit allocation scheme, which is designed for rate control in MPEG-2 video coding. Firstly, a target bit rate is calculated for each GOP. Within the GOP, bits are allocated to each frame according to its picture type and the global complexity measures of the previous frames. Within the frame, the quantization parameter for each macroblock is calculated and adaptively modulated based on the target bit rate allocated for the current frame, the virtual buffer fullness and the spatial activity of this macroblock. The detailed algorithm is as follows.

For each type of picture (I, P, or B), the corresponding “global complexity measure” (X_i , X_p , or X_b) is maintained. After encoding a picture, (X_i , X_p , or X_b) is updated as follows:

$$X_i = S_i \times Q_i, \quad (2.1)$$

$$X_p = S_p \times Q_p, \quad (2.2)$$

$$X_b = S_b \times Q_b, \quad (2.3)$$

where S_i , S_p , S_b are the number of bits generated by encoding this picture, and Q_i , Q_p , Q_b are the average quantization parameter of all the macroblocks. The target number of bits allocated for the next picture in the GOP is computed as:

$$T_i = \max \left\{ \frac{R}{1 + \frac{N_p X_p}{X_i K_p} + \frac{N_b X_b}{X_i K_b}}, \frac{bit_rate}{8 \times picture_rate} \right\}, \quad (2.4)$$

$$T_p = \max \left\{ \frac{R}{N_p + \frac{N_b K_p X_b}{K_b X_p}}, \frac{bit_rate}{8 \times picture_rate} \right\}, \quad (2.5)$$

$$T_b = \max \left\{ \frac{R}{N_b + \frac{N_p K_b X_p}{K_p X_b}}, \frac{bit_rate}{8 \times picture_rate} \right\}, \quad (2.6)$$

where K_p and K_b are constants, R is the remaining number of bits allocated to the current GOP, and N_p and N_b are the numbers of P and B pictures remaining in the current GOP. In addition, three virtual buffers are maintained for the corresponding class of coded picture. Based on the current status of buffer fullness and target bit rate for the picture, the quantization parameter is calculated for each macroblock. Finally, the chosen quantization parameter is further modulated according to the spatial activity of the macroblock. The quantization parameter will be increased if there is a high variance in the four luminance blocks of the macroblock.

2.4.2 TMN8 Rate Control Algorithm

TMN8 rate control algorithm is developed to perform the rate control in the DCT video coders for low-delay video communications [45]. The R-Q (Rate-Quantization) model

utilized in TMN8 rate control algorithm is as follows:

$$R(Q) = \begin{cases} \frac{1}{2} \log_2(2e^2 \frac{\sigma^2}{Q^2}), & \frac{\sigma^2}{Q^2} > \frac{1}{2e} \\ \frac{e}{\ln 2} \frac{\sigma^2}{Q^2}, & \frac{\sigma^2}{Q^2} \leq \frac{1}{2e}, \end{cases} \quad (2.7)$$

where σ^2 is the variance of the motion-compensated difference frame pixels. Let Q_i denote the quantization parameter for the i -th macroblock, the distortion model utilized in TMN8 is

$$D = \frac{1}{N} \sum_{i=1}^N \alpha_i^2 \frac{Q_i^2}{12}, \quad (2.8)$$

where N is the number of macroblocks in a frame and α_i is the distortion weight of the i -th macroblock.

For the purpose of minimizing the distortion based on the constraint of the total number of bits, the optimal quantization parameter to encode the i -th macroblock is derived in TMN8 with the Lagrange theory, which is given by

$$Q_i = \sqrt{\frac{AK}{B - ANC} \frac{\sigma_i}{\alpha_i} \sum_{k=1}^N \alpha_k \sigma_k}, \quad (2.9)$$

where A is the number of pixels in a block, K is equal to $e/\ln 2$, B is the total number of bits, and C is the bit rate used to encode the motion vectors and the header information.

2.4.3 MPEG-4 Rate Control Algorithm

Different from other frame-based video coding standards, MPEG-4 is an object-based video coding standard. In MPEG-4, each frame is considered as a composition of several arbitrarily shaped video objects. The single video object (SVO) rate control algorithm for MPEG-4 video coding was proposed in [46, 42], which is based on a second-order rate distortion model, i.e.,

$$R - H = \frac{X_1 \cdot M}{Q} + \frac{X_2 \cdot M}{Q^2}, \quad (2.10)$$

where R is the total number of bits used for encoding the current frame, H is the bits used for header, motion vectors and shape information, M is mean absolute difference

CHAPTER 2. VIDEO CODING AND TRANSMISSION TECHNIQUES

(MAD) which is calculated after motion compensation for the Y component, Q is the quantization parameter used for the current frame, and X_1 , X_2 are the first- and the second-order coefficients. By introducing the video coding complexity, i.e., MAD, this model becomes more scalable with the video contents. Based on the assumption that the current frame has similar rate and distortion curves to the previous frames, the model parameters X_1 and X_2 can be calculated with the linear regression technique and the selected data points from previous frames [47].

In [48], the above SVO rate control algorithm is further extended to multiple video objects (MDO). The employed R-D model is similar to that in SVO rate control and is given as follows:

$$R_i - H_i = \frac{X_{1,i} \cdot M_i}{Q_i} + \frac{X_{2,i} \cdot M_i}{Q_i^2}, \quad (2.11)$$

where i denotes the index of the video object, R_i is the target number of bits for the i -th object, and H_i is the amount of header, motion and shape bits used for the i -th object.

In order to improve the perceptual video quality, each video object should be encoded with different quality. Therefore, it is necessary to optimally allocate the total number of bits among the video objects. In [48], three measures are chosen for the bit allocation, i.e., size, motion, and variance. The target number of bits for object i is given by

$$R_i = R \cdot (\omega_s SIZE_i + \omega_m MOT_i + \omega_v VAR_i), \quad (2.12)$$

where R is the total number of bits for the frame, $SIZE_i$, MOT_i and VAR_i are the size, motion, and variance of object i . The weights $\omega_s, \omega_m, \omega_v \in [0, 1]$ satisfy the constraint $\omega_s + \omega_m + \omega_v = 1$.

2.4.4 H.264 Rate Control Algorithm

The rate control for H.264 is more difficult than the previous video coding standards. Particularly, the quantization parameter is used in both the rate control algorithm and

the rate-distortion optimization (RDO) algorithm [49]. On one hand, RDO requires the pre-determined quantization parameter to select the optimal coding modes and motion vectors. On the other hand, the rate control algorithm selects the appropriate quantization parameter only after the coding modes and motion vectors have been determined by the RDO. Therefore, this results in a chicken and egg dilemma [50]. Recently, many rate control schemes have been proposed for H.264 to solve the above problem [50, 51].

In [50], a linear prediction model is proposed to estimate the MAD of a basic unit in the current frame. Let MAD_{cb} denote the predicted MAD of the current basic unit in the current frame, and MAD_{pb} denote the actual MAD of the basic unit at the co-located position of the previous frame, the linear prediction model is given by

$$MAD_{cb} = a_1 \cdot MAD_{pb} + a_2, \quad (2.13)$$

where a_1 and a_2 are two coefficients of the prediction model. The initial value of a_1 and a_2 are set to 1 and 0, and they are updated after encoding each basic unit. The basic unit can be a frame, a slice or a MB. The target bit rate allocated to encode the current frame is calculated according to the target buffer level, the frame rate, the available channel bandwidth and the actual buffer occupancy. The target number of bits is further allocated to each basic unit. Based on the quadratic rate model proposed in [42], the quantization parameter can be calculated with the MAD predicted by Eq. (2.13). The obtained quantization parameter is further adjusted to smooth the visual quality. After that, RDO is performed and the current basic unit is encoded.

2.5 Error Control Techniques

Besides the bandwidth limitation, there exists severe error corruption in wireless networks. Errors occurring in wireless channels range from random errors to burst errors or even an intermittent loss of connection [52]. Under such an error-prone environment,

highly compressed video data become extremely vulnerable [53, 54]. A single bit error may cause severe error propagation and thus result in great performance degradation. Besides the error corruption in the physical layer, there also exists the packet loss problem in the network layer of all the packet-switch networks due to router buffer overflow. Therefore, error control techniques [55, 56] are usually used to combat channel errors to ensure a reliable video transmission.

2.5.1 FEC Coding

One of the classic techniques to combat transmission errors or packet loss is Forward Error Correction (FEC) codes [57]. FEC can achieve good performance in both error detection and error correction. However, applying FEC results in increasing transmission overhead, or reducing the available bandwidth for the payload data even when the channel is in good states.

There are two main categories of FEC: block coding and convolutional coding. Convolutional coding can work on arbitrary length input. In contrast, block coding can only work on fixed-size blocks of bits or symbols. There are many types of block codes, and the Reed-Solomon (RS) code is the one being widely used. A RS code can be denoted as $RS(N,K)$ with m bits per symbol. It means that the coder takes blocks of K symbols as input and adds $N - K$ parity symbols to make an N -symbol codeword. The maximum block length is $N_{max} = 2^m - 1$. A RS code can correct up to $\lfloor \frac{N-K}{2} \rfloor$ error symbols if the locations of the error symbols are not known. If the error locations are known in advance, i.e., symbol erasures, the RS code can correct up to $\lfloor N - K \rfloor$ erasures.

2.5.2 Automatic Repeat Request

Automatic Repeat Request (ARQ) [58] is an error control technique in which the receiver automatically sends a retransmission request to the transmitter when it detects

transmission errors or packet loss. Compared with FEC, ARQ needs a feedback channel between the receiver and the transmitter. On the other hand, the appropriate choice of FEC length depends on the accurate estimation of the channel error rate or packet loss rate before the transmission, but ARQ can adaptively transmit the lost packets according to the channel condition. There are three basic ARQ schemes: Stop and Wait (SW), Go Back N (GN), and Selective Repeat (SR). One critical problem of ARQ is the delay. A simple method to reduce the delay is to limit the maximal number of retransmissions at the cost of reliability. ARQ and hybrid ARQ, which is the combination of ARQ and FEC, have been shown to be more effective than FEC and can provide higher throughput because it can effectively adapt to the varying channel conditions [59, 60].

2.5.3 Motion-Compensated Error Concealment

Due to the temporal correlation in the adjacent video frames, the information in the previous decoded frames can be employed to correct the error macroblocks in the current frame. The straightforward way for error concealment is to simply replace the damaged macroblock with the spatially corresponding macroblock in the previous frame. However, this simple method may produce unacceptable visual artifacts when the motion is very large. Considering the motion information, significant improvement can be achieved by replacing the damaged macroblock with the motion compensated block, i.e., the block pointed by the motion vector of the damaged block. In the case that the motion vector of the damaged block is lost, it can be estimated as the median of the motion vectors of the nearest three macroblock in the preceding GOB (group of blocks) [61]. If the previous GOB is lost, the estimated motion vector can be set to zero.

2.5.4 Layered Video Coding with Transport Prioritization

In layered video coding, one video sequence is encoded into several layer bitstreams, i.e., the base layer bitstream and the enhancement layer bitstreams [25], which has been

described in Section 2.2. The base layer bitstream can be decoded independently and provide coarse yet acceptable video quality. The enhancement layer bitstreams can only be decoded and provide a higher video quality when the base layer bitstream is completely received and decoded.

The layered video coding structure is very suitable for the video transmission over multi-priority networks, such as an ATM network [25]. To combat the channel impairment, the base layer bitstream can be transmitted with higher priority or higher degree of error protection than the enhancement layer bitstreams.

2.5.5 Multiple Description Coding

Although layered coding works well for the networks with priority channels such as using DiffServ, it is highly inefficient for the networks without priority channels because of the dependency between layers. In addition, layered video coding requires the base layer to be delivered almost error-free, which is not feasible for many applications. For the lossy networks without priority, strong FEC or retransmission has to be applied in the application layer for the base layer bitstream. However, retransmission is not acceptable for real-time interactive applications such as video conferencing since it will cause long delay. On the other hand, strong FEC protection costs large bandwidth and is not effective in bandwidth-limited channels.

Different from layered coding, multiple description coding (MDC) encodes a video sequence into two or more equal important descriptions [62, 63, 64]. Each description can be decoded independently and can provide low but acceptable quality. Higher video quality can be achieved if more descriptions are correctly received and decoded. The good error resilience of MDC is achieved at the cost of more redundancy. Specially, in order to achieve the same quality as a conventional single description (SD) coder in the absence of transmission errors, the multiple description (MD) coder needs to use more

bits. Since motion-compensated prediction can achieve high coding efficiency, it has been incorporated in many MD video coders. However, when the signal used for prediction in the encoder is not available in the decoder due to the transmission loss, mismatch will occur. Hence, the challenge in MD coding design is to balance prediction efficiency and mismatch control, which are the two fundamental design issues for MD video coders.

One simple implementation of MDC is the multiple state encoder [63], which encodes the video into multiple independently decodable streams, each with its own prediction process and state. In [64], the pairwise correlation transform (PCT) is utilized to introduce controlled correlation between two descriptions, which can help reduce the distortion when only one description is received.

2.5.6 Joint Source and Channel Coding

The objective of source coding is to compress the data by removing the redundancy as much as possible, while channel coding introduces the redundancy in order to protect the data from error corruption. According to Shannon's theorem, source and channel coding can be designed separately without loss of the overall performance. However, Shannon's source and channel coding separation theorem is based on three assumptions: ergodic channel, infinite delay, and point to point communications, which are not realistic. Therefore, for practical systems, source coding and channel coding need to be jointly considered in order to achieve optimal performance. This is referred as joint source and channel coding (JSCC) [65, 53, 66].

One type of JSCC schemes is error-resilient source coding, where the source coding does not aim to maximally remove the redundancy since the redundancy can be used to combat the transmission errors to compensate the inadequacy of the channel coding methods [65]. The mode selection is a good example. In particular, mode selection is to determine the optimal prediction mode to encode each block in a video frame. In

CHAPTER 2. VIDEO CODING AND TRANSMISSION TECHNIQUES

general, there are two basic coding modes, intra-coding and inter-coding. Inter-coding usually improves the coding efficiency while the generated bitstream is very sensible to the channel errors, and the errors in one frame will propagate to future frames along the prediction path. In contrast, intra-coding is more error-resilient to the error propagations but the coding efficiency is low. Traditional mode selection schemes such as [49] are purely optimized for improving the coding efficiency of source coding. In the context of JSCC, the mode selection schemes such as [53, 61] are optimized for improving the end-to-end video quality under noisy channels.

Another type of JSCC schemes is with FEC or ARQ. For many video coding schemes, such as layered video coding, some parts of the generated video bitstreams are more important than others. Thus, the bitstream can be applied with unequal error protection (UEP) [67, 68], i.e., the more important information is given more protection. Compared with equal error protection (EEP), UEP can obtain considerable performance gain and has the property of graceful performance degradation during channel mismatch cases. However, the complexity of UEP is much higher than that of EEP since it is not trivial to find the optimal UEP solution.

Once the source coding scheme and the channel coding scheme have been chosen, the main problem in JSCC is how to optimally allocate the available bits between the source coder and the channel coder in order to achieve the minimum distortion [69]. In order to perform the optimal bit allocation, an accurate estimation of the end-to-end distortion is very important. Many end-to-end distortion estimation schemes have been proposed in the past several years [54, 61, 53]. In [54], a distortion model for the video transmission over lossy channels was proposed, which includes the rate-distortion performance of the video encoder, forward error correction, intra coding rate, and spatial filtering. In this model, many parameters need to be estimated by the off-line analysis. Hence, it is not suitable for real-time video encoding. Zhang et al. proposed an accurate recursive

CHAPTER 2. VIDEO CODING AND TRANSMISSION TECHNIQUES

optimal per-pixel estimate (ROPE) algorithm in [61], which can accurately estimate the end-to-end distortion at the pixel level at the cost of high computational complexity. He et al. proposed a statistical model in [53], which considers the effects of the MB intra refreshing rate and error propagation among inter-frames. The proposed model has low computational complexity since the estimation is at the frame level. However, a feedback channel is required in order to achieve an accurate estimation.

Chapter 3

Rate-Distortion Analysis of Leaky Prediction Based FGS Video for Constant Quality Constrained Rate Adaptation

In this chapter, we study the optimal bit allocation for the applications of streaming pre-encoded videos. Specially, we consider the constant quality constrained bit allocation for leaky prediction based FGS (L-FGS). The challenge lies in obtaining accurate R-D (rate-distortion) curves for L-FGS due to the dependency among video frames.

This chapter is organized as follows. Section 3.1 describes the related work on constant quality constrained bit allocation for FGS video transmission. Section 3.2 gives an overview of the leaky prediction based FGS. Section 3.3 presents our proposed R-D model for L-FGS and the corresponding simulation results. Based on the R-D model, Section 3.4 describes the proposed constant quality constrained bit allocation for L-FGS and shows the experimental results to demonstrate that our proposed scheme can effectively minimize the variation in video quality. Finally, Section 3.5 gives the summary of this chapter.

3.1 Related Work

Although rate adaptation can be easily realized in FGS video coding by bit truncation, there is a fundamental problem, i.e., how to truncate bits given the channel bandwidth. The simplest method is the uniform bit truncation [26]. However, this will cause low coding efficiency and video quality fluctuation among adjacent frames due to the non-stationary property of video signals, i.e., the flickering artifact. Therefore, it is desired to have optimal bit truncation/allocation among video frames in order to achieve constant video quality and improve the coding efficiency. Note that constant quality constrained bit allocation on the other hand will cause rate fluctuation which can be solved by carefully choosing the buffer size and the pre-loading time in video streaming applications.

Related work on constant quality constrained bit allocation can be traced back to the original VBR (variable bit rate) single-layer video coding [70], where it was roughly assumed that using the same quantization factor throughout the entire video sequence can achieve the consistent video quality. Later on, the performance of VBR video coding was further improved by using multiple-pass video coding [71, 72, 73].

Constant quality constrained bit allocation for FGS video coding has also been well studied and many schemes [74, 75, 76] have been proposed in the past few years. All these optimal bit allocation schemes are based on explicit R-D analysis, where the R-D relationship for each video frame is obtained through either empirical or modelling approaches. In [74], Zhang et al. constructed accurate R-D curves by extracting some R-D points such as the end points of each bitplane during the offline encoding process followed by linear interpolation between neighbor R-D points. Based on the obtained R-D curves, a sliding window bit allocation scheme was proposed in [74], which can completely fulfill the aim of constant quality. In [77], Dai et al. proposed a closed-form R-D model through analyzing the properties of the input to the MPEG-4 FGS enhancement layer, which significantly outperforms the common distortion models such

CHAPTER 3. RATE-DISTORTION ANALYSIS OF LEAKY PREDICTION BASED FGS VIDEO FOR
CONSTANT QUALITY CONSTRAINED RATE ADAPTATION

as the quadratic model. Zhao et al. [75] also proposed a constant quality rate control scheme for MPEG-4 FGS video. Compared with [74], the work in [75] performed the constant quality rate control for both the base layer and the enhancement layer.

However, constant quality constrained bit allocation for L-FGS is still an open problem. The difficulty lies in obtaining accurate R-D curves for L-FGS due to the dependency among video frames. For example, the R-D curve of the n -th frame may depend on the quality of the $(n - 1)$ -th reference frame, which can only be determined during the streaming stage, i.e., given the available bandwidth at that time. A possible solution is to apply those frame-level bit allocation schemes originally designed for single-layer video coding for the enhancement layer bit allocation in L-FGS such as in [76], where a common exponential model was adopted for bit allocation in PFGS. These schemes are basically approximate approaches and work fine at high bit rates or with a short GOP structure. However, for low bit rate video coding and with long frame dependency [7], directly applying these approximate approaches will have intolerable accumulated errors.

In [53], the authors have proposed a model to estimate the distortion propagated from the reference frame due to channel errors for non-scalable video coding. In this chapter, we modify this model for estimating the propagated distortion due to the rate adaptation of the reference frame in L-FGS. Considering the motion compensation process, intuitively, the propagated distortion should be correlated with the distortion introduced in the current frame. Thus, instead of simply assuming the independency of the two distortions, we add in this correlation to the overall R-D model, which greatly improves the accuracy of the proposed R-D model. Based on our proposed R-D model, a sliding window technique is further developed to adapt to the current available bandwidth under the constant quality constraint. Experimental results show that the proposed R-D model is very accurate and the corresponding bit allocation algorithm can achieve much smoother video quality than the traditional uniform bit allocation under both CBR and VBR channels.

3.2 Overview of Leaky Prediction Based FGS

Using symbol definitions in Table 3.1, in a typical L-FGS system, the base layer residue $e_B(n, i)$ is obtained by

$$e_B(n, i) = F(n, i) - \hat{F}_B(n - 1, j), \quad (3.1)$$

where $\hat{F}_B(n - 1, j)$ is the motion-compensation reference frame. $e_B(n, i)$ and motion vectors are compressed into the base layer. The base layer reconstruction, $\hat{F}_B(n, i)$, is given by

$$\hat{F}_B(n, i) = \hat{F}_B(n - 1, j) + \hat{e}_B(n, i), \quad (3.2)$$

which is stored in the buffer for the encoding of the next frame. After the prediction from the base layer, the enhancement layer data can be represented as

$$\begin{aligned} F_E(n, i) &= F(n, i) - \hat{F}_B(n, i) \\ &= F(n, i) - \hat{F}_B(n - 1, j) - \hat{e}_B(n, i). \end{aligned} \quad (3.3)$$

In the baseline FGS [26], the enhancement layer data is directly compressed into the enhancement layer, i.e.,

$$e_E(n, i) = F_E(n, i). \quad (3.4)$$

However, in the L-FGS, the high quality reference frame is introduced in the enhancement layer with a leaky factor α_n , which can be written as

$$G_E(n, i) = \hat{F}_B(n, i) + \alpha_n \hat{F}_E^p(n, i), \quad (3.5)$$

where $\alpha_n \in [0, 1]$. Note that only the partial data of the reconstructed enhancement layer frame is used as the reference frame, similar to the partial prediction in [33]. This is for a better tradeoff between the coding efficiency and the robustness to drift errors.

CHAPTER 3. RATE-DISTORTION ANALYSIS OF LEAKY PREDICTION BASED FGS VIDEO FOR
CONSTANT QUALITY CONSTRAINED RATE ADAPTATION

Table 3.1: The symbol definitions.

$F(n, i)$	: The original value of pixel i in the n -th video frame.
$\tilde{F}(n, i)$	: The received high quality value of pixel i in the n -th video frame at the decoder.
$\hat{F}_B(n, i)$	: The base layer reconstruction value of pixel i in the n -th video frame in the base layer prediction loop at the encoder.
$G_E(n, i)$	: The high quality reference frame used in the enhancement layer prediction loop.
$e_B(n, i)$	: The base layer residue value of pixel i in the n -th video frame.
$\hat{e}_B(n, i)$	: The reconstructed base layer residue.
$F_E(n, i)$	: The enhancement layer data after the prediction from the base layer.
$\hat{F}_E(n, i)$	: The reconstructed enhancement layer data in the enhancement layer prediction loop at the encoder.
$\tilde{F}_E(n, i)$	: The reconstructed enhancement layer data at the decoder.
$\hat{F}_E^p(n, i)$	: The partial data of the reconstructed enhancement layer data, which is used in the enhancement layer prediction loop at the encoder.
$\tilde{F}_E^p(n, i)$	: the partial data of the reconstructed enhancement layer data, which will be used in the enhancement layer prediction loop at the decoder.
$e_E(n, i)$	: The enhancement layer residue after all the predictions, including the predictions from both the base layer and the enhancement layer.
$\hat{e}_E(n, i)$	: The reconstructed enhancement layer residue at the encoder.
$\tilde{e}_E(n, i)$	: The enhancement layer residue received at the decoder.

The redundancy is further removed by subtracting $\hat{e}_B(n, i)$ from the difference between $F(n, i)$ and $G_E(n - 1, j)$, and the resulted residue is compressed into the enhancement layer. This enhancement layer residue $e_E(n, i)$ can be expressed as

$$e_E(n, i) = F(n, i) - \hat{e}_B(n, i) - (\hat{F}_B(n - 1, j) + \alpha_{n-1} \hat{F}_E^p(n - 1, j)), \quad (3.6)$$

where the same motion vectors generated in the base layer encoding are applied to the enhancement layer prediction coding in order to avoid re-conducting the most time-consuming motion estimation process. Combining with Eq. (3.3), we simplify $e_E(n, i)$ as

$$e_E(n, i) = F_E(n, i) - \alpha_{n-1} \hat{F}_E^p(n - 1, j) \quad (3.7)$$

CHAPTER 3. RATE-DISTORTION ANALYSIS OF LEAKY PREDICTION BASED FGS VIDEO FOR
CONSTANT QUALITY CONSTRAINED RATE ADAPTATION

Correspondingly, the enhancement layer reconstruction at the encoder is

$$\hat{F}_E(n, i) = \alpha_{n-1} \hat{F}_E^p(n-1, j) + \hat{e}_E(n, i), \quad (3.8)$$

which will be stored in the buffer for the encoding of the next enhancement layer frame.

Similar as in [33], the high quality reference frames can also be employed in the base layer to further improve the coding efficiency of the base layer. However, in this chapter, we only consider introducing high quality reference frames in the enhancement layer, and keep the same encoding structure in the base layer as the baseline FGS. In addition, we assume that the bandwidth is always enough for transmitting the entire base layer and thus the truncation for rate adaptation only happens in the enhancement layer. H.263+ instead of MPEG-4 is employed to encode the base layer for simplicity. We only consider encoding video sequences with a pattern of one I-frame followed by all P-frames and TMN8 is used as the rate control scheme for the base layer. The L-FGS enhancement layer is encoded bitplane by bitplane, the same as that in the MPEG-4 FGS.

3.3 Rate-Distortion Analysis of L-FGS

3.3.1 Proposed R-D Model

Since we assume that the entire base layer of L-FGS can be transmitted without having any distortion, the overall distortion $D(n)$ between $F(n, i)$ and $\tilde{F}(n, i)$ is actually the same as the distortion between $F_E(n, i)$ and $\tilde{F}_E(n, i)$. Therefore, $D(n)$ can be expressed as

$$\begin{aligned} D(n) &= E\{[F_E(n, i) - \tilde{F}_E(n, i)]^2\} \\ &= E\{[F_E(n, i) - (\alpha_{n-1} \tilde{F}_E^p(n-1, j) + \tilde{e}_E(n, i))]^2\} \\ &= E\{[(F_E(n, i) - \alpha_{n-1} \hat{F}_E^p(n-1, j) - \tilde{e}_E(n, i)) + \\ &\quad \alpha_{n-1}(\hat{F}_E^p(n-1, j) - \tilde{F}_E^p(n-1, j))]^2\}, \end{aligned} \quad (3.9)$$

CHAPTER 3. RATE-DISTORTION ANALYSIS OF LEAKY PREDICTION BASED FGS VIDEO FOR CONSTANT QUALITY CONSTRAINED RATE ADAPTATION

where $E\{\}$ represents the average values over all pixels.

From Eq. (3.9), we can see that the distortion $D(n)$ comes from two sources, i.e., the distortion produced by truncating bits in the current enhancement layer frame, denoted as $D_I(n)$, and the distortion propagated from the previous frame due to the leaky prediction, denoted as $D_P(n-1)$. $D_I(n)$ and $D_P(n-1)$ are defined as

$$D_I(n) = E\{[F_E(n, i) - \alpha_{n-1}\hat{F}_E^p(n-1, j) - \tilde{e}_E(n, i)]^2\},$$

$$D_P(n-1) = E\{[\hat{F}_E^p(n-1, i) - \tilde{F}_E^p(n-1, i)]^2\}.$$

Similar to [53], we assume that

$$E\{[\hat{F}_E^p(n-1, j) - \tilde{F}_E^p(n-1, j)]^2\} = \rho_{n-1}D_P(n-1), \quad (3.10)$$

where ρ_{n-1} is a constant describing the motion randomness of a video scene. To determine ρ_{n-1} , we can choose K pairs of $\hat{F}_E^p(n-1, i)$ and $\tilde{F}_E^p(n-1, i)$, which are obtained by randomly allocating K different amounts of bits in the enhancement layer, with the constraint that the amount of the allocated bits in the enhancement layer is smaller than that for the partial leaky prediction. Let $\hat{F}_E^{p,k}(n-1, i)$ and $\tilde{F}_E^{p,k}(n-1, i)$ denote the k -th pair of $\hat{F}_E^p(n-1, i)$ and $\tilde{F}_E^p(n-1, i)$, ρ_{n-1} can be calculated as

$$\rho_{n-1} = \frac{1}{K} \sum_{k=1}^K \frac{E\{[\hat{F}_E^{p,k}(n-1, j) - \tilde{F}_E^{p,k}(n-1, j)]^2\}}{E\{[\hat{F}_E^{p,k}(n-1, i) - \tilde{F}_E^{p,k}(n-1, i)]^2\}}, \quad (3.11)$$

where j in the $(n-1)$ -th frame is the motion prediction pixel for the pixel i in the n -th frame.

Initially, we assumed that $D_I(n)$ and $D_P(n-1)$ are independent. That implies $D(n) = D_I(n) + \alpha_{n-1}^2 \rho_{n-1} D_P(n-1)$. To justify the assumption, we code the QCIF Akiyo sequence at 10 fps, 16 kbps for the base layer, 11 kbits for the enhancement layer partial prediction and $\alpha = 0.8$. In Fig. 3.1, we plot the $D(n)$ and $D_I(n) + \alpha_{n-1}^2 \rho_{n-1} D_P(n-1)$ for each frame. We find that $D(n)$ is actually not equal to $D_I(n) + \alpha_{n-1}^2 \rho_{n-1} D_P(n-1)$, which means

CHAPTER 3. RATE-DISTORTION ANALYSIS OF LEAKY PREDICTION BASED FGS VIDEO FOR
CONSTANT QUALITY CONSTRAINED RATE ADAPTATION

Figure 3.1: Comparison between the overall distortion $D(n)$, the solid line with “+” marks, and $D_I(n) + \alpha_{n-1}^2 \rho_{n-1} D_P(n-1)$, the solid line with “*” marks. Left: QCIF Akiyo at 48 kbps. Right: QCIF Akiyo at 80 kbps.

the assumption of the independency is not correct. Considering the dependency between $D_I(n)$ and $D_P(n-1)$, through extensive experiments over different video sequences and different encoding settings, we find that $D(n)$ can be approximately composed as

$$D(n) = D_I(n) + \alpha_{n-1}^2 \rho_{n-1} D_P(n-1) + (a_n + b_n \sqrt{D_I(n)}) \sqrt{D_P(n-1)}, \quad (3.12)$$

where a_n and b_n are constants, and the third term corresponds to the correlation between $D_I(n)$ and $D_P(n-1)$. Fig. 3.2 gives an example to justify Eq. (3.12) by encoding the 50-th frame of QCIF Foreman with different values of $D_I(n)$ and $D_P(n-1)$. Let $D_e(n)$ denote the estimated value of $D(n)$ using Eq. (3.12) and $D_r(n)$ denote the actual value of $D(n)$, we can find that the relative difference defined by $\frac{D_r(n) - D_e(n)}{D_r(n)}$ is quite small, always within the range of $[-0.04, 0.03]$. This implies that $D_e(n)$ is a very accurate estimation of $D(n)$.

In [74], Zhang et al. found that the R-D characteristics of the baseline FGS within each bitplane is more accurate to be modelled as linear functions than exponential functions. In this research, we conduct some experiments to see whether the statement of the linear

CHAPTER 3. RATE-DISTORTION ANALYSIS OF LEAKY PREDICTION BASED FGS VIDEO FOR CONSTANT QUALITY CONSTRAINED RATE ADAPTATION

Figure 3.2: The estimation error of using Eq. (3.12) to calculate $D(n)$ of the 50-th frame of QCIF Foreman.

R-D relationship within each bitplane still holds for L-FGS. Fig. 3.3 shows the R-D curves of the six bitplanes in the 20-th L-FGS enhancement layer frame of the QCIF Foreman video sequence. It can be observed that the linear relationship roughly holds in all the bitplanes. Hence, $D_I(n)$ can be calculated using the linear interpolation technique, which can be performed at the offline stage.

It is not easy to compute $D_P(n-1)$ since it depends on the bits allocated to the $(n-1)$ -th enhancement layer frame and the bits used for the partial leaky prediction. There are two situations.

- (1) The amount of the allocated bits in the enhancement layer is smaller than that for the partial leaky prediction, which implies $\tilde{F}_E(n-1, i) = \tilde{F}_E^p(n-1, i)$. In this case, $D(n-1)$

CHAPTER 3. RATE-DISTORTION ANALYSIS OF LEAKY PREDICTION BASED FGS VIDEO FOR
CONSTANT QUALITY CONSTRAINED RATE ADAPTATION

becomes

$$\begin{aligned}
D(n-1) &= E\{[F_E(n-1, i) - \tilde{F}_E^p(n-1, i)]^2\} \\
&= E\{[F_E(n-1, i) - \hat{F}_E^p(n-1, i) + \hat{F}_E^p(n-1, i) - \tilde{F}_E^p(n-1, i)]^2\} \\
&= E\{[F_E(n-1, i) - \hat{F}_E^p(n-1, i)]^2\} + \\
&\quad E\{[\hat{F}_E^p(n-1, i) - \tilde{F}_E^p(n-1, i)]^2\} + \\
&\quad (c_{n-1} + d_{n-1}\sqrt{E\{[F_E(n-1, i) - \tilde{F}_E^p(n-1, i)]^2\}}) \\
&= E\{[F_E(n-1, i) - \hat{F}_E^p(n-1, i)]^2\} + D_P(n-1) + \\
&\quad (c_{n-1} + d_{n-1}\sqrt{D(n-1)}), \tag{3.13}
\end{aligned}$$

where c_{n-1} and d_{n-1} are two constants. Note that the third identity in Eq. (3.13) is obtained through extensive experiments. Therefore, $D_P(n-1)$ is derived as

$$\begin{aligned}
D_P(n-1) &= D(n-1) - E\{[F_E(n-1, i) - \hat{F}_E^p(n-1, i)]^2\} - \\
&\quad (c_{n-1} + d_{n-1}\sqrt{D(n-1)}), \tag{3.14}
\end{aligned}$$

where $E\{[F_E(n-1, i) - \hat{F}_E^p(n-1, i)]^2\}$ can be computed in the off-line stage. Fig. 3.4 gives an example to justify Eq. (3.14) by comparing the actual values of $D_P(n)$ and the estimated values using Eq. (3.14) for the 50-th frames of different video sequences. It can be seen that the difference between the estimated values and the actual values of $D_P(n)$ is negligible.

(2) The amount of the allocated bits in the enhancement layer is larger than or equal to that for the partial leaky prediction. Thus, $D_P(n-1)$ is given by

$$\begin{aligned}
D_P(n-1) &= E\{[\alpha_{n-2}\hat{F}_E^p(n-2, j) - \alpha_{n-2}\tilde{F}_E^p(n-2, j)]^2\} \\
&= \alpha_{n-2}^2 \rho_{n-2} D_P(n-2). \tag{3.15}
\end{aligned}$$

In this way, we set up the complete R-D model for L-FGS, which considers not only the distortion introduced in the current frame and the propagated distortion from the

Table 3.2: Experiment configuration for each video sequence.

Video Sequence	The base layer coding rate (kbps)	The amount of bits used in the partial prediction (kbits)	Leaky factor
Foreman	32	16	0.7
Akiyo	16	11	0.8
Mother & Daughter	16	10	0.8
News	32	12	0.8
Coastguard	32	16	0.7

reference frame but also the correlation between them. Note that, since the first frame (Frame 0) is intra-coded, $D(0) = D_I(0)$ and $D_P(0)$ can be calculated in the offline stage. At the online stage, based on the proposed R-D model, we can estimate $D(n)$ for any value of n given the current allocated bandwidth.

3.3.2 Experimental Results

In this section, we conduct experiments to test the accuracy of the proposed R-D model. The experiments are performed on five QCIF format video sequences including Foreman, Akiyo, Mother & Daughter, News, and Coastguard. All these video sequences are with 300 frames and coded at 10 fps. The configuration of each video sequence is shown in Table 3.2. We would like to point out that although in this chapter we apply the same amount of bits for partial prediction and the same leaky factor to each frame, our proposed R-D model can still be applied when different bits and different leaky factors are used in different frames, which is out of the scope of this research.

We consider both CBR and VBR channels. We assume we know and only know the available bandwidth at the current time, and we just allocate the available bandwidth to the current frame. Figures. 3.5-3.9 show the results of the distortion estimation in the cases of CBR channels, where the allocated amount of bits is always less than the original amount of bits for the partial prediction. Figures. 3.10-3.14 show the results for

CHAPTER 3. RATE-DISTORTION ANALYSIS OF LEAKY PREDICTION BASED FGS VIDEO FOR
CONSTANT QUALITY CONSTRAINED RATE ADAPTATION

the VBR channels, where the allocated amount of bits is sometimes less and sometimes larger than the original amount of bits for the partial prediction. It can be seen that the differences between the estimated distortion values and the actual distortion values are always very small, almost not noticeable. For example, the maximum differences in Fig. 3.5 and Fig. 3.6 are 1.55 and 0.9, respectively. Note that the estimated distortions are calculated completely based on the proposed R-D model without any update from the actual distortion values.

3.4 Constant Quality Constrained Rate Adaptation

In the previous section, we have constructed the overall R-D function, as shown in Eq. (3.12). In this section, we apply the proposed R-D model to solve the problem of constant quality constrained bit allocation for L-FGS.

3.4.1 Proposed Bit Allocation Algorithm

Since the base layer is pre-encoded at the fixed bit rate, i.e., the lower bound of a bandwidth range, we only consider the bit allocation in the enhancement layer. We assume the transportation channels are piece-wise static channels, covering both the considered CBR and VBR channels. The constant quality constrained bit allocation problem can be formulated as

$$\min \sum_{j=i+1}^{i+W_i-1} |D(j) - D(j-1)| \quad \text{and} \quad \max \sum_{j=i}^{i+W_i-1} B_j, \quad \text{subject to} \quad \sum_{j=i}^{i+W_i-1} B_j \leq \frac{R_i \cdot W_i}{F}, \quad (3.16)$$

where i is the index of the current frame, F is the frame rate, W_i is the current sliding window size (in number of frames), and R_i is the current channel bit rate (in kbps). Note that at the time of allocating bits for the i -th frame, we only know the available channel bandwidth R_i at that time and we assume R_i remains unchanged over the entire sliding

CHAPTER 3. RATE-DISTORTION ANALYSIS OF LEAKY PREDICTION BASED FGS VIDEO FOR
CONSTANT QUALITY CONSTRAINED RATE ADAPTATION

window W_i . This assumption is reasonable for piece-wise static channels, in which there exists strong correlation among adjacent time intervals.

Similar to [75], we develop a sliding window scheme with bisection search to solve the problem shown in Eq. (3.16). The basic idea is that, in each sliding window, we set a distortion D and make each frame have a distortion as close to D as possible by choosing appropriate rates. We repeat this procedure until a minimum distortion under the rate constraint is found. The detailed algorithm is described as follows.

Step 1: Initialize $i = 1$ and $\delta_0 = 0$, where δ_i is used to denote the difference between the total allocated number of bits and the total available channel bandwidth up to the frame i .

Step 2: Set $D_{low} = 0$, and assign the minimum distortion and the maximum distortion among all the base layer frames within the current sliding window W_i to D_{temp} and D_{high} , respectively.

Step 3: Calculate the target number of bits $B_j, j = i, \dots, i + W_i - 1$, for each frame in the sliding window to make D_j equal to or closest to D_{temp} based on the proposed R-D model.

Step 4: If $\sum_{j=i}^{i+W_i-1} B_j < \frac{R_i \cdot W_i}{F} - \delta_{i-1}$, set $D_{high} = D_{temp}$ and $D_{temp} = \frac{D_{low} + D_{high}}{2}$. Or else, If $\sum_{j=i}^{i+W_i-1} B_j > \frac{R_i \cdot W_i}{F} - \delta_{i-1}$, set $D_{low} = D_{temp}$ and $D_{temp} = \frac{D_{low} + D_{high}}{2}$.

Step 5: If $|\sum_{j=i}^{i+W_i-1} B_j - \frac{R_i \cdot W_i}{F} + \delta_{i-1}| > Mindiff \cdot W_i$, where $Mindiff$ is the predefined threshold which is used to control the rate adaptation accuracy, return to Step 3. Otherwise, the current value of B_i is used as the allocated number of bits for the frame i .

Step 6: Move to the next window. Update $\delta_i = \delta_{i-1} + B_i - \frac{R_i}{F}$ and $i = i + 1$. If there is any frame left, go to Step 2; otherwise stop.

Note that in this algorithm, for simplicity, we only consider the optimal bit allocation among all P frames and just assign the available bandwidth to the I-frame.

3.4.2 Experimental Results

We use the same setup as that in Section 3.3.2 to conduct experiments. In this simulation, we compare three bit allocation approaches: the uniform bit allocation method, our proposed bit allocation algorithm with $W = 40$ and our proposed bit allocation algorithm with $W = 80$, where W is the sliding window size. Fig. 3.15 shows the distortion results of adapting the QCIF Foreman sequence to CBR channels of 64 kbps and 128 kbps. Fig. 3.16 shows the distortion results of adapting the QCIF Akiyo sequence to CBR channels of 48 kbps and 80 kbps. It can be seen that our proposed constant quality constrained bit allocation scheme achieves much smoother video quality than the uniform bit allocation. In addition, we can see that the case with larger sliding window size normally has smaller distortion variation. This is because for larger sliding window size the frame qualities are averaged over a wider range. Note that, for the case of adapting Foreman sequence to 64 kbps, there is an inconsistent hole as shown in Fig. 3.15. That is caused by the CBR coding in the base layer and insufficient bits for the enhancement layer. Tables 3.3-3.7 give a summary of adapting different video sequences to different CBR channels. Specially, “Average Distortion” ($\bar{D}$) denotes the average MSE of a frame, which is calculated as

$$\bar{D} = 10\log_{10}\left(\frac{1}{L}\sum_{i=1}^L D(i)\right), \quad (3.17)$$

where L is the total number of frames. “Distortion STD” (σ_D) denotes the standard deviation of the frame distortions, which is calculated as

$$\sigma_D = 10\log_{10}\sqrt{\frac{1}{L}\sum_{i=1}^L (D(i) - \frac{1}{L}\sum_{j=1}^L D(j))^2}. \quad (3.18)$$

From the tables, we can see that the average distortion of our proposed bit allocation is slightly lower than or nearly equal to that of uniform bit allocation. The reduction of the distortion STD is outstanding with at least 3 dB reduction while the cost is that

CHAPTER 3. RATE-DISTORTION ANALYSIS OF LEAKY PREDICTION BASED FGS VIDEO FOR
CONSTANT QUALITY CONSTRAINED RATE ADAPTATION

our proposed scheme requires larger decoder buffer size and longer pre-loading time to prevent the buffer underflow and overflow. For the calculations of the required decoder buffer size and pre-loading time, please refer to [72].

Figures. 3.17 and 3.18 show the distortion performance of adapting the QCIF Foreman and Akiyo sequences to the time varying channels, respectively. Again, our proposed bit allocation scheme outperforms the uniform bit allocation method. It can be observed from the figures that our proposed bit allocation scheme can adapt to the time varying channels while still able to locally keep the consistent quality.

3.5 Summary

In this chapter, we have proposed an accurate R-D model for L-FGS, which has included not only the distortion introduced in the current frame and the propagated distortion from the reference frame but also the correlation between them. Based on the proposed R-D model, we have developed a sound practical scheme, i.e., the sliding window bit allocation scheme with bisection search, to solve the problem of constant quality constrained bit allocation. The experimental results have demonstrated that the proposed scheme significantly outperforms the traditional uniform bit allocation and is able to achieve very consistent video quality.

CHAPTER 3. RATE-DISTORTION ANALYSIS OF LEAKY PREDICTION BASED FGS VIDEO FOR
CONSTANT QUALITY CONSTRAINED RATE ADAPTATION

Figure 3.3: The R-D relationship within each bitplane of the 20-th enhancement layer frame of the QCIF Foreman video sequence at 10 fps and 32 kbps for the base layer. The R-D curves for all the bitplanes are placed in an increasing order (starting with the first bitplane) from left to right and from top to bottom. The x-axis represents number of bits and the y-axis represents MSE.

CHAPTER 3. RATE-DISTORTION ANALYSIS OF LEAKY PREDICTION BASED FGS VIDEO FOR
CONSTANT QUALITY CONSTRAINED RATE ADAPTATION

Figure 3.4: The comparison between the actual values of $D_P(n)$ and the estimated values using Eq. (3.14) for the 50-th frames of different video sequences. Left: QCIF Foreman. Right: QCIF Akiyo.

Figure 3.5: Distortion estimation results for adapting the L-FGS coded QCIF Foreman video sequence to CBR channels. Left: QCIF Foreman at 64 kbps. Right: QCIF Foreman at 128 kbps.

CHAPTER 3. RATE-DISTORTION ANALYSIS OF LEAKY PREDICTION BASED FGS VIDEO FOR CONSTANT QUALITY CONSTRAINED RATE ADAPTATION

Figure 3.6: Distortion estimation results for adapting the L-FGS coded QCIF Akiyo video sequence to CBR channels. Left: QCIF Akiyo at 48 kbps. Right: QCIF Akiyo at 80 kbps.

Figure 3.7: Distortion estimation results for adapting the L-FGS coded QCIF Mother & Daughter video sequence to CBR channels. Left: QCIF Mother & Daughter at 48 kbps. Right: QCIF Mother & Daughter at 80 kbps.

CHAPTER 3. RATE-DISTORTION ANALYSIS OF LEAKY PREDICTION BASED FGS VIDEO FOR
CONSTANT QUALITY CONSTRAINED RATE ADAPTATION

Figure 3.8: Distortion estimation results for adapting the L-FGS coded QCIF News video sequence to CBR channels. Left: QCIF News at 64 kbps. Right: QCIF News at 128 kbps.

Figure 3.9: Distortion estimation results for adapting the L-FGS coded QCIF Coastguard video sequence to CBR channels. Left: QCIF Coastguard at 64 kbps. Right: QCIF Coastguard at 128 kbps.

CHAPTER 3. RATE-DISTORTION ANALYSIS OF LEAKY PREDICTION BASED FGS VIDEO FOR CONSTANT QUALITY CONSTRAINED RATE ADAPTATION

Figure 3.10: Distortion estimation results for adapting the L-FGS coded QCIF Foreman video sequence to the VBR channel. Left: Available bandwidth. Right: Distortion estimation results.

Figure 3.11: Distortion estimation results for adapting the L-FGS coded QCIF Akiyo video sequence to the VBR channel. Left: Available bandwidth. Right: Distortion estimation results.

CHAPTER 3. RATE-DISTORTION ANALYSIS OF LEAKY PREDICTION BASED FGS VIDEO FOR
CONSTANT QUALITY CONSTRAINED RATE ADAPTATION

Figure 3.12: Distortion estimation results for adapting the L-FGS coded QCIF Mother & Daughter video sequence to the VBR channel. Left: Available bandwidth. Right: Distortion estimation results.

Figure 3.13: Distortion estimation results for adapting the L-FGS coded QCIF News video sequence to the VBR channel. Left: Available bandwidth. Right: Distortion estimation results.

CHAPTER 3. RATE-DISTORTION ANALYSIS OF LEAKY PREDICTION BASED FGS VIDEO FOR CONSTANT QUALITY CONSTRAINED RATE ADAPTATION

Figure 3.14: Distortion estimation results for adapting the L-FGS coded QCIF Coast-guard video sequence to the VBR channel. Left: Available bandwidth. Right: Distortion estimation results.

Figure 3.15: Distortion for adapting the L-FGS coded QCIF Foreman sequence to CBR channels. Left: QCIF Foreman at 64 kbps. Right: QCIF Foreman at 128 kbps.

CHAPTER 3. RATE-DISTORTION ANALYSIS OF LEAKY PREDICTION BASED FGS VIDEO FOR CONSTANT QUALITY CONSTRAINED RATE ADAPTATION

Figure 3.16: Distortion for adapting the L-FGS coded QCIF Akiyo sequence to CBR channels. Left: QCIF Akiyo at 48 kbps. Right: QCIF Akiyo at 80 kbps.

CHAPTER 3. RATE-DISTORTION ANALYSIS OF LEAKY PREDICTION BASED FGS VIDEO FOR
CONSTANT QUALITY CONSTRAINED RATE ADAPTATION

Table 3.3: The performance comparison between the uniform bit allocation and our proposed scheme for QCIF Foreman.

Channel Rate (kbps)	Bit Allocation Scheme	D (dB)	σ_D (dB)	Buffer Size (kbits)	Pre-loading Time (second)
32+32	Uniform	17.88	11.80	3.20	0.10
	Proposed ($W = 40$)	17.68	6.64	65.69	0.76
	Proposed ($W = 80$)	17.67	4.47	79.07	0.61
32+64	Uniform	16.40	10.16	6.40	0.10
	Proposed ($W = 40$)	16.31	5.03	64.70	0.28
	Proposed ($W = 80$)	16.31	-1.30	80.08	0.22
32+96	Uniform	14.98	9.08	9.60	0.10
	Proposed ($W = 40$)	14.82	4.73	63.38	0.11
	Proposed ($W = 80$)	14.86	-1.00	103.30	0.10

Table 3.4: The performance comparison between the uniform bit allocation and our proposed scheme for QCIF Akiyo.

Channel Rate (kbps)	Bit Allocation Scheme	D (dB)	σ_D (dB)	Buffer Size (kbits)	Pre-loading Time (second)
16+32	Uniform	11.72	4.84	3.20	0.10
	Proposed ($W = 40$)	11.64	1.60	50.79	1.59
	Proposed ($W = 80$)	11.59	-2.92	80.78	2.52
16+64	Uniform	9.88	3.56	6.40	0.10
	Proposed ($W = 40$)	9.54	0.46	44.68	0.70
	Proposed ($W = 80$)	9.48	-6.18	67.47	1.05
16+96	Uniform	7.76	-1.59	9.60	0.10
	Proposed ($W = 40$)	7.65	-6.24	32.04	0.33
	Proposed ($W = 80$)	7.66	-7.41	50.69	0.53

CHAPTER 3. RATE-DISTORTION ANALYSIS OF LEAKY PREDICTION BASED FGS VIDEO FOR
CONSTANT QUALITY CONSTRAINED RATE ADAPTATION

Table 3.5: The performance comparison between the uniform bit allocation and our proposed scheme for QCIF Mother & Daughter.

Channel Rate (kbps)	Bit Allocation Scheme	$\bar{D}$ (dB)	σ_D (dB)	Buffer Size (kbits)	Pre-loading Time (second)
16+32	Uniform	13.20	5.88	3.20	0.10
	Proposed ($W = 40$)	12.96	1.23	70.76	1.35
	Proposed ($W = 80$)	12.93	-2.61	82.23	1.81
16+64	Uniform	11.66	3.96	6.40	0.10
	Proposed ($W = 40$)	11.63	-0.45	74.63	0.69
	Proposed ($W = 80$)	11.61	-4.44	87.06	0.95
16+96	Uniform	9.80	3.51	9.60	0.10
	Proposed ($W = 40$)	9.95	-0.59	81.53	0.56
	Proposed ($W = 80$)	9.96	-5.33	91.57	0.70

Table 3.6: The performance comparison between the uniform bit allocation and our proposed scheme for QCIF News.

Channel Rate (kbps)	Bit Allocation Scheme	$\bar{D}$ (dB)	σ_D (dB)	Buffer Size (kbits)	Pre-loading Time (second)
32+32	Uniform	15.00	6.92	3.20	0.10
	Proposed ($W = 40$)	14.81	0.80	26.97	0.41
	Proposed ($W = 80$)	14.82	1.02	25.05	0.48
32+64	Uniform	13.84	4.26	6.40	0.10
	Proposed ($W = 40$)	13.77	-4.42	35.03	0.40
	Proposed ($W = 80$)	13.78	-4.38	36.75	0.43
32+96	Uniform	12.55	3.22	9.60	0.10
	Proposed ($W = 40$)	12.55	-1.69	39.31	0.23
	Proposed ($W = 80$)	12.56	-2.25	37.99	0.22

CHAPTER 3. RATE-DISTORTION ANALYSIS OF LEAKY PREDICTION BASED FGS VIDEO FOR
CONSTANT QUALITY CONSTRAINED RATE ADAPTATION

Table 3.7: The performance comparison between the uniform bit allocation and our proposed scheme for QCIF Coastguard.

Channel Rate (kbps)	Bit Allocation Scheme	$\bar{D}$ (dB)	σ_D (dB)	Buffer Size (kbits)	Pre-loading Time (second)
32+32	Uniform	19.66	12.75	3.20	0.10
	Proposed ($W = 40$)	19.53	8.17	77.78	1.22
	Proposed ($W = 80$)	19.52	6.78	89.11	1.72
32+64	Uniform	18.54	11.50	6.40	0.10
	Proposed ($W = 40$)	18.47	5.85	91.19	0.58
	Proposed ($W = 80$)	18.47	1.37	101.80	0.89
32+96	Uniform	17.30	10.81	9.60	0.10
	Proposed ($W = 40$)	17.22	5.78	91.20	0.44
	Proposed ($W = 80$)	17.22	-0.07	102.12	0.68

Figure 3.17: The simulation results for adapting the L-FGS coded Foreman sequence to the VBR channel. Left: Available bandwidth. Right: Distortion performance with window size=80.

CHAPTER 3. RATE-DISTORTION ANALYSIS OF LEAKY PREDICTION BASED FGS VIDEO FOR CONSTANT QUALITY CONSTRAINED RATE ADAPTATION

Figure 3.18: The simulation results for adapting the L-FGS coded Akiyo sequence to the VBR channel. Left: Available bandwidth. Right: Distortion performance with window size=80.

Chapter 4

Joint Quality and Rate Smoothed Encoding for Real-Time Video Streaming Applications

In the previous chapter, we have proposed the optimal bit allocation scheme for pre-encoded videos. In this chapter, we will study the bit allocation problem for real-time video encoding. This chapter is organized as follows. Section 4.1 introduces the existing works on quality-smoothed real-time video encoding and points out the need for rate-smoothed real-time video encoding. Section 4.2 gives an overview of the ρ -domain rate control algorithm, which is used in all the smoothed encoding algorithms proposed in this chapter. Section 4.3 discusses the state-of-the-art quality-smoothed encoding algorithms and shows their results. In Section 4.4, we analyze the problem of rate-smoothed encoding and prove that using the arithmetic averaging filtering to smooth video rate can automatically satisfy the average quality constraint. Section 4.5 further proposes a joint quality and rate smoothed encoding scheme. Finally, the summary of this chapter is given in Section 4.6.

4.1 Problem Statement

For video streaming applications over the constant bit rate channels, it is highly desired that video signals can be encoded in not only good average quality but also smooth video quality or less quality fluctuations among adjacent frames. In general, two-pass encoding schemes [71, 78] are often employed to achieve smooth video quality given a low bandwidth constraint. In particular, two-pass encoding schemes collect the necessary feature information in the first pass and then make use of these features in the second pass to achieve smooth video quality. However, for real-time video streaming applications, it is impossible to pre-encode a video sequence and thus global statistical information for the entire video sequence is not available.

Recently, we have seen some interesting research work [79, 80, 81, 82] on quality-smoothed single-pass encoding for real-time video streaming applications. In [79], a novel low-pass filtering scheme was proposed and the authors theoretically proved that using the geometric averaging filtering to smooth video distortion can automatically satisfy the average bandwidth constraint. In [80], Chen et al. employed the similar idea to that in [79], but used a different rate control method, i.e. the TMN8 rate control scheme. Chen et al. also considered the decoder buffer size and pre-loading time constraints in the encoding process. In [81], a novel sequence based bit allocation scheme is proposed based on the idea of allocating more bits to high complexity frames, and less bits to low complexity frames. Specially, mean absolute difference (MAD) is used to measure the coding complexity, and the bits allocated to the current frame is based on the square root of the ratio between the MAD of the current frame and the average MAD of all previous frames. In [82], a single-pass constant-distortion bit allocation scheme was proposed, which is based on the ρ -domain rate control algorithm [83] and takes the average distortion of all previous coded frames as the target distortion to encode the current frame. The experimental results show that the scheme proposed in [82] can

achieve better performance than the MAD-based allocation scheme in [81] at the cost of high computational complexity.

Although these quality-smoothed encoding schemes can reduce the quality fluctuation among adjacent video frames, they will increase rate fluctuation. Specifically, the VBR bitstreams of the quality-smoothed encoding schemes exhibit significant burstiness, where the high peak and the large variation in bit rates can substantially increase bandwidth requirement for the continuous playback at the client site. Therefore, in this chapter, we propose a rate-smoothed encoding scheme for real-time video streaming applications, which can reduce the traffic rate burstiness. In the case that the network resource is sufficiently large and the video quality has reached the target quality, the quality smoothing is not necessary and the rate smoothing can be performed to avoid overusing the unnecessary network resource but also achieve a smoothed traffic rate. Thus, we further propose a joint quality and rate smoothed encoding scheme, which can provide adaptive smoothing according to different situations.

4.2 Overview of ρ -Domain Rate Control Algorithm

In image and video compression algorithms based on transform coding, the high compression ratios are mainly achieved by efficiently coding zeros. Let ρ be the percentage of zeros among the quantized transform coefficients. It has been shown in [43] that there exists a linear relationship between the actual coding bit rate R and ρ . Specially, the rate model can be written as

$$R(\rho) = \theta(1 - \rho) + C_h, \quad (4.1)$$

where θ is a constant and C_h refers to the number of bits for header information and motion vectors.

CHAPTER 4. JOINT QUALITY AND RATE SMOOTHED ENCODING FOR REAL-TIME VIDEO
STREAMING APPLICATIONS

Let q be the quantization parameter used to quantize the DCT coefficients. In typical transform coding systems, ρ monotonically increases with q . Therefore, there is one-to-one mapping between ρ and q . Let $D_0(x)$ and $D_1(x)$ be the distributions of the DCT coefficients in the intra-coded and inter-coded macroblocks (MBs), respectively. For H.263+ quantization scheme, ρ can be calculated as [43]

$$\rho = \frac{1}{L} \sum_{|x| < 2q} D_0(x) + \frac{1}{L} \sum_{|x| < 2.5q} D_1(x), \quad (4.2)$$

where L is the number of coefficients in the current video frame.

Since the actual value of θ cannot be calculated before encoding a frame, an adaptive estimation algorithm was proposed in [84] to estimate the value θ during the encoding process. Let N_m be the number of already coded MBs in the current frame, R_m be the number of bits used to code these MBs, and Z_m be the corresponding number of zeros produced. The value of θ can be estimated as

$$\theta = \frac{R_m}{384 \cdot N_m - Z_m}. \quad (4.3)$$

This estimated θ will be used to encode the current MB.

Based on the linear rate model and the adaptive estimation of θ described above, the ρ -domain rate control algorithm can be described as follows [84].

Step 1. *Initialization.* Set $N_m = R_m = Z_m = 0$. Generate the distributions of DCT coefficients $D_0(x)$ and $D_1(x)$, respectively. Compute the target number of bits for coding the DCT coefficients as

$$R_C = R_T - R_{MV} - R_H, \quad (4.4)$$

where R_T is the target bit rate for the current video frame, R_{MV} is the number of bits for the motion vectors, and R_H is the number of bits for the header information. Set $\theta = 7$, which is the average value of θ obtained by extensive simulations.

CHAPTER 4. JOINT QUALITY AND RATE SMOOTHED ENCODING FOR REAL-TIME VIDEO
STREAMING APPLICATIONS

Step 2. *Determine the quantization parameter q .* Let Z be the number of zeros to be produced by the quantization of the non-encoded macroblocks. According to (4.1), Z can be calculated as

$$Z = 384 \cdot (N - N_m) - \frac{R_C - R_m}{\theta}. \quad (4.5)$$

Based on the one-to-one mapping table between Z and q , the corresponding quantization parameter q can be determined. Use $QP = q$ to quantize the current macroblock and further encode this macroblock with the video coding algorithm.

Step 3: *Update.* Set $Z_m = Z_m + Z_0$ and $R_m = R_m + R_0$, where Z_0 is the actual number of zeros and R_0 is the actual number of bits produced by encoding the current macroblock. Set $N_m = N_m + 1$. If $N_m \geq 10$, update the value of θ according to (4.3). Subtract the frequencies of the DCT coefficients in the current macroblock from $D_0(x)$ if the current macroblock is intra-coded, or from $D_1(x)$ if it is inter-coded.

Step 4. *Loop.* Repeat step 2 and 3 for the next macroblock until all the macroblocks in the current frame have been encoded.

4.3 Quality-Smoothed Encoding

4.3.1 Quality Smoothing Algorithm

The objective of quality-smoothed encoding is to minimize the variance among adjacent video frames. In this chapter, for the sake of completeness, we re-implement the lowpass filtering based quality smoothing algorithm in [79] with some modifications. We use the geometric averaging filter similar to that in [79], i.e.

$$D^*(n) = \prod_{i=1}^M [D(n-i)]^{1/M}, \quad (4.6)$$

where M is the filter length. Note that $D(j)$ in (4.6) is the actual distortion of the j -th frame instead of the CBR distortion being used in [79]. This is because it has been shown

CHAPTER 4. JOINT QUALITY AND RATE SMOOTHED ENCODING FOR REAL-TIME VIDEO
STREAMING APPLICATIONS

in [82] that using actual distortions can still achieve the smoothed video quality while the complexity of calculating CBR distortions can be avoided. In addition, we consider encoding video sequences with a pattern of one I-frame followed by all P-frames, and the first I-frame is coded with a fixed quantization parameter. The detailed algorithm is described as follows:

Step 1: *Initialization.* The first M P-frames are coded at the constant bit rate, i.e. at the bit rate of C/F , where C is the channel bit rate and F is the video frame rate. Let $b(n)$ denote the actual bit rate to encode the frame n . After encoding the first M frames, set $n = M + 1$ and $\delta(M) = \frac{C}{F} \cdot M - \sum_{j=1}^M b(j)$, which is to describe the accumulated mismatch between the given bit rate and the actual bit rate.

Step 2: *Determine the target distortion.* Calculate the target distortion $D^*(n)$ based on Eq. (4.6). According to the distributions of the DCT coefficients, $Z^*(n)$ can be determined, which is the corresponding number of zeros that will be produced for the distortion $D^*(n)$. The accurate estimation of $Z^*(n)$ can be found in [82].

Step 3: *Determine the target bit rate.* With the θ parameter of the previous frame, the target bit rate $R^*(n)$ can be calculated using Eq. (4.1). To meet the average bandwidth constraint, $R^*(n)$ is further updated as $R^*(n) = R^*(n) + \delta(n-1)/L_w$, where L_w is the length of a sliding window.

Step 4: *Encoding.* Using the ρ -domain rate control algorithm to encode the current frame. If the number of the coded MBs N_m in the current frame is larger than 10, update the value of target bit rate $R^*(n)$ with the new value of θ .

Step 6: *Loop.* Update $\delta(n) = \delta(n-1) - b(n) + \frac{C}{F}$ and $n = n + 1$. If there is any frame left, go to step 2; otherwise stop.

4.3.2 Results of Quality Smoothing

In this section, we conduct the simulations to show the performance of our developed quality-smoothed encoding scheme. The experiments are performed on three QCIF for-

CHAPTER 4. JOINT QUALITY AND RATE SMOOTHED ENCODING FOR REAL-TIME VIDEO
STREAMING APPLICATIONS

mat video sequences (300 frames and 30 fps): Foreman, News and Coastguard. The first I-frame is coded with a fixed QP of 13. The filter length M and the sliding window size L_w are set to 30. We compare three bit allocation schemes: CBR, the original quality-smoothed encoding scheme in [79] (termed as LP-Smoothing) and our developed quality-smoothed encoding scheme (termed as D-smoothing). The CBR scheme allocates constant bit rate to each video frame and uses the sliding window to compensate the coding mismatch, i.e. $R^*(n) = \frac{C}{F} + \delta(n-1)/L_w$.

The performance comparison of different bit allocation schemes is shown in Table 4.1, where $\overline{D}$ denotes the average MSE and $\overline{PSNR}$ denotes the average PSNR for the entire video sequence. The video quality variation is measured by $\overline{\Delta D}$, which is calculated as

$$\overline{\Delta D} = \frac{1}{n-1} \sum_{i=2}^n |D(i) - D(i-1)|. \quad (4.7)$$

Since the first I-frame and the first M P-frames are encoded in the same way under all the bit allocation schemes, for simplicity, they are not included in the calculations of $\overline{D}$, $\overline{PSNR}$, and $\overline{\Delta D}$ in Table 4.1, and the actual bit rate does not include the bit rate for the first I-frame.

From the table, we can see that compared with CBR, the LP-Smoothing scheme in [79] significantly reduces the quality fluctuation as indicated in the results of $\overline{\Delta D}$. However, the LP-Smoothing scheme has the problem of meeting the target average bit rate in some cases. For instance, it only spends an average bit rate of 309.96 kbps when coding the Coastguard sequence at a given bit rate of 400 kbps. Such a big mismatch at bit rate results in a 1.3 dB PSNR loss. On the contrary, our developed D-Smoothing scheme matches the target average bit rate quite well. This is mainly because we adopt the method in [82] for more accurate estimation of $Z^*(n)$. Moreover, our developed D-Smoothing scheme achieves even lower quality fluctuation in term of $\overline{\Delta D}$, about 80% quality fluctuation reduction when compared with CBR and more than 50% quality fluctuation reduction when compared with the LP-Smoothing scheme.

CHAPTER 4. JOINT QUALITY AND RATE SMOOTHED ENCODING FOR REAL-TIME VIDEO
STREAMING APPLICATIONS

Table 4.1: The performance comparison of different bit allocation schemes.

Video Sequence	Bit Allocation Scheme	Actual bit rate (kbps)	$\overline{PSNR}$ (dB)	$\overline{D}$	$\overline{\Delta D}$
Foreman (128 kbps)	CBR	128.04	32.11	41.79	2.36
	LP-Smoothing	119.46	31.88	42.88	1.07
	D-Smoothing	120.27	31.80	44.86	0.45
Foreman (256 kbps)	CBR	255.96	35.07	21.11	1.42
	LP-Smoothing	223.98	34.47	23.52	0.82
	D-Smoothing	247.59	34.89	21.76	0.19
News (64 kbps)	CBR	63.90	32.90	33.99	2.22
	LP-Smoothing	57.84	32.20	39.38	0.59
	D-Smoothing	64.05	32.63	35.70	0.27
News (96 kbps)	CBR	95.91	35.04	20.97	1.84
	LP-Smoothing	96.63	34.83	21.56	0.56
	D-Smoothing	95.97	34.56	22.92	0.19
Coastguard (256 kbps)	CBR	256.11	32.93	34.34	2.20
	LP-Smoothing	212.22	32.04	41.09	1.24
	D-Smoothing	261.18	32.89	36.21	0.45
Coastguard (400 kbps)	CBR	399.93	34.96	21.65	1.64
	LP-Smoothing	309.96	33.63	28.47	0.86
	D-Smoothing	405.18	34.98	22.33	0.24

Figures. 4.1-4.3 show the PSNR results of individual video frames for different video sequences coded at different bit rates. As expected, the PSNR curves of the LP-Smoothing scheme and the D-Smoothing scheme are much smoother than that of the CBR scheme. Comparing the LP-Smoothing scheme and the D-Smoothing scheme, we find that the PSNR curve of the D-Smoothing scheme is smoother at small scale while that of the LP-Smoothing scheme is smoother at large scale. We believe the main reason is that we use the actual distortions of previous frames in Eq. (4.6) instead of the CBR distortions proposed in [79], which tends to smooth out the quality fluctuation among adjacent frames but not distant frames. This also helps in matching the target average bit rate.

4.4 Rate-Smoothed Encoding

4.4.1 Rate Smoothing Algorithm

The objective of rate-smoothed encoding is to achieve smoothed traffic rate, while the quality of the entire video sequence can be kept at a certain level. This is very useful for single-pass VBR encoding. Existing single-pass VBR encoding schemes either use a fixed quantization parameter or fix the distortion to derive the quantization parameter, and rate smoothing has not been well considered. Motivated by the novel low-pass filtering scheme proposed in [79] for quality smoothing during real-time video encoding, in this section, we apply this low-pass filtering idea into the rate-smoothing problem, i.e. given a certain average quality requirement, how to smooth out the traffic rate during the real-time video encoding so that the complexity of the video delivery can be reduced. The detailed algorithm of our proposed scheme and the corresponding theoretical justification are described as follows.

Assume the target average PNSR of the entire video sequence is U_T . The main task is to determine the number of bits $R(n)$ allocated to encode the current n -th frame. To smooth out the traffic bit rate, we use the arithmetic averaging filter and $R(n)$ is calculated as

$$R(n) = \frac{1}{M} \sum_{i=1}^M R_T(n-i), \quad (4.8)$$

where M is the filter length and $R_T(i)$ is the corresponding bit rate when the i -th frame is coded with the PSNR value of U_T . Similar to [79], we can prove that the target average PSNR of the entire video sequence can be automatically achieved by using this arithmetic averaging filter. In particular, the average PNSR of the entire video sequence is calculated as

$$\bar{U}[N] = \frac{1}{N} \sum_{n=1}^N U(n), \quad (4.9)$$

CHAPTER 4. JOINT QUALITY AND RATE SMOOTHED ENCODING FOR REAL-TIME VIDEO
STREAMING APPLICATIONS

where N is the total number of encoded video frames and $U(n)$ is the PSNR value of the n -th frame.

Assume the transform coefficients have a Gaussian distribution. From the rate distortion theory [85], the rate distortion function for a Gaussian source is given by

$$R(D) = \frac{1}{2} \log_2 \frac{\sigma^2}{D}, \quad (4.10)$$

where σ^2 is the picture variance. Let $\sigma^2(n)$ be the variance of the n -th frame, where $1 \leq n \leq N$, and D_T be the corresponding mean square error (MSE) for the PSNR value of U_T . To encode the video with the PSNR value of U_T , the number of bits required can be derived as

$$R_T(n) = \frac{1}{2} \log_2 \frac{\sigma^2(n)}{D_T}. \quad (4.11)$$

According to (4.8) and (4.11), $R(n)$ can be expressed as

$$\begin{aligned} R(n) &= \frac{1}{M} \sum_{i=1}^M \left[\frac{1}{2} \log_2 \frac{\sigma^2(n-i)}{D_T} \right] \\ &= \frac{1}{2} \log_2 \frac{[\prod_{i=1}^M \sigma^2(n-i)]^{\frac{1}{M}}}{D_T}. \end{aligned} \quad (4.12)$$

Combining (4.12) and (4.10), the MSE value of the n -th frame becomes

$$\begin{aligned} D(n) &= \sigma^2(n) 2^{-2R(n)} \\ &= \frac{\sigma^2(n) D_T}{[\prod_{i=1}^M \sigma^2(n-i)]^{\frac{1}{M}}}. \end{aligned} \quad (4.13)$$

From (4.13), the corresponding PSNR value $U(n)$ can be calculated as

$$\begin{aligned} U(n) &= 10 \log_{10} \frac{255^2}{D(n)} \\ &= 10 \log_{10} \frac{255^2}{D_T} - 10 [\log_{10} \sigma^2(n) - \frac{1}{M} \sum_{i=1}^M \log_{10} \sigma^2(n-i)] \\ &= U_T - 10 [\log_{10} \sigma^2(n) - \frac{1}{M} \sum_{i=1}^M \log_{10} \sigma^2(n-i)]. \end{aligned} \quad (4.14)$$

CHAPTER 4. JOINT QUALITY AND RATE SMOOTHED ENCODING FOR REAL-TIME VIDEO
STREAMING APPLICATIONS

Thus, we have

$$\begin{aligned}\bar{U}[N] &= \frac{1}{N} \sum_{n=1}^N U(n) \\ &= U_T - \frac{10}{N} \sum_{n=1}^N [\log_{10} \sigma^2(n) - \frac{1}{M} \sum_{i=1}^M \log_{10} \sigma^2(n-i)].\end{aligned}\quad (4.15)$$

Since $\log_{10} \sigma^2(n)$ is bounded, we have

$$\lim_{N \rightarrow +\infty} \frac{10}{N} \sum_{n=1}^N [\log_{10} \sigma^2(n) - \frac{1}{M} \sum_{i=1}^M \log_{10} \sigma^2(n-i)] = 0, \quad (4.16)$$

and

$$\lim_{N \rightarrow +\infty} \bar{U}[N] = U_T. \quad (4.17)$$

From the above analysis, we can see that applying the low-pass filter on the coding rate can meet the target average PSNR value. Therefore, we can design the rate-smoothed encoding algorithm similar to the quality-smoothing algorithm described in Section 4.3. In particular, we encode the first M P-frames of the video sequence with the target PSNR U_T and use Eq. (4.8) to determine the target bit rate for the subsequent frames. Note that since R_T cannot be directly obtained in the single-pass encoding process except the first M P-frames, we first estimate the number of zeros Z_T corresponding to D_T using the method in [82] and then compute R_T according to Eq. (4.1) (θ is known after encoding the current frame).

In addition to the constraint of maintaining an average PSNR value, we can further limit the PSNR of each individual video frame within a certain range $[U_T^{min}, U_T^{max}]$, where $U_T^{min} < U_T < U_T^{max}$. The corresponding number of bits $R_T^{min}(n)$ and $R_T^{max}(n)$ can be obtained using the same method as that for estimating R_T except that we use the estimated value of θ obtained in the ρ -domain rate control process. This is because $R_T^{min}(n)$ and $R_T^{max}(n)$ are needed before encoding the current frame, and the accurate θ value is not known until we finish encoding the current frame. After obtaining $R_T^{min}(n)$ and $R_T^{max}(n)$, the target bit rate $R(n)$ is further updated as $R(n) = \min(R_T^{max}(n), \max(R_T^{min}(n), R(n)))$ in order to guarantee the PSNR of the current frame lies in the desired range.

4.4.2 Results of Rate Smoothing

In this section, we conduct experiments to test the performance of our proposed real-time rate-smoothed video encoding scheme. The same setup as that in Section 4.3.2 is utilized in the experiments. We compare the performance of three bit allocation schemes: constant distortion, the proposed rate-smoothing scheme (termed as R-Smoothing) and the constrained rate-smoothing scheme (termed as Constrained R-Smoothing). For the constant distortion scheme, each frame is encoded with the constant PSNR value U_T . The constrained rate-smoothing scheme denotes the proposed rate-smoothing scheme with the constraint of limiting the target PSNR value of each frame within the range of $[U_T^{min}, U_T^{max}]$. We choose $U_T^{min} = U_T - \delta$ and $U_T^{max} = U_T + \delta$, and set δ to be 1 dB, which describes the allowable PSNR fluctuation.

Table 4.2 and 4.3 show the performance comparison among different bit allocation schemes with the target average PSNR of 35 dB and 38 dB, respectively. Note that σ_R denoting the standard deviation of the bit rates is calculated as

$$\sigma_R = \sqrt{\frac{1}{n} \sum_{i=1}^n (R(i) - \frac{1}{n} \sum_{j=1}^n R(j))^2}, \quad (4.18)$$

and P_R is the maximal number of bits used by one frame. From the tables, we can see that the average PNSR results of all the three schemes match the target values very well. Compared with the constant distortion scheme, the proposed R-Smoothing scheme significantly reduces the rate variation σ_R , up to 34% reduction, and the peak rate P_R , up to 53% reduction, at the cost of increasing quality fluctuation. Although the constrained R-Smoothing scheme cannot reduce the peak rate as significantly as the R-Smoothing scheme, it reduces the quality fluctuation of the R-Smoothing scheme, up to 44% reduction. In terms of balancing the quality fluctuation and the rate fluctuation, we can regard the constrained R-Smoothing scheme as an approach between the constant distortion scheme and the R-Smoothing scheme. Obviously, if we loosen the constraint of the

CHAPTER 4. JOINT QUALITY AND RATE SMOOTHED ENCODING FOR REAL-TIME VIDEO
STREAMING APPLICATIONS

Table 4.2: The performance comparison of different bit allocation schemes for a target average PSNR of 35 dB.

Video Sequence	Bit Allocation Scheme	$\overline{PSNR}$ (dB)	$\overline{D}$	$\overline{\Delta D}$	$\overline{R}$ (kbps)	σ_R (kbits)	P_R (kbits)
Foreman	Constant Distortion	34.97	20.70	0.07	251.97	2.73	16.25
	R-Smoothing	34.94	21.64	1.50	255.42	2.31	14.96
	Constrained R-Smoothing	34.93	21.18	1.00	250.92	2.18	13.81
News	Constant Distortion	35.00	20.54	0.05	108.90	1.72	12.98
	R-Smoothing	35.10	20.85	1.78	97.05	1.16	6.08
	Constrained R-Smoothing	35.01	20.77	1.05	98.94	1.20	10.14
Coastguard	Constant Distortion	34.99	20.62	0.08	400.35	3.83	30.90
	R-Smoothing	35.01	20.82	1.52	410.34	3.15	20.48
	Constrained R-Smoothing	35.00	20.75	1.26	409.17	3.14	26.27

PSNR range, we can achieve smoother rate performance for the constrained R-Smoothing scheme. To illustrate the tradeoff between quality fluctuation and rate fluctuation in a clearer way, we depict the rate-distortion results of individual video frames for different video sequences under the three bit allocation schemes with a target average PSNR of 38 dB in Figures. 4.4-4.6.

4.5 Joint Distortion and Rate Smoothed Encoding

4.5.1 Joint Smoothing Algorithm

Considering the pros and cons of quality-smoothed encoding and rate-smoothed encoding, it is meaningful to combine them together and apply them for different scenarios in video streaming. In particular, when the bandwidth is insufficient, it is difficult to maintain a certain visual quality and thus quality smoothing becomes critical. On the other hand, when the bandwidth is sufficient, it is very easy to maintain a certain visual quality and thus it is more important to reduce the rate fluctuation. The bandwidth sufficiency is a relative concept. It depends on the complexity of video scenes. In this research, we use the target PSNR value U_T to determine the bandwidth sufficiency. That is when

CHAPTER 4. JOINT QUALITY AND RATE SMOOTHED ENCODING FOR REAL-TIME VIDEO
STREAMING APPLICATIONS

Table 4.3: The performance comparison of different bit allocation schemes for a target average PSNR of 38 dB.

Video Sequence	Bit Allocation Scheme	$\overline{PSNR}$ (dB)	$\overline{D}$	$\overline{\Delta D}$	$\overline{R}$ (kbps)	σ_R (kbits)	P_R (kbits)
Foreman	Constant Distortion	37.89	10.57	0.06	468.51	4.49	28.32
	R-Smoothing	37.84	11.13	0.90	464.37	3.52	25.51
	Constrained R-Smoothing	37.85	10.82	0.59	461.46	3.48	24.15
News	Constant Distortion	37.99	10.34	0.03	184.89	2.70	20.38
	R-Smoothing	38.05	10.76	1.13	158.10	1.77	9.52
	Constrained R-Smoothing	37.97	10.52	0.63	162.75	1.78	15.19
Coastguard	Constant Distortion	37.89	10.56	0.07	695.79	5.42	45.74
	R-Smoothing	37.85	10.88	0.97	699.24	4.43	32.58
	Constrained R-Smoothing	37.85	10.79	0.79	697.89	4.55	41.36

the video quality has reached U_T , we consider the network bandwidth is sufficiently large and we propose to smooth the traffic rate under the average constraint of U_T instead of smoothing the video quality under the traffic rate constraint. In this way, we will not only avoid overusing the unnecessary network resource but also achieve a smoothed traffic rate. The detailed algorithm is described as follows:

Step 1: *Initialization* (see Step 1 in the quality-smoothed encoding algorithm).

Step 2: *Determine the target distortion*. Calculate the target distortion $D^*(n)$ based on Eq. (4.6). If we overuse the bandwidth, $\delta(n-1) \leq 0$, or the target visual quality is below U_T , $D^*(n) > D_T$, go to step 3 (quality-smoothed encoding). Otherwise, set $h = 0$, where h is a counter for recording the number of rate-smoothed frames, and go to step 4 (rate-smoothed encoding).

Step 3: *Determine the target bit rate for quality-smoothed encoding* (see Step 2 and 3 in the quality-smoothed encoding algorithm).

Step 4: *Determine the target bit rate for rate-smoothed encoding*. If $h = 0$, set $R^*(n) = R_T(n)$ and go to step 5. On the other hand, if $h > 0$, we set $h = \min\{h, M\}$, where M is the maximal filter length. We calculate $R^*(n)$ based on Eq. (4.8) with

CHAPTER 4. JOINT QUALITY AND RATE SMOOTHED ENCODING FOR REAL-TIME VIDEO
STREAMING APPLICATIONS

the filter length of h . We determine the coding bit rates $R_T^{min}(n)$ and $R_T^{max}(n)$ which correspond to the picture quality of U_T^{min} and U_T^{max} , respectively. Then, $R^*(n)$ is further updated as $R^*(n) = \min(R_T^{max}(n), \max(R_T^{min}(n), R^*(n)))$.

Step 5: *Encoding* (see Step 4 in the quality-smoothed encoding algorithm). After encoding the current frame, calculate $R_T(n)$ if the rate-smoothed encoding is being performed.

Step 6: *Loop*. Update $\delta(n) = \delta(n-1) - b(n) + \frac{C}{F}$ and $n = n+1$. If there is any frame left and the current frame is coded with distortion smoothing or $\delta(n) \leq -\frac{C}{F}$, go to step 2; if the current frame is coded with rate smoothing and $\delta(n) > -\frac{C}{F}$, set $h = h+1$ and go to step 4; otherwise stop.

4.5.2 Results of Joint Smoothing

In this section, we conduct the experiments to show the performance of our proposed joint smoothing scheme. We use the same setup as that in Section 4.3.2 to conduct experiments. The filter length M is changed to 15 and the sliding window length L_w is 30. The target PSNR value U_T is set to 38 dB. QCIF Foreman, News and Coastguard are coded at 420 kbps, 160 kbps, and 560 kbps, respectively. We compare three bit allocation schemes: CBR, D-Smoothing and RD-Smoothing. The CBR scheme and the D-Smoothing scheme are the same as those in Section 4.3.2. The RD-Smoothing scheme is our proposed joint quality and rate smoothed encoding scheme. Note that the D-Smoothing scheme is almost the same as the RD-Smoothing scheme except that it only performs the distortion smoothing.

Figures. 4.7-4.9 show the rate and PSNR performance of each frame for different video sequences under different encoding schemes. The rate-smoothed encoding in the RD-Smoothing scheme is being used for encoding the 55-th to 220-th frames in QCIF Foreman, the 60-th to 105-th frames and the 245-th to 299-th frames in QCIF News,

and the 174-th to 272-nd frames in QCIF Coastguard. From the figures, we can see that the RD-Smoothing scheme achieves smoother traffic rate than the D-Smoothing scheme when the rate-smoothed encoding is being activated. On the other hand, the RD-Smoothing scheme achieves smoother video quality than the CBR scheme when the quality-smoothed encoding is being activated. Table 4.4 summarizes the rate-distortion performance of different encoding schemes. Note that in the calculation of the PSNR of each frame, we clip all the PSNR values larger than U_T to U_T since the quality beyond U_T is considered as unnecessary. From the table, it can be seen that the statistical data are consistent with our observations in Figures. 4.7 - 4.9. Particularly, compared with the D-Smoothing, the RD-Smoothing reduces σ_R , up to 21% reduction, and P_R , up to 20% reduction. For the News sequence, the RD-Smoothing scheme achieves better average PSNR performance than the D-Smoothing scheme and the gain is about 0.27 dB. This is because the unnecessary extra bandwidth for some frames is utilized by other frames. Although the reduction in quality fluctuation of the RD-Smoothing scheme is not as good as the D-Smoothing scheme, it still achieves up to 77% reduction, compared with the CBR scheme.

4.6 Summary

In this chapter, we have examined the existing quality-smoothed encoding scheme for real-time video streaming applications. We found that the concept of low pass filter can be applied for rate smoothing as well. We have proved that using arithmetic average filter can automatically achieve a target average PSNR. The proposed rate-smoothed encoding scheme is very useful in single-pass VBR video encoding, where the bandwidth is not a major constraint. We have further proposed a joint quality and rate smoothed encoding scheme, which can provide adaptive smoothing according to different situa-

CHAPTER 4. JOINT QUALITY AND RATE SMOOTHED ENCODING FOR REAL-TIME VIDEO
STREAMING APPLICATIONS

tions. Improved performance in terms of balancing the quality fluctuation and the rate fluctuation has been demonstrated in the experimental results.

Table 4.4: The performance comparison of different encoding schemes.

Video Sequence	Bit Allocation Scheme	$PSNR$ (dB)	$\bar{D}$	ΔD	$\bar{R}$ (kbps)	σ_R (kbits)	P_R (kbits)
Foreman	CBR	37.15	12.87	0.69	419.76	0.15	14.65
	D-Smoothing	37.14	12.83	0.09	412.29	4.00	27.03
	RD-Smoothing	37.13	12.89	0.25	409.62	3.49	25.45
News	CBR	37.60	11.50	0.70	161.52	0.07	5.44
	D-Smoothing	37.34	12.29	0.11	165.63	2.91	22.18
	RD-Smoothing	37.61	11.34	0.21	160.95	2.29	17.60
Coastguard	CBR	36.62	14.76	1.21	559.41	0.09	18.83
	D-Smoothing	36.55	15.23	0.15	553.35	3.87	30.88
	RD-Smoothing	36.54	15.23	0.28	555.27	3.58	30.88

Figure 4.1: The PSNR results of individual video frames for QCIF Foreman. Left: QCIF Foreman at 128 kbps. Right: QCIF Foreman at 256 kbps.

CHAPTER 4. JOINT QUALITY AND RATE SMOOTHED ENCODING FOR REAL-TIME VIDEO STREAMING APPLICATIONS

Figure 4.2: The PSNR results of individual video frames for QCIF News. Left: QCIF News at 64 kbps. Right: QCIF News at 96 kbps.

Figure 4.3: The PSNR results of individual video frames for QCIF Coastguard. Left: QCIF Coastguard at 256 kbps. Right: QCIF Coastguard at 400 kbps.

CHAPTER 4. JOINT QUALITY AND RATE SMOOTHED ENCODING FOR REAL-TIME VIDEO STREAMING APPLICATIONS

Figure 4.4: The rate-distortion performance of individual video frames for QCIF Foreman under different bit allocation schemes with the target average PSNR of 38 dB. Left: the rate results. Right: the PSNR results.

Figure 4.5: The rate-distortion performance of individual video frames for QCIF News under different bit allocation schemes with the target average PSNR of 38 dB. Left: the rate results. Right: the PSNR results.

CHAPTER 4. JOINT QUALITY AND RATE SMOOTHED ENCODING FOR REAL-TIME VIDEO STREAMING APPLICATIONS

Figure 4.6: The rate-distortion performance of individual video frames for QCIF Coast-guard under different bit allocation schemes with the target average PSNR of 38 dB. Left: the rate results. Right: the PSNR results.

Figure 4.7: The rate-distortion performance of individual video frames for QCIF Foreman coded at 420 kbps. Left: the rate results. Right: the PSNR results.

CHAPTER 4. JOINT QUALITY AND RATE SMOOTHED ENCODING FOR REAL-TIME VIDEO STREAMING APPLICATIONS

Figure 4.8: The rate-distortion performance of individual video frames for QCIF News coded at 160 kbps. Left: the rate results. Right: the PSNR results.

Figure 4.9: The rate-distortion performance of individual video frames for QCIF Coast-guard coded at 560 kbps. Left: the rate results. Right: the PSNR results.

Chapter 5

Joint Mode Selection and Unequal Error Protection for Bitplane Coded Video Transmission over Wireless Channels

In the previous chapters, we have studied two important problems in video streaming applications, i.e., optimal frame-level bit allocation for pre-encoded videos and for real-time videos. The optimal bit allocation schemes we proposed are for video streaming over error-free channels. In this chapter, we consider video streaming over error-prone wireless channels, and study FEC-based joint source channel coding (JSCC) for dependent bitplane based video coding and transmission. We use frame-level intra mode to stop error propagation and use UEP to protect bitplane coded video frames. In particular, we use MPEG-4 FGS as the bitplane based video codec and two modes are introduced in encoding the FGS enhancement layer of each video frame, i.e., with prediction or without prediction. Our focus is on how to allocate bits between source coding and channel coding with multiple priorities. Specially, we are interested in how to optimally select coding modes and find UEP solutions for bitplane based video coding. Based on the end-to-end R-D function we proposed in this chapter, we are able to find the optimal solutions for both mode selection and UEP so that the optimal tradeoff between efficiency

and robustness can be achieved.

Note that, in our proposed system, using MPEG-4 FGS is just for simplicity. We have no intention to provide a better FGS codec to compete with single-layer video coding. In fact, we only focus on the bitplane coded enhancement layer and the proposed scheme can be applied to single-layer bitplane based video coding.

This chapter is organized as follows. Section 5.1 describes the related work on bitplane coding and FEC-based joint source and channel coding. The system setup and problem formulation are described in Section 5.2. Section 5.3 analyzes the overall distortion and constructs the end-to-end R-D function. Based on the R-D function, how to find the optimal solution is discussed in Section 5.4. Section 5.5 presents simulation results. Finally, concluding remarks are given in Section 5.6.

5.1 Related Work

Bitplane coding has become more and more popular in image codecs such as SPIHT [86] and JPEG-2000 [87] and also in video codecs such as MPEG-4 FGS [26]. This is mainly because of the simplicity of bitplane coding, the good coding efficiency and the progressive property provided by bitplane coding, which allows a bitstream to be truncated in any position and thus provides rate scalability. Although bitplane coding can provide nice compression and scalability performance, the inter-frame prediction between adjacent frames can still make video bitstreams very vulnerable in transmission over wireless links since in wireless mobile networks there usually exist severe error corruption due to multipath and time-varying characteristics of wireless channels [88]. Therefore, error control techniques are usually used to combat channel errors.

Recently, we have seen extensive studies in FEC-based joint source channel coding (JSCC) for progressive image transmission over noisy channels or packet loss channels [89, 90, 67, 91, 92, 93]. The common idea of these schemes is to combine progressive

CHAPTER 5. JOINT MODE SELECTION AND UNEQUAL ERROR PROTECTION FOR BITPLANE CODED VIDEO TRANSMISSION OVER WIRELESS CHANNELS

source bitstreams with unequal error/loss protection (UEP/ULP), i.e., the more important information is given more protection. The progressive image bitstreams can be well fit into UEP/ULP because the earlier portions of a progressive image bitstream are always more important than the later parts. In [89], dynamic programming was employed to find the optimal UEP solution with fixed-length source data blocks. In [90], the authors also considered fixed-length source data blocks and developed an empirical model for optimal source channel rate allocation. In [92], Mohr et al. developed a ULP framework with fixed-length channel coding blocks, and used a greedy and iterative search algorithm to find the optimal channel coding rates, which costs comparatively long execution time. Kim et al. [93] further reduced the complexity of the ULP by employing dynamic programming to find the optimal channel coding rate for each bitplane instead of each channel coding block.

FEC-based joint source channel coding (JSCC) for video transmission is even more challenging. This is because video frames are not compressed independently. Distortion introduced in the current frame can propagate to the following frames. Therefore, in addition to using FEC for combating errors, regular intra-mode refreshment in macroblock level or frame level is needed. The research works in [53, 54] are the most representative techniques for joint mode and FEC selection. However, those works are not designed for bitplane based video codecs and only equal error protection (EEP) is considered. In fact, for bitplane coded video transmission over noisy channels, FEC-based joint source channel coding (JSCC) has been studied in [94, 95, 68, 96]. In [94, 95], UEP was applied for baseline MPEG-4 FGS, in which each enhancement layer frame is independent and it is essentially a progressive image. In [68, 96], empirical methods were used to find UEP solutions and propagated channel distortion was not well considered.

5.2 Problem Statement

5.2.1 Mode-Adaptive FGS System Architecture

In this research, we consider a mode adaptive FGS (MA-FGS) architecture, as shown in Fig. 5.1. In particular, each FGS enhancement layer frame can be encoded into one of the two modes, with prediction or without prediction. An FGS enhancement layer frame coded with prediction is termed as P-FGS while the other is termed as I-FGS. In the case of P-FGS, we directly apply the motion vectors, generated in the base layer encoding, to the enhancement layer prediction coding in order to avoid re-conducting the most time-consuming motion estimation process. Using the symbol definitions in Table 5.1, for I-FGS, the enhancement layer residue $G(n, u)$ can be expressed as

$$G(n, u) = F(n, u) = Y(n, u) - \hat{X}(n, u). \quad (5.1)$$

For intra coded macroblocks (MBs) in P-FGS, $G(n, u)$ is the same as that in I-FGS while, for inter coded MBs, $G(n, u)$ is given by

$$G(n, u) = Y(n, u) - \hat{X}(n, u) - \hat{F}(n - 1, v), \quad (5.2)$$

where $\hat{F}(n - 1, v)$ is the motion prediction of $F(n, u)$. In the case of using the half-pel motion estimation, the motion prediction pixel v in $\hat{F}(n - 1, v)$ could point to a half-pel

Figure 5.1: The MA-FGS architecture.

CHAPTER 5. JOINT MODE SELECTION AND UNEQUAL ERROR PROTECTION FOR BITPLANE CODED VIDEO TRANSMISSION OVER WIRELESS CHANNELS

Table 5.1: The symbol definitions.

$Y(n, u)$	: The original value of pixel u in the n -th video frame.
$\hat{Y}(n, u)$	: The reconstructed high quality value of pixel u in the n -th video frame without channel impairment.
$\tilde{Y}(n, u)$	: The received high quality value of pixel u in the n -th video frame at the receiver end.
$\hat{X}(n, u)$	: The base layer reconstruction value of pixel u in the n -th video frame in the feedback loop at the encoder.
$F(n, u)$	: The enhancement layer data after the prediction from the base layer, i.e., $Y(n, u) - \hat{X}(n, u)$.
$\hat{F}(n, u)$	: The reconstructed enhancement layer data without channel impairment, i.e., $\hat{Y}(n, u) - \hat{X}(n, u)$.
$\tilde{F}(n, u)$	: The received enhancement layer data at the receiver end, i.e., $\tilde{Y}(n, u) - \hat{X}(n, u)$.
$G(n, u)$	: The enhancement layer residue after the predictions from both the base layer and the enhancement layer.
$\hat{G}(n, u)$	: The reconstructed enhancement layer residue without channel impairment.
$\tilde{G}(n, u)$	: The received enhancement layer residue at the receiver end.

position.

Since we focus on the transmission of the bitplane coded enhancement layer in this research, we use H.263+ [44] instead of MPEG-4 to encode the base layer for simplicity. We consider encoding video sequences with a pattern of one I-frame followed by all P-frames, and TMN8 is used as the rate control scheme for the base layer. The FGS enhancement layer is encoded bitplane by bitplane, the same as that in the MPEG-4 FGS [26]. Fig. 5.2 shows the PSNR performance of encoding the QCIF Foreman sequence and the QCIF Akiyo sequence at different bit rates with two coding structures, i.e., the baseline FGS and the MC-FGS. The baseline FGS is the same as the MA-FGS when there is no prediction in the enhancement layer while the MC-FGS is the same as the MA-FGS when the prediction is always used in the enhancement layer. From Fig. 5.2,

CHAPTER 5. JOINT MODE SELECTION AND UNEQUAL ERROR PROTECTION FOR BITPLANE CODED VIDEO TRANSMISSION OVER WIRELESS CHANNELS

Figure 5.2: The PSNR performance of encoding the test video sequences at different bit rates. Left: QCIF Foreman with 10 fps and 32 kbps for the base layer. Right: QCIF Akiyo with 10 fps and 16 kbps for the base layer.

we can see that, with the prediction in the enhancement layer, the PSNR gap between the single layer video coding and the MA-FGS can be reduced from 2.5 dB to 0.5 dB for the Foreman sequence at 256 kbps and from 6.3 dB to 0.8 dB for the Akiyo sequence at 112 kbps.

5.2.2 Unequal Error Protection

As discussed in Section 5.1, UEP has been well studied in the past. In this research, we consider using UEP to transmit the FGS enhancement layer residue $G(n, u)$ while we assume that the base layer can be transmitted error-free. We employ the commonly used non-binary Reed-Solomon (RS) codes [55] with 8 bits/symbol as channel codes. An (l, k) RS code encodes each segment of k source symbols into a channel block of l symbols, and it can correct up to t symbol errors, where $t = \lfloor \frac{l-k}{2} \rfloor$.

We apply the UEP structure of fixed-length source blocks, as that in [89], to each FGS enhancement layer frame. Fig. 5.3 shows the adopted UEP structure. Let k denote the length of a source block, l_i denote the length of the i -th packet (A packet is the same

Figure 5.3: The adopted UEP structure.

as a channel block.), and M denote the number of packets. Given the total bandwidth for an FGS enhancement layer frame and the fixed value of k , the problem of UEP is to find the optimal l_i , $i = 1, 2, \dots, M$, so that the end-to-end distortion can be minimized.

5.2.3 Problem Formulation

Let α , $\alpha \in \{0, 1\}$, be the mode decision for an FGS enhancement layer frame, where $\alpha = 0$ means I-FGS while $\alpha = 1$ means P-FGS. Let $\vec{l} = (l_1, l_2, \dots, l_M)$ be a UEP solution. Let D be the overall picture distortion at the receiver end, defined as the mean square error (MSE) between the decoded high quality FGS video frame and the original one. The problem can be stated as: given a total bandwidth and the channel symbol error rate ϵ , how to choose α and $\vec{l}$ so that the overall distortion $D(\alpha, \vec{l})$ can be minimized.

It is well-known that the overall distortion consists of two parts: source distortion D_s and channel distortion D_c . In our proposed bitplane coded video transmission system, D_s is due to short of bandwidth for source coding while D_c is because of channel impairment introduced in current FGS frame or previous FGS frames. For a fixed $\vec{l}$, $\alpha = 0$ will prevent any channel distortion propagating from previous FGS frames at the cost of spending more bits to encode current FGS frame than that in the mode of P-FGS in terms of reaching the same picture quality. This is because introducing prediction in the

CHAPTER 5. JOINT MODE SELECTION AND UNEQUAL ERROR PROTECTION FOR BITPLANE CODED VIDEO TRANSMISSION OVER WIRELESS CHANNELS

enhancement layer can largely remove the temporal redundancy between two neighbor FGS frames. On the other hand, $\alpha = 1$ will improve the source coding efficiency while it is vulnerable if there is any channel distortion in the previous FGS frame. For a fixed α , larger values of the elements in $\vec{l}$ result in correcting more errors at the cost of less bits spent to source coding and thus introduce more source distortion. Smaller values of the elements in $\vec{l}$ will cause more bits allocated to the source coding while the source data are more likely to be corrupted due to the reduced capability of channel coding. Therefore, there exist optimal solutions for α and $\vec{l}$. In order to find the optimal solutions, we need to analyze the end-to-end distortion, which is discussed in Section 5.3.

Note that, in this research we consider a fixed overall total bandwidth for transmitting a video sequence, and after excluding the bandwidth for the base layer coding, the remaining bandwidth is equally allocated to each FGS enhancement layer frame. The searching for optimal α and $\vec{l}$ is performed on each FGS frame. Recently, we have seen extensive studies in how to allocate bandwidth among FGS frames [74], which can achieve much better performance than the uniform bit allocation. It would be interesting to apply those frame-layer bit allocation schemes in our proposed system. However, it is out of the scope of this research.

5.3 Analysis of End-to-End Distortion

Since we assume the base layer of FGS can be transmitted error free, the end-to-end distortion D_n between $Y(n, u)$ and $\tilde{Y}(n, u)$ becomes the same as the distortion between $F(n, u)$ and $\tilde{F}(n, u)$. $\tilde{F}(n, u)$ is essentially a random variable and thus D_n for the n -th frame can only be expected as

$$D_n = E\{[F(n, u) - \tilde{F}(n, u)]^2\}. \quad (5.3)$$

CHAPTER 5. JOINT MODE SELECTION AND UNEQUAL ERROR PROTECTION FOR BITPLANE CODED VIDEO TRANSMISSION OVER WIRELESS CHANNELS

Correspondingly, the source distortion D_s and the channel distortion D_c can be expressed as

$$D_s(n) = E\{[F(n, u) - \hat{F}(n, u)]^2\}, \quad (5.4)$$

$$D_c(n) = E\{[\hat{F}(n, u) - \tilde{F}(n, u)]^2\}, \quad (5.5)$$

respectively. It has been shown in [53] that D_s and D_c are generally uncorrelated. Hence, we can let

$$D_n = D_s(n) + D_c(n). \quad (5.6)$$

For I-FGS, the source distortion D_s^I is determined by the number of allocated source bits and thus depends on $\vec{l}$. Let d_i^I denote the distortion between the original FGS enhancement layer frame and the reconstructed one using the first i packets. D_s^I can be written as

$$D_s^I(n, \vec{l}) = d_M^I(\vec{l}). \quad (5.7)$$

The channel distortion D_c^I is caused by unrecovered corrupted packets. In this research, for simplicity, no error concealment algorithm is used to handle the video data loss. In other words, the decoding of one video frame is stopped at the first uncorrected packet loss no matter whether the following packets of current video frame are correct or not. Therefore, D_c^I is given by

$$\begin{aligned} D_c^I(n, \vec{l}) &= E\{[\hat{G}(n, u) - \tilde{G}(n, u)]^2\} \\ &= \sum_{i=0}^M P_i(\vec{l})[d_i^I(\vec{l}) - d_M^I(\vec{l})], \end{aligned} \quad (5.8)$$

where $P_i(\vec{l})$ is the probability that no uncorrected error remains in the first i packets but uncorrected errors occur in the $(i+1)$ -th packet. $P_i(\vec{l})$ can be calculated as

$$P_i(\vec{l}) = \begin{cases} p_e(l_1) & \text{if } i = 0 \\ p_e(l_{i+1}) \prod_{j=1}^i (1 - p_e(l_j)) & \text{if } i = 1, 2, \dots, M-1 \\ \prod_{j=1}^M (1 - p_e(l_j)) & \text{if } i = M \end{cases} \quad (5.9)$$

CHAPTER 5. JOINT MODE SELECTION AND UNEQUAL ERROR PROTECTION FOR BITPLANE CODED VIDEO TRANSMISSION OVER WIRELESS CHANNELS

where $p_e(l_i)$ is the probability that a packet contains uncorrected errors after channel decoding with parameter l_i . For a binary symmetric channel (BSC) with symbol error rate (SER) ϵ and a RS code (l_i, k) , $p_e(l_i)$ can be computed as

$$p_e(l_i) = 1 - \sum_{j=0}^{\lfloor \frac{l_i-k}{2} \rfloor} \binom{l_i}{j} \epsilon^j (1-\epsilon)^{l_i-j}. \quad (5.10)$$

Combining (5.7) and (5.8), we obtain the end-to-end distortion for I-FGS as

$$\begin{aligned} D^I(n, \vec{l}) &= D_s^I(n, \vec{l}) + D_c^I(n, \vec{l}) \\ &= \sum_{i=0}^M P_i(\vec{l}) d_i^I(\vec{l}) \end{aligned} \quad (5.11)$$

Similar to the source distortion of I-FGS, the source distortion of P-FGS can be written as

$$D_s^P(n, \vec{l}) = d_M^P(\vec{l}). \quad (5.12)$$

Notice that d_i^P is different from d_i^I since under different modes the R-D curves of the FGS residue are quite different. It is not easy to derive the channel distortion D_c^P of P-FGS since it can come from the channel impairment introduced in the current FGS frame and also the distortion propagated from the previous frame. According to the definition of D_c , we can derive D_c^P as

$$\begin{aligned} D_c^P(n, \vec{l}) &= E\{[\hat{F}(n, u) - \tilde{F}(n, u)]^2\} \\ &= E\{[\hat{F}(n, u) - (\tilde{F}(n-1, v) + \tilde{G}(n, u))]^2\} \\ &= E\{[(\hat{F}(n, u) - \hat{F}(n-1, v) - \tilde{G}(n, u)) \\ &\quad + (\hat{F}(n-1, v) - \tilde{F}(n-1, v))]^2\} \\ &= E\{[(\hat{G}(n, u) - \tilde{G}(n, u)) + (\hat{F}(n-1, v) - \tilde{F}(n-1, v))]^2\} \\ &= E\{[\hat{G}(n, u) - \tilde{G}(n, u)]^2\} + E\{[\hat{F}(n-1, v) - \tilde{F}(n-1, v)]^2\} \end{aligned} \quad (5.13)$$

We denote $E\{[\hat{G}(n, u) - \tilde{G}(n, u)]^2\}$ as D_{cc}^P , the distortion due to current loss, and we denote $E\{[\hat{F}(n-1, v) - \tilde{F}(n-1, v)]^2\}$ as D_{cp}^P , the distortion due to propagation. We

CHAPTER 5. JOINT MODE SELECTION AND UNEQUAL ERROR PROTECTION FOR BITPLANE CODED VIDEO TRANSMISSION OVER WIRELESS CHANNELS

would like to point out that the fifth identity in Eq. (5.13) is based on the assumption that the distortion due to current loss and the distortion due to propagation are uncorrelated. Also, notice that P-FGS may have some intra coded MBs. For intra coded MBs in P-FGS, the derivation above still holds since in those cases $\hat{F}(n-1, v) = \tilde{F}(n-1, v) = 0$ and $\hat{F}(n, u) = \hat{G}(n, u)$. The distortion due to current loss is the same as that in Eq. (5.8). For the distortion due to propagation, similar to [53], we assume that

$$D_{cp}^P = E\{[\hat{F}(n-1, v) - \tilde{F}(n-1, v)]^2\} = (1 - \beta)bD_c(n-1), \quad (5.14)$$

where β is the percentage of intra coded MBs in P-FGS and b is a constant describing the motion randomness of the video scene. Therefore, D_c^P can be further derived as

$$\begin{aligned} D_c^P(n, \vec{l}) &= D_{cc}^P + D_{cp}^P \\ &= \sum_{i=0}^M P_i(\vec{l})[d_i^P(\vec{l}) - d_M^P(\vec{l})] + (1 - \beta)bD_c(n-1). \end{aligned} \quad (5.15)$$

Combining (5.12) and (5.15), we obtain the end-to-end distortion for P-FGS as

$$\begin{aligned} D^P(n, \vec{l}) &= D_s^P(n, \vec{l}) + D_c^P(n, \vec{l}) \\ &= \sum_{i=0}^M P_i(\vec{l})d_i^P(\vec{l}) + (1 - \beta)bD_c(n-1). \end{aligned} \quad (5.16)$$

Finally, the optimal problem of joint mode selection and UEP can be formulated as

$$\begin{aligned} \min_{\alpha, \vec{l}} D_n(\alpha, \vec{l}) &= (1 - \alpha)D^I(n, \vec{l}) + \alpha D^P(n, \vec{l}), \\ \text{subject to} \quad &\sum_{i=0}^M l_i \leq R_n \end{aligned} \quad (5.17)$$

where $\alpha \in \{0, 1\}$ and R_n is the allocated bandwidth for the n -th FGS enhancement layer frame.

5.4 Joint Source Channel Optimization

In the previous section, we have constructed the overall end-to-end R-D function, as shown in Eq. (5.17). In this section, we discuss how to solve such a joint source channel optimization problem. According to Eq. (5.17), it is clear that, in order to minimize the overall distortion, we need to calculate the optimal $D^I(n, \vec{l})$ and $D^P(n, \vec{l})$, and then choose the one having smaller value as the coding mode. However, to obtain the optimal $D^I(n, \vec{l})$ and $D^P(n, \vec{l})$, we need to have the source R-D curves, d_i^I and d_i^P , and also we need to find the optimal UEP solution, $\vec{l}^*$. In addition, the accuracy of the approximated model for computing D_c^P , as given in Eq. (5.15), needs to be justified. In the following, we will discuss these three key issues in detail.

5.4.1 R-D Extraction of FGS Source Coding

The performance of model-based methods for rate allocation depends heavily on the accuracy of the adopted model. Recently, Zhang et al. found that the R-D characteristics within each bitplane of the baseline FGS is more accurate to be modeled as linear functions than exponential functions, and the R-D curve of the FGS enhancement layer can be estimated by extracting R-D points at the end of each bitplane during the encoding process followed by linear interpolation [74].

In this research, we conduct some experiments to see whether the statement of the linear R-D relationship within each bitplane still holds for FGS enhancement layer frames with temporal prediction, i.e. P-FGS. Fig. 5.4 shows the R-D curves of the seven bitplanes in the third FGS enhancement layer frame (coded in the P-FGS mode) of the QCIF Foreman sequence. We can see that the linear relationship exists among almost all bitplanes except the first one. This is because the first bitplane of P-FGS typically has very small amount of data. The flat steps in the R-D curve of the first bitplane are possibly caused by the fragmentation in coding a long sequence of all zero bits. We have

CHAPTER 5. JOINT MODE SELECTION AND UNEQUAL ERROR PROTECTION FOR BITPLANE CODED VIDEO TRANSMISSION OVER WIRELESS CHANNELS

Figure 5.4: The R-D relationship within each bitplane of the third frame of the QCIF Foreman video sequence at 10 fps and 32 kbps for the base layer. The previous two enhancement frames are coded at 6400 bits. The R-D curves for all the bitplanes are placed in an increasing order (starting with the first bitplane) from left to right and from top to bottom. The x-axis represents number of bits while the y-axis represents MSE.

tested other frames and other video sequences at different coding rates, and similar R-D relationship can be observed.

Although the linear relationship does not hold in the first bitplane of P-FGS, it can be neglected since the number of bits in the first bitplane is so small (usually less than 50 bytes) that we can deem the data of the first bitplane will be encapsulated into one

packet. Therefore, d_i can be calculated as

$$d_i = \delta_{j-1} - \frac{\delta_{j-1} - \delta_j}{B_j - B_{j-1}}(ik - B_{j-1}), \quad (5.18)$$

where δ_j and B_j are the sampling distortion and rate values at the end of the j -th bitplane, and $B_{j-1} \leq ik \leq B_j$.

5.4.2 Channel Distortion Estimation

To verify the model of estimating channel distortion of P-FGS, first, we consider the case without channel feedback. We can directly use Eq. (5.15) to calculate the estimated channel distortion starting from the first I-FGS frame. Notice that, in Eq. (5.15), β and b are different for different frames. For one frame, β is known after the encoding of the FGS base layer. To determine b for the n -th FGS frame, we choose a possible pair of $\hat{F}(n-1, u)$ and $\tilde{F}(n-1, u)$ and calculate b as

$$b = \frac{E\{[\hat{F}(n-1, v) - \tilde{F}(n-1, v)]^2\}}{(1 - \beta)E\{[\hat{F}(n-1, u) - \tilde{F}(n-1, u)]^2\}}, \quad (5.19)$$

where v in the $(n-1)$ -th frame is the motion prediction pixel for the pixel u in the n -th frame.

We also consider the cases with channel feedback as that in [53]. With the channel feedback, the encoder can determine the quality of the received $(n - \Delta)$ -th FGS frame and its previous frames, where Δ is the feedback delay in the unit of frame interval. In this way, the channel distortion estimation becomes how to estimate $D_c(n)$ given that we are able to obtain the actual values of $\{D_c(n - \Delta - i), \quad i \geq 0\}$. Based on Eq. (5.15), we can re-formulate the channel distortion estimation with feedback as

$$\begin{aligned} D_c^P(n) = D_{cc}^P(n) + \sum_{i=1}^{\Delta-1} D_{cc}^P(n-i) & \left[\prod_{j=0}^{i-1} (1 - \beta(n-j)) b(n-j) \right] \\ & + D_c(n - \Delta) \left[\prod_{i=0}^{\Delta-1} (1 - \beta(n-i)) b(n-i) \right]. \end{aligned} \quad (5.20)$$

To test the performance of the proposed channel distortion estimation scheme, we use the MC-FGS system without UEP to conduct the experiments. We choose the packet loss rate to be 5% for each packet (with the fixed packet size of 40 bytes) and the loss only occurs in the latter 30% portion of each frame. This is reasonable because UEP will be employed eventually which will result in that packet loss is more likely to happen in the latter portion. We run the simulation 20 times for each test scenario and then compute the average channel distortion. Fig. 5.5 shows the experimental results of channel distortion estimation with different feedback delays. We can see that the estimated channel distortion matches the actual channel distortion very well, especially in the case of $\Delta = 1$. Although for larger values of Δ the estimation performance is not as good as the case of $\Delta = 1$, it can still consistently match the actual channel distortion and the performance will not become worse with the further increasing of Δ .

5.4.3 Dynamic Programming for UEP

One way to find the optimal UEP solutions for our proposed system is through exhaustive search similar to the greedy and iterative search algorithm proposed in [92]. However, the exhaustive search algorithm will cost extremely long execution time and thus not feasible for video applications, which have very tight delay constraints. Since we use the UEP architecture with fixed-length source blocks, we are able to apply the dynamic programming approach proposed in [89] to find the optimal solution. The basic idea of the dynamic programming approach is to use a sum of incremental rewards, which are accumulated with each successfully decoded packet, to express the expected quality [93]. Similar to [89], we can construct the accumulated reward $g(i, \vec{l})$ and rate $r(i, \vec{l})$ into two recursive functions as

$$g(i, \vec{l}) = (1 - p_e(l_i))[(d_{i-1} - d_i) + g(i + 1, \vec{l})], \quad (5.21)$$

$$r(i, \vec{l}) = l_i + r(i + 1, \vec{l}). \quad (5.22)$$

CHAPTER 5. JOINT MODE SELECTION AND UNEQUAL ERROR PROTECTION FOR BITPLANE CODED VIDEO TRANSMISSION OVER WIRELESS CHANNELS

Figure 5.5: The results of the channel distortion estimation for the QCIF Foreman video sequence coded at 10 fps and 96 kbps with 32 kbps for the base layer, no FEC and 5% packet loss rate which only occurs in the latter 30% part of each frame. Top: $\Delta = 1$. Bottom Left: $\Delta = 5$. Bottom Right: $\Delta = 10$.

Although the dynamic programming approach can find the optimal UEP solution much faster than using exhaustive search, the execution time will still become intolerable with the increase of total number of packets in an FGS frame. Therefore, in the case of relatively higher bit rates or more packets available, we only consider providing layered UEP while in the case of lower bit rates we still provide different protection among different packets. In this way, we try to use the dynamic programming to quickly find the optimal UEP solution.

The idea of layered UEP is to partition the packets into several layers and provide different protection among different layers instead of different packets. Given the total bits r for each enhancement frame, the total number of packets M is unknown until the end of dynamic programming because the length of channel code for each packet is uncertain. By assuming the channel coding rate is always not less than $1/2$, we can calculate the minimum number of packets N as $N = \lfloor r/(2 \cdot k) \rfloor$. If we want to partition all the packets into S layers, a simple approach is to partition the first N packets into $S - 1$ layers with the equal number of packets, i.e., $\lfloor N/(S - 1) \rfloor$, and all the remainder packets belong to the S -th layer. If we want to reduce the time to obtain the optimal UEP solution, we can reduce the value of S . It is a tradeoff between the executing time and the UEP performance.

5.5 Experimental Results

In this section, we perform experiments to illustrate the effectiveness of the proposed system, i.e., the MA-FGS with UEP. The experiments are performed on two QCIF format video sequences. The first video sequence is the Foreman sequence with 300 frames, which contains large facial movements and camera panning at the end. The second one is the Akiyo sequence with 300 frames, which contains low activity (slow motion). Both video sequences are coded at 10 fps. The base layers are coded at 32 kbps for the Foreman sequence and 16 kbps for the Akiyo sequence, respectively. We compare our proposed system with other 3 schemes, i.e., MC-FGS with EEP, baseline FGS with UEP and baseline FGS with EEP. For EEP, channel coding rate is selected based on pre-determined threshold, i.e., the corresponding probability of corrupted packets should be less than the threshold. In this simulation, we choose 10^{-6} as the threshold for MC-FGS with EEP and 10^{-3} as the threshold for baseline FGS with EEP since MC-FGS has severe error propagation problem while baseline FGS does not have. We apply our proposed

CHAPTER 5. JOINT MODE SELECTION AND UNEQUAL ERROR PROTECTION FOR BITPLANE CODED VIDEO TRANSMISSION OVER WIRELESS CHANNELS

model with no mode selection for baseline FGS with UEP. The length of source data in a packet is fixed to 40 bytes and the FEC range for UEP is from 0 byte to 40 bytes. $\Delta = 5$ is chosen as the feedback distance for applying our proposed model. Note that, in our proposed MA-FGS, if there are several I-FGS frames between the last feedback frame and the current frame, the estimated channel distortion of the closest I-FGS frame will be utilized as the feedback. It is the same when we code the first 5 frames since there is no feedback available. The experimental results are obtained over 20 simulations for each configuration.

Fig. 5.6 shows the PSNR performance under different total bandwidths with a fixed SER of 0.1 and Fig. 5.7 shows the PSNR performance under different SERs with a fixed total bandwidth. We can see that the MA-FGS with UEP achieves much better performance than the baseline FGS with UEP. For example, as shown in Fig. 5.6, in the case of coding the Foreman sequence, the MA-FGS with UEP outperforms the baseline FGS with UEP about 0.9 dB at 256 kbps. In the case of coding the Akiyo sequence, the MA-FGS with UEP obtains about 2.2 dB gain at 112 kbps. Although in most cases MC-FGS with EEP achieves better performance than baseline FGS with UEP or EEP, it is always inferior to MA-FGS with UEP. Moreover, for Foreman sequence with large values of SER, the performance of MC-FGS with EEP has significant performance degradation, even worse than baseline FGS with EEP. This is because large amount of bits are used for channel coding in the cases of large SERs. Fig. 5.8 shows the average PSNR of different video frames with $\text{SER} = 0.1$ and a fixed total bandwidth. It can be observed that the MA-FGS with UEP achieves better performance for almost all the video frames.

Fig. 5.9 shows an example of the channel distortion and the corresponding coding mode for each frame over one simulation in the case of transmitting the Foreman QCIF sequence using MA-FGS with UEP at 128 kbps and $\text{SER} = 0.1$. From the figure, we can see that frames (1, 40, 53, 63, 64, 65, 66, 68, 69) are coded with I-FGS. Encoding the 40-th

frame with I-FGS is because that the channel distortion for the 35-th frame is unexpected large and this information can only be available to the encoder at the time of coding the 40-th frame. With the correct mode decision in the 40-th frame, the channel distortion is reduced immediately. For frames (53, 63, 64, 65, 66, 68, 69), they are coded with I-FGS mainly because these frames are with large motion. From this example, we can see that our proposed mode selection scheme can accurately adapt to both channel distortion and motion characteristics of video frames. We believe, for more realistic wireless channels with unexpected severe fading, using I-FGS due to channel distortion will occur more often.

5.6 Summary

In this chapter, we have derived an overall end-to-end R-D function for the proposed system of jointly considering the mode selection and UEP. In the derived R-D function, we have included not only the source distortion and the channel distortion introduced in the current frame but also the propagated channel distortion from the previous frames. Based on the overall R-D analysis, we have developed a sound practical scheme to find the optimal solutions of mode selection and UEP for delay-constrained video applications. Extensive experimental results have demonstrated that the proposed system significantly improves the end-to-end video quality for wireless video coding and transmission.

CHAPTER 5. JOINT MODE SELECTION AND UNEQUAL ERROR PROTECTION FOR BITPLANE CODED VIDEO TRANSMISSION OVER WIRELESS CHANNELS

Figure 5.6: The PSNR performance of transmitting the video sequences with different total bandwidth over a BSC channel with $\epsilon = 0.1$. Left: QCIF Foreman. Right: QCIF Akiyo.

Figure 5.7: The PSNR performance of transmitting the test video sequences with a fixed total bandwidth over BSC channels with different SERs. Left: QCIF Foreman with 128 kbps. Right: QCIF Akiyo with 64 kbps.

CHAPTER 5. JOINT MODE SELECTION AND UNEQUAL ERROR PROTECTION FOR BITPLANE CODED VIDEO TRANSMISSION OVER WIRELESS CHANNELS

Figure 5.8: The average PSNR performance of all the video frames for transmitting the test video sequences with a fixed total bandwidth over a BSC channel with $\epsilon = 0.1$. Left: QCIF Foreman with 128 kbps. Right: QCIF Akiyo with 64 kbps.

Figure 5.9: An example of the channel distortion and the corresponding coding mode for each frame over one simulation in the case of transmitting the QCIF Foreman at 128 kbps over a BSC channel with $\epsilon = 0.1$. Left: Channel distortion. Right: Coding Mode.

Chapter 6

Conclusions and Future Directions

6.1 Conclusions

In this dissertation, we have studied the bit allocation issues in video coding and transmission for various video streaming scenarios. The objective of our research is to provide best video quality at the receiver, as well as smooth traffic rate to improve the transmission efficiency and benefit the network resource allocation. Our work throughout this dissertation is based on the accurate rate distortion analysis and modelling of the entire video transmission system, and the experimental results have shown improved performance.

We have studied both pre-encoded video and real-time video streaming applications. For pre-encoded video, the off-line rate distortion analysis can be performed since the whole video sequence is available. In order to perform constant quality constrained bit allocation for L-FGS coded video, we have proposed an accurate R-D model for L-FGS, which has included not only the distortion introduced in the current frame and the propagated distortion from the reference frame but also the correlation between them. Based on the proposed R-D model, we have developed a sound practical scheme, i.e., the sliding window bit allocation scheme with bisection search, to solve the problem of constant quality constrained bit allocation.

CHAPTER 6. CONCLUSIONS AND FUTURE DIRECTIONS

We have also proposed a rate-smoothed encoding scheme for real-time video streaming applications, which can reduce the traffic rate burstiness. In the case that the network resource is sufficiently large and the video quality has reached the target quality, the quality smoothing is not necessary and the rate smoothing can be performed to avoid overusing the unnecessary network resource but also achieve a smoothed traffic rate. Therefore, we have further proposed a joint quality and rate smoothed encoding scheme, which can provide adaptive smoothing according to different situations.

Wireless channels are with high error rate and channel coding is often employed to make video data more resilient in the error-prone environment. In this dissertation, we have studied joint source channel coding for bitplane based video coding over wireless channels. We consider using frame-level intra mode to stop error propagation and using UEP to combat channel errors. An overall end-to-end R-D function has been proposed in order to solve the optimization problem. In the derived R-D function, we have included not only the source distortion and the channel distortion introduced in the current frame but also the propagated channel distortion from the previous frames. Based on the overall R-D analysis, we have developed a sound practical scheme to find the optimal solutions of mode selection and UEP for delay-constrained video applications.

6.2 Future Directions

Some ideas for future extensions of our research are listed as follows:

- Constant quality constrained bit allocation for L-FGS with adaptive leaky prediction factor and buffer constraints

In our current bit allocation scheme, we do not consider the constraints of the buffer size and pre-loading time. In order to keep variation of video quality as small as possible, an optimal buffer control algorithm is necessary instead of only

CHAPTER 6. CONCLUSIONS AND FUTURE DIRECTIONS

preventing the buffer from underflow or overflow. In addition, in the R-D model that we proposed for L-FGS, the leaky prediction factor is fixed for each frame. It is possible to adaptively select the leaky prediction factor for each frame so that the coding efficiency can be further improved. Therefore, it would be interesting to extend our proposed R-D model for the adaptive selection of the leaky prediction factor.

- Frame-level bit allocation for single-layer video coding

Although many frame-level bit allocation schemes have been proposed, the bit allocation performance for single-layer video coding is still far from optimal. This is mainly because of the interframe dependency. Some works have been conducted to model the relationship between the distortion of current frame and the combination of quantization choices of the previous frame and the current frame [7, 8]. Since the number of combinations for the successive quantization choices grows exponentially for a group of video frames, it is not possible to get the exact solution for the practical video coding system due to the huge computational complexity. The R-D model that we proposed for L-FGS considers not only the distortion introduced in the current frame and the propagated distortion from the reference frame but also the correlation between them. Although there are some difference between L-FGS and the current single layer video coding standards, it will be very interesting to extend our current model to the single layer video encoding.

- Error resilient quality-smoothed video encoding scheme for real-time video transmission

Most of the current quality-smoothed real-time video encoding schemes assume error-free channel conditions. Recently, a novel source-channel constant distortion

CHAPTER 6. CONCLUSIONS AND FUTURE DIRECTIONS

model was proposed in [97], which is based on the recursive optimal per-pixel estimate (ROPE) algorithm [61]. Due to the high computational complexity of ROPE and the heuristic method employed to find the optimal solutions, the proposed scheme has extremely high complexity and is not suitable for practical systems. It is necessary to further study quality-smoothed video encoding in order to reduce the complexity and improve the error-resilience.

References

- [1] ISO/IEC JTC 1/SC 29/WG 11, “Information technology - generic coding of audio-visual objects - part 2: visual,” *WG11 N2688*, March 1999.
- [2] Joint Video Team of ITU-T VCEG and ISO/IEC MPEG, “Draft ITU-T recommendation and final draft international standard of joint video specification (ITU-T Rec.H.264/ISO/IEC 14496-10 AVC),” *JVT-G050*, Mar. 2003.
- [3] A. Vetro, C. Christopoulos, and H. Sun, “Video transcoding architectures and techniques: an overview,” *IEEE Signal Processing Magazine*, pp. 18–29, March 2003.
- [4] M. Karczewicz and R. Kurceren, “The SP- and SI- frames design for H.264/AVC,” *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 13, pp. 637–644, Jul. 2003.
- [5] J.-R. Ohm, “Advances in scalable video coding,” *Proceedings of the IEEE*, pp. 42–56, Jan. 2005.
- [6] B. Girod, A. Aaron, S. Rane, and D. Rebollo-Monedero, “Distributed video coding,” *Proceedings of the IEEE*, pp. 71 – 83, Jan. 2005.
- [7] K. Ramchandran, A. Ortega, and M. Vetterli, “Bit allocation for dependent quantization with applications to multiresolution and MPEG video coders,” *IEEE Trans. on Image Processing*, pp. 533–545, Sept. 1994.
- [8] L.-J. Lin and A. Ortega, “Bit-rate control using piecewise approximated rate-distortion characteristics,” *IEEE Trans. on Circuits and Systems for Video Technology*, pp. 446–459, Aug. 1998.

REFERENCES

- [9] ITU-T Recommendation H.261, “Video codec for audiovisual services at $p \times 64$ kb/s,” 1990.
- [10] ITU-T Recommendation H.263, “Video coding for low bitrate communications,” Mar. 1990.
- [11] 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*, pp. 849 – 866, Nov. 1998.
- [12] S. Wenger, G. Knorr, J. Ott, and F. Kossentini, “Error resilience support in H.263+,” *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 8, pp. 867 – 877, Nov. 1998.
- [13] ISO/IEC IS 11172 (MPEG-1), “Coding of moving pictures and associated audio for digital storage media up to about 1.5 Mbits/s,” *Technical Report, Motion Picture Experts Group*, 1993.
- [14] D. L. Gall, “MPEG: a video compression standard for multimedia applications,” *Communications of the ACM*, pp. 46–58, Apr. 1991.
- [15] ISO/IEC DIS 13818 (MPEG-2), “Generic coding of moving pictures and associated audio information,” *Technical Report, Motion Picture Experts Group*, 1994.
- [16] T. Sikora, “The MPEG-4 video standard verification model,” *IEEE Trans. on Circuits and Systems for Video Technology*, pp. 19–31, Feb. 1997.
- [17] T. Wiegand, G. Sullivan, G. Bjntegaard, and A. Luthra, “Overview of the H.264/AVC video coding standard,” *IEEE Trans. on Circuits and Systems for Video Technology*, pp. 560– 576, July 2003.
- [18] G. Sullivan and T. Wiegand, “Video compression - from concepts to the H.264/AVC standard,” *Proceedings of the IEEE*, pp. 18– 31, Jan. 2005.
- [19] H. Malvar, A. Hallapuro, M. Karczewicz, and L. Kerofsky, “Low-complexity transform and quantization in H.264/AVC,” *IEEE Trans. on Circuits and Systems for Video Technology*, pp. 598 – 603, Jul. 2003.

REFERENCES

- [20] P. List, A. Joch, J. Lainema, G. Bjntegaard, and M. Karczewicz, “Adaptive deblocking filter,” *IEEE Trans. on Circuits and Systems for Video Technology*, pp. 614–619, Jul. 2003.
- [21] D. Marpe, H. Schwarz, and T. Wiegand, “Context-based adaptive binary arithmetic coding in the H.264/AVC video compression standard,” *IEEE Trans. on Circuits and Systems for Video Technology*, pp. 620 – 636, Jul. 2003.
- [22] T. K. Tan, K. K. Pang, and K. N. Ngan, “A frequency scalable coding scheme employing pyramid and subband techniques,” *IEEE Trans. on Circuits and Systems for Video Technology*, pp. 203–207, April 1994.
- [23] H. Sun, W. Kwok, and J. W. Zdepski, “Architectures for MPEG compressed bitstream scaling,” *IEEE Trans. on Circuits and Systems for Video Technology*, pp. 191–199, April 1996.
- [24] G. J. Conklin and S. S. Hemami, “A comparison of temporal scalability techniques,” *IEEE Trans. on Circuits and Systems for Video Technology*, pp. 909–919, Sept. 1999.
- [25] R. Aravind, M. Civanlar, and A. Reibman, “Packet loss resilience of MPEG-2 scalable video coding algorithms,” *IEEE Trans. on Circuits and Systems for Video Technology*, pp. 426 – 435, Oct. 1996.
- [26] W. Li, “Overview of fine granularity scalability in MPEG-4 video standard,” *IEEE Trans. on Circuits and Systems for Video Technology*, pp. 301–317, March 2001.
- [27] M. van der Schaar and H. Radha, “A hybrid temporal-SNR fine-granular scalability for internet video,” *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 11, pp. 318–331, March 2001.
- [28] H. Radha, M. Schaar, and Y. Chen, “The MPEG-4 fine-grained scalable video coding method for multimedia streaming over IP,” *IEEE Trans. on Multimedia*, pp. 53–68, March 2001.
- [29] F. Wu, S. Li, and Y. Q. Zhang, “A framework of efficient progressive fine granularity scalable video coding,” *IEEE Trans. on Circuits and Systems for Video Technology*, pp. 332–344, March 2001.

REFERENCES

- [30] M. Schaar and H. Radha, "Motion-compensation based fine-granular scalability (MC-FGS)," in *ISO/IEC JTC1/SC29/WG11, MPEG00/M6475*, Oct. 2000.
- [31] M. Schaar and H. Radha, "Adaptive motion-compensation fine-granular-scalability (AMC-FGS) for wireless video," *IEEE Trans. on Circuits and Systems for Video Technology*, pp. 360–371, June 2002.
- [32] W. Peng and Y. Chen, "Mode-adaptive fine granularity scalability," in *Proceedings of IEEE ICIP 2001*, pp. 993–996, Oct. 2001. Greece.
- [33] H. Huang, C. Wang, and T. Chiang, "A robust fine granularity scalability using trellis-based predictive leak," *IEEE Trans. on Circuits and Systems for Video Technology*, pp. 372–385, June 2002.
- [34] X. Sun, F. Wu, S. Li, W. Gao, and Y.-Q. Zhang, "Macroblock-based progressive fine granularity scalable video coding," in *International Conference on Multimedia and Expo*, Aug. 2001.
- [35] Y. Liu, Z. Li, P. Salama, and E. Delp, "A discussion of leaky prediction based scalable coding," in *IEEE ICME 2003*, pp. 565 – 568, Jul. 2003.
- [36] S. Han and B. Girod, "Robust and efficient scalable video coding with leaky prediction," in *IEEE ICIP 2002*, pp. II-41 – II-44, Sept. 2002.
- [37] Joint Video Team of ITU-T VCEG and ISO/IEC MPEG, "Scalable video coding - working draft 1," *JVT-N020*, Jan. 2005.
- [38] R. Schafer, H. Schwarz, D. Marpe, T. Schierl, and T. Wiegand, "MCTF and scalability extension of H.264/AVC and its application to video transmission, storage, and surveillance," in *SPIE VCIP 2005*, pp. 993–996, Jul. 2005.
- [39] H. Schwarz, D. Marpe, and T. Wiegand, "Analysis of hierarchical B pictures and MCTF," in *IEEE ICME 2006*, pp. 1929–1932, Jul. 2006.
- [40] J. Reichel, H. Schwarz, and M. Wien, "Draft of joint scalable video model (JSVM) 3.0 reference encoding algorithm description," *ISO/IEC JTC1/SC29/WG11 N7311*, Jul. 2005.

REFERENCES

- [41] M. Rabbani and P. W. Jones, *Digital image compression techniques*, vol. TT 7. SPIE Optical Engineering Press, 1991.
- [42] 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*, pp. 878–894, Sept. 2000.
- [43] Z. He and S. K. Mitra, “A unified rate-distortion analysis framework for transform coding,” *IEEE Trans. on Circuits and Systems for Video Technology*, pp. 1221–1236, Dec. 2001.
- [44] ITU-T Draft Recommendation H.263 Version 2 (H.263+), “Video coding for low bitrate communications,” Sept. 1997.
- [45] 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*, pp. 172 – 185, Feb. 1999.
- [46] H.-J. Lee, T. Chiang, and Y.-Q. Zhang, “Scalable rate control for very low bit rate (VLBR) video,” in *Proc. of IEEE ICIP 1997*, pp. 768 – 771, Oct. 1997.
- [47] T. Chiang and Y.-Q. Zhang, “A new rate control scheme using quadratic rate distortion model,” *IEEE Trans. on Circuits and Systems for Video Technology*, pp. 246 – 250, Feb. 1997.
- [48] A. Vetro, H. Sun, and Y. Wang, “MPEG-4 rate control for multiple video objects,” *IEEE Trans. on Circuits and Systems for Video Technology*, pp. 186 – 199, Feb. 1999.
- [49] G. J. Sullivan and T. Wiegand, “Rate-distortion optimization for video compression,” *IEEE Signal Processing Magazine*, pp. 74–90, Nov. 1998.
- [50] Z. Li, F. Pan, K. P. Lim, G. Feng, X. Lin, and S. Rahardja, “Adaptive basic unit layer rate control for JVT,” *ISO/IEC JTC1/SC29/WG11 JVT-G012*, Mar. 2003.
- [51] S. Ma, W. Gao, and Y. Lu, “Rate-distortion analysis for H.264/AVC video coding and its application to rate control,” *IEEE Trans. on Circuits and Systems for Video Technology*, pp. 1533 – 1544, Dec. 2005.

REFERENCES

- [52] B. Girod and N. Farber, "Wireless video," in *Compressed Video over Networks* (M. T. Sun and A. Reibman, eds.), Marcel Dekker, Inc. New York, NY, 2000.
- [53] Z. He, J. Cai, and C. W. Chen, "Joint source channel rate-distortion analysis for adaptive mode selection and rate control in wireless video coding," *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 12, pp. 511–523, June 2002.
- [54] K. Stuhlmuller, N. Farber, M. Link, and B. Girod, "Analysis of video transmission over lossy channels," *IEEE Journal on Selected Areas in Communications*, vol. 18, pp. 1012–1032, June 2000.
- [55] S. B. Wicker, *Error Control Systems for Digital Communication and Storage*. Prentice-Hall, Inc., 1995.
- [56] Y. Wang, S. Wenger, J. Wen, and A. K. Katsaggelos, "Error resilient video coding techniques," *IEEE Signal Processing Magazine*, vol. 17, pp. 61–82, July 2000.
- [57] E. Ayanoglu, R. Pancha, A. Reibman, and S. Talwar, "Forward error control for MPEG-2 video transport in a wireless ATM LAN," *ACM/Baltzer Mobile Networks Applicat.*, vol. 1, pp. 245–258, Dec. 1996.
- [58] S. Lin, D. Costello, and M. Miller, "Automatic-repeat-request error-control schemes," *IEEE Communications Magazine*, vol. 22, pp. 5–17, Dec. 1984.
- [59] M. Khansari, A. Jalali, E. Dubois, and P. Mermelstein, "Low bit-rate video transmission over fading channels for wireless microcellular systems," *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 6, pp. 1–11, Feb. 1996.
- [60] H. Liu and M. E. Zarki, "Adaptive source rate control for real-time wireless video transmission," *ACM/Baltzer Mobile Networks Applicat.*, vol. 3, pp. 49–60, Dec. 1998.
- [61] 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.
- [62] Y. Wang, A. Reibman, and S. Lin, "Multiple description coding for video delivery," *Proceedings of the IEEE*, pp. 57–70, Jan. 2005.

REFERENCES

- [63] J. Apostolopoulos, “Reliable video communication over lossy packet networks using multiple state encoding and path diversity,” in *Proc. of VCIP*, pp. 392–409, 2001.
- [64] Y. Wang, M. Orchard, V. Vaishampayan, and A. Reibman, “Multiple description coding using pairwise correlation transforms,” *IEEE Trans. on Circuits and Systems for Video Technology*, pp. 351–366, March 2001.
- [65] R. van Dyck and D. Miller, “Transport of wireless video using separate, concatenated, and joint source-channel coding,” *Proceedings of the IEEE*, vol. 87, pp. 1734 – 1750, Oct. 1999.
- [66] L. Kondi, F. Ishtiaq, and A. Katsaggelos, “Joint source-channel coding for motion-compensated DCT-based SNR scalable video,” *IEEE Trans. on Image Processing*, vol. 11, pp. 1043– 1052, Sep. 2002.
- [67] V. Stankovic, Y. Charfi, R. Hamzaoui, and Z. Xiong, “Real-time unequal error protection for distortion-optimal progressive image transmission,” in *Proc. of IEEE WCNC 2003*, pp. 317– 321, March 2003.
- [68] K. Stuhlmuller, M. Link, and B. Girod, “Scalable internet video streaming with unequal error protection,” in *Packet Video Workshop 99*, April 1999.
- [69] A. Ortega and K. Ramchandran, “Rate-distortion methods for image and video compression,” *IEEE Signal Processing Magazine*, vol. 15, pp. 23–50, Nov. 1998.
- [70] T. V. Lakshman, A. Ortega, and A. R. Reibman, “VBR video: tradeoffs and potentials,” *The Proceedings of IEEE*, pp. 952–972, May 1998.
- [71] Y. Yue, J. Zhou, Y. Wang, and C. W. Chen, “A novel two-pass VBR coding algorithm for fixed size storage applications,” *IEEE Trans. on Circuits and Systems for Video Technology*, pp. 345–356, March 2001.
- [72] I. M. Pao and M. T. Sun, “Encoding stored video for streaming applications,” *IEEE Trans. on Circuits and Systems for Video Technology*, pp. 199–209, Feb. 2001.

REFERENCES

- [73] H. Song and C.-C. J. Kuo, "Rate control for low-bit-rate video via variable-encoding frame rates," *IEEE Trans. on Circuits and Systems for Video Technology*, pp. 512–521, April 2001.
- [74] X. M. Zhang, A. Vetro, Y. Q. Shi, and H. Sun, "Constant quality constrained rate allocation for FGS-coded video," *IEEE Trans. on Circuits and Systems for Video Technology*, pp. 121–130, Feb. 2003.
- [75] L. Zhao, J. W. Kim, and C.-C. Kuo, "Constant quality rate control for streaming MPEG-4 FGS video," in *IEEE ISCAS 2002*, pp. IV-544 – IV-547, May 2002.
- [76] Q. Wang, Z. Xiong, F. Wu, and S. Li, "Optimal rate allocation for progressive fine granularity scalable video coding," *IEEE Signal Processing Letters*, vol. 9, pp. 33 – 39, Feb. 2002.
- [77] M. Dai, D. Loguinov, and H. Radha, "Analysis of rate-distortion functions and congestion control in scalable internet video streaming," in *ACM NOSSDAV '03*, pp. 60 – 69, Jun. 2003.
- [78] J. Cai, Z. He, and C. W. Chen, "Optimal bit allocation for low bit rate video streaming applications," in *IEEE ICIP 2002*, pp. I-73– I-76, Sept. 2002.
- [79] Z. He, W. Zeng, and C. W. Chen, "Low-pass filtering of rate-distortion functions for quality smoothing in real-time video communication," *IEEE Trans. on Circuits and Systems for Video Technology*, pp. 973 – 981, Aug. 2005.
- [80] T. Chen and Z. He, "Single-pass distortion-smoothing encoding for low bit-rate video streaming applications," in *IEEE ICME*, pp. II-237 – II-240, Jul. 2003.
- [81] B. Xie and W. Zeng, "Sequence-based rate control for constant quality video," in *Proc. of IEEE ICIP*, pp. I-77 – I-80, Sept. 2002.
- [82] J. Lan, W. Zeng, and X. Zhuang, "Operational distortion-quantization curve-based bit allocation for smooth video quality," *Journal of Visual Communication and Image Representation*, pp. 527–543, Aug. 2005.

REFERENCES

- [83] Z. He and S. Mitra, "Optimum bit allocation and accurate rate control for video coding via ρ -domain source modeling," *IEEE Trans. on Circuits and Systems for Video Technology*, pp. 840 – 849, Oct. 2002.
- [84] Z. He and S. K. Mitra, "A linear source model and a unified rate control algorithm for DCT video coding," *IEEE Trans. on Circuits and Systems for Video Technology*, pp. 970 – 982, Nov. 2002.
- [85] T. M. Cover and J. A. Thomas, *Elements of Information Theory*. John Wiley and Sons, Inc., 1991.
- [86] 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, June 1996.
- [87] D. S. Taubman and M. W. Marcellin, "JPEG2000: Standard for interactive imaging," *Proceedings of The IEEE*, vol. 90, pp. 1336–1357, Aug. 2002.
- [88] T. S. Rappaport, *Wireless Communications*. Prentice Hall PTR, 1996.
- [89] V. Chande and N. Farvardin, "Progressive transmission of images over memoryless noisy channels," *IEEE Journal on Selected Areas in Communications*, pp. 850–860, June 2000.
- [90] A. Nosratinia, J. Lu, and B. Aazhang, "Source-channel rate allocation for progressive transmission of images," *IEEE Trans. on Communications*, pp. 186–196, Feb. 2003.
- [91] Z. Wu, A. Bilgin, and M. W. Marcellin, "Unequal error protection for transmission of JPEG2000 codestreams over noisy channels," in *Proc. of IEEE ICIP 2002*, pp. I–213– I–216, Sept. 2002.
- [92] A. E. Mohr, E. A. Riskin, and R. E. Ladner, "Unequal loss protection: graceful degradation of image quality over packet erasure channels through forward error correction," *IEEE Journal on Selected Areas in Communications, special issue on Error-Resilient Image and Video Transmission*, vol. 18, pp. 819–828, June. 2000.

REFERENCES

- [93] J. Kim, R. M. Mersereau, and Y. Altunbasak, “Error-resilient image and video transmission over the Internet using unequal error protection,” *IEEE Trans. on Image Processing*, pp. 121–131, Feb. 2003.
- [94] M. Schaar and H. Radha, “Unequal packet loss resilience for fine-granular-scalability video,” *IEEE Trans. on Multimedia*, pp. 381–394, Dec. 2001.
- [95] X. K. Yang, C. Zhu, Z. Li, G. N. Feng, S. Wu, and N. Ling, “A degressive error protection algorithm for MPEG-4 FGS video streaming,” in *Proc. of IEEE ICIP 2002*, pp. 737–740, Sept. 2002.
- [96] T. Stockhammer, “Progressive video transmission for packet lossy channels exploiting feedback and unequal erasure protection,” in *Proc. of IEEE ICIP 2002*, pp. II–169 – II–172, Sept. 2002.
- [97] H. Yang and L. Lu, “A novel source-channel constant distortion model and its application in error resilient frame-level bit allocation,” in *Proc. of IEEE ICASSP 2004*, pp. III–277–280, May 2004.

Publications

Journal Papers:

- [1] Jianhua Wu, Jianfei Cai and Chang Wen Chen, "Single-pass rate-smoothed video encoding with quality constraint," accepted by IEEE Signal Processing Letters.
- [2] Jianhua Wu, Jianfei Cai, and Chang Wen Chen, "Rate-distortion analysis of leaky prediction based FGS video for constant quality constrained rate adaptation," Journal of Visual Communication and Image Representation, pp. 45-58, vol. 18, 2007.
- [3] Jianfei Cai, Jianhua Wu, King Ng Ngan, and Zhihai He, "Joint mode selection and unequal error protection for bitplane coded video transmission over wireless channels," Journal of Visual Communication and Image Representation, Special Issue on Visual Communication in the Ubiquitous Era, pp. 412-431, vol. 16, 2005.

Conference Papers:

- [1] Jianhua Wu and Jianfei Cai, "Rate-smoothed encoding for real-time video streaming applications," SPIE VCIP 2007, Jan. 2007.
- [2] Jianhua Wu, Jianfei Cai, and Chang Wen Chen, "Constant quality constrained bit allocation for leaky prediction based FGS video streaming," SPIE VCIP 2005, pp. 2212-2219, Jul. 2005.
- [3] Jianhua Wu and Jianfei Cai, "Quality-smoothed encoding for real-time video streaming applications," IEEE ISCE 2005, pp. 445-449, Jun. 2005.

REFERENCES

- [4] Jianhua Wu and Jianfei Cai, "A novel rate-distortion model for leaky prediction based FGS video coding," IEEE Pacific-Rim Conference on Multimedia (PCM2004), pp. 673-680, Nov. 2004.
- [5] Jianhua Wu and Jianfei Cai, "Wireless FGS video transmission using adaptive mode selection and unequal error protection," SPIE VCIP 2004, pp. 1305-1316, Jan. 2004.
- [6] Jianhua Wu and Jianfei Cai, "A simple FGS to single layer transcoding scheme for wireless video streaming," International Workshop on Advanced Image Technology (IWAIT'04), pp. 11-14, Jan. 2004.
- [7] Jianhua Wu and Jianfei Cai, "Using EREC for fast block retrieving in JPEG-coded large images," International Workshop on Advanced Image Technology (IWAIT'04), pp. 185-188, Jan. 2004.