

國立交通大學

電子工程學系電子研究所

博士論文

具強韌細緻架構之可調視訊編碼演算法

Scalable Video Coding Algorithms
with Robust Fine Granularity Structure

研究生：黃項群

指導教授：蔣迪豪

中華民國九十五年十月



具強韌細緻架構之可調視訊編碼演算法
Scalable Video Coding Algorithms
with Robust Fine Granularity Structure

研究生：黃項群
指導教授：蔣迪豪

Student: Hsiang-Chun Huang
Advisor: Dr. Tihao Chiang

國立交通大學
電子工程學系電子研究所
博士論文



A Dissertation
Submitted to Department of Electronics Engineering & Institute of Electronics
College of Electrical and Computer Engineering
National Chiao Tung University
in Partial Fulfillment of the Requirements
for the Degree of Doctor of Philosophy
In
Electronics Engineering

October 2006

Hsinchu, Taiwan, Republic of China

中華民國九十五年十月



具強韌細緻架構之可調視訊編碼演算法

學生：黃項群

指導教授：蔣迪豪教授

國立交通大學電子工程學系電子研究所 博士班

摘要

MPEG-4 標準委員會制訂了細緻可調視訊編碼以進行視訊串流及廣播。細緻可調視訊編碼的加強層使用了幀內預測以及位元層編碼，使其位元流能被截斷於任意位置並提供了細緻的影像畫質調節。因為缺乏幀間預測，MPEG-4 細緻可調視訊編碼之加強層有良好的容錯能力，卻有較差的編碼效率。本論文提出了新的技術以加強幀間預測及編碼效率，同時仍保持良好的錯誤容忍度。這些技術並已被正制定中的 H. 264/AVC 可調視訊編碼標準所採用。

本論文首先提出了強韌細緻可調視訊編碼，利用加強層資訊以及滲漏式預測來改善幀間預測效果。這個方法利用兩個參數，位元層數目 β （介於 0 與最大位元層數目之間）及滲漏預測係數 α （介於 0 與 1 之間），來控制加強層參考幀的產生。 α 與 β 可在不同幀之間調整，以在加強壓縮效率及降低錯誤飄移之間取得平衡。本方法提供了整體且彈性的架構以利更進一步的最佳化調整。加強層的資訊同時也能用來幀間預測基礎層，以進一步加強壓縮效率。實驗結果顯示，在 MPEG-4 的測試條件下，本方法最多提供了超過 4dB 的 PSNR 改善。

本論文更進一步提出了堆疊式強韌細緻可調視訊編碼來改善強韌細緻可調視訊編碼。堆疊式強韌細緻可調視訊編碼簡化了強韌細緻可調視訊編碼的架構，並將其擴展為多層堆疊架構。堆疊式強韌細緻可調視訊編碼可針對不同應用最佳化在多個操作點，並仍然保持強韌細

緻可調視訊編碼的細緻性與容錯能力。我們同時提出一個以宏塊為基礎，最佳化選擇 α 的方法來加強壓縮效率。我們並且提出一個單加強層迴圈解碼結構以簡化解碼端的複雜度。實驗結果顯示，相較於強韌細緻可調視訊編碼，堆疊式強韌細緻可調視訊編碼提供了 0.4 至 3.0dB 的 PSNR 改善。堆疊式強韌細緻可調視訊編碼已被 MPEG 委員會審閱過，並在可調視訊編碼的 Call for Evidence 競賽中被評比為最佳技術之一。

基於前述提出之滲漏式預測以及堆疊式架構，我們再進一步提出強韌可調視訊編碼，以同時提供在影像大小、播放幀數、以及影像畫質三個方面的可調視訊編碼。在影像大小以及影像畫質的可調性上，我們提出一個只增加有限周邊資訊的彈性層間預測方式，以去除層間的冗餘資訊。在影像畫質的可調性上，我們同時支援非細緻可調及細緻可調視訊編碼。其中，在細緻可調視訊編碼的部分，我們擴展了 H.264/AVC 中的 CABAC 技術，使其能提供位元層編碼及細緻可調性。在播放幀數的可調性上，我們提出一個能減少解碼端影像暫存記憶體的实现方法。相較於 H.264/AVC 可調視訊編碼標準，強韌可調視訊編碼提供的 PSNR 改善在 -0.7dB 至 +0.8dB 之間。

總結，本論文提出的強韌細緻可調視訊編碼及堆疊式強韌細緻可調視訊編碼顯著的改善了 MPEG-4 細緻可調視訊編碼的壓縮效果，並仍保持良好的細緻性及容錯能力。此技術也被採用於 H.264/AVC 可調視訊編碼標準。基於前述提出之技術，我們進一步提出強韌可調視訊編碼以同時提供在影像大小、播放幀數、以及影像畫質三個方面的可調性。此技術並提供了能與 H.264/AVC 可調視訊編碼標準相抗衡的壓縮效率。最後，我們還建立了一個影像串流模擬架構以展示可調視訊編碼的應用。

Scalable Video Coding Algorithms with Robust Fine Granularity Structure

Student: Hsiang-Chun Huang

Advisor: Dr. Tihao Chiang

Department of Electronics Engineering & Institute of Electronics
National Chiao Tung University

Abstract

The MPEG-4 committee has defined the MPEG-4 Fine Granularity Scalability (FGS) Profile as a streaming video tool. The MPEG-4 FGS enhancement layer is intra coded with bitplane coding. It can be truncated at any location to provide fine granularity of reconstructed video quality. The lack of temporal prediction at the MPEG-4 FGS enhancement layer leads to inherent robustness at the expense of coding efficiency. In this dissertation, we propose novel techniques to improve the temporal prediction at the enhancement layer so that coding efficiency is superior to the MPEG-4 FGS. The proposed techniques are also adopted in the developing H.264/AVC SVC.

We propose the Robust FGS (RFGS) that utilize enhancement layer information and leaky prediction technique to improve the temporal prediction efficiency. Our approach utilizes two parameters, the number of bitplanes β ($0 \leq \beta \leq$ Maximal number of bitplanes) and the amount of predictive leak α ($0 \leq \alpha \leq 1$), to control the construction of the reference frame at the enhancement layer. These parameters α and β can be selected for each frame to provide tradeoffs between coding efficiency and error drift. Our approach offers a general and flexible framework that allows further optimization. The enhancement layer is also used to predict the base layer for further improvement. Experimental results show over 4 dB PSNR improvements in coding efficiency using the MPEG-4 testing conditions.

We further present Stack Robust FGS (SRFGS) to improve the RFGS performance.

SRFGS simplifies the RFGS architecture and extends it into multi-layer stack architecture. SRFGS can be optimized at several operating points to meet the requirement for various applications, while maintaining the fine granularity and error robustness of RFGS. An optimized macroblock-based alpha adaptation scheme is proposed to improve the coding efficiency. A single-loop enhancement layer decoding scheme is proposed to reduce the decoder complexity. Simulation results show that SRFGS improves the performance of RFGS by 0.4 to 3.0 dB in PSNR. SRFGS has been reviewed by the MPEG committee and ranked as one of the best algorithms in the Call for Evidence on Scalable Video Coding.

Based on the proposed leaky prediction and stack structure, we further propose the Robust Scalable Video Coding (RSVC) to support spatio-temporal and SNR scalability simultaneously. To remove the inter-layer redundancy, a flexible inter-layer prediction with limited overhead is proposed for spatial and SNR scalability. For SNR scalability, both coarse granularity scalability (CGS) and FGS are supported. The H.264/AVC CABAC is extended to support the bitplane coding and FGS. A lower Decoded Picture Buffer (DPB) requirement method is used to implement the temporal scalability. The simulation results show that we have -0.7dB to +0.8dB PSNR difference comparing with the H.264/AVC SVC.

In conclusion, the proposed RFGS and SRFGS architectures significantly improve the coding efficiency of MPEG-4 FGS, while still maintaining the fine granularity and error robustness. The proposed ideas have been adopted in H.264/AVC SVC. Based on leaky prediction and stack structure, we further propose RSVC to support spatio-temporal and SNR scalability simultaneously. RSVC provides comparable performance against H.264/AVC SVC. Finally, we develop a video streaming architecture for mobile WiMAX to show an application scenario of scalable video coding.

Acknowledgement

能完成論文，首先要感謝蔣迪豪老師的指導。蔣老師提供了有趣又有挑戰性的研究題目，讓我在研究討論的過程中逐步學習成長。蔣老師也提供了國際性的研究計畫及競賽，在研究及競爭的過程中，讓我學到難得的經驗，並增加了國際視野。而在學校的研究之外，蔣老師在工作的選擇上也給我許多建議。另外，我也非常謝謝蔣老師這幾年來在各方面給我的幫助及鼓勵。

再來，我要感謝杭學鳴老師。杭老師除了在研究討論時給我很多的指導外，杭老師待人處事的態度，也是我最需要學習的。另外，我也要感謝俊能及文孝兩位學長。兩位學長多年來在研究，論文寫作，以及上台報告各方面，都給我非常多的建議及指導。此外，我也要感謝俊毅，士豪，志鴻，家揚，耀中，以及其他 Commlab 的同學們，在各方面給我的幫助。我也要謝謝 Ambarella 的上司及同事們，在過去幾年裡，不時的幫忙減少或分擔我的工作，讓我有時間繼續學校的研究。

最後，我要謝謝我的母親多年來對我的支持，照顧及教育，讓我能自由自在，無後顧之憂的選擇自己的人生，做自己想做的事。我還要感謝老天爺，總是給我足夠的幸運。

To my mother



Contents

摘要	I
ABSTRACT	III
ACKNOWLEDGEMENT	V
CONTENTS	VII
LIST OF FIGURES	X
LIST OF TABLES	XII
LIST OF NOTATIONS	XIII
CHAPTER 1 INTRODUCTION	1
1.1 OVERVIEW OF DISSERTATION	1
1.1.1 Scalable Video Coding Standard	4
1.1.2 Robust Fine Granularity Scalability (RFGS).....	4
1.1.3 Stack Robust Fine Granularity Scalability (SRFGS).....	5
1.1.4 Relevance to H.264/AVC SVC	6
1.1.5 Robust Scalable Video Coding (RSVC).....	6
1.1.6 Streaming Video Application.....	6
1.2 ORGANIZATION AND CONTRIBUTION	7
CHAPTER 2 SCALABLE VIDEO CODING STANDARD	10
2.1 MPEG-4 FGS.....	10
2.2 H.264/AVC SVC	11
2.2.1 Overview	11
2.2.2 Overall Encoder Structure	12
2.2.3 Temporal Scalability	15
2.2.3.1 Motion Compensated Temporal Filtering.....	15
2.2.3.2 Hierarchical-B Structure	16
2.2.3.3 Adaptive Reference Fine Granularity Scalability	17
2.2.4 SNR Scalability	19
2.2.4.1 Coarse Grain Scalability (CGS).....	19
2.2.4.2 Fine Grain Scalability (FGS).....	19
2.2.5 Spatial Scalability	20
2.2.5.1 Inter-layer Prediction Structure	21
2.2.5.2 Intra Texture Prediction.....	22
2.2.5.3 Motion Prediction	23
2.2.5.4 Residue prediction.....	23
2.2.6 Interlaced Coding	24
2.2.7 Bit stream Extraction and Adaptation.....	25
2.2.7.1 Simple Truncation.....	25
2.2.7.2 Quality Layer Adaptation.....	25
2.2.8 Performance Comparison between H.264/AVC and H.264/AVC SVC	26
2.2.8.1 H.264/AVC SVC with Spatial Scalability Only	26
2.2.8.2 H.264/AVC SVC with SNR Scalability Only.....	27
2.2.8.3 H.264/AVC SVC with Combined Scalability.....	27
2.2.9 Summary	29

CHAPTER 3 ROBUST FINE GRANULARITY SCALABILITY (RFGS)	30
3.1 INTRODUCTION	30
3.2 PREDICTION TECHNIQUES OF THE ENHANCEMENT LAYER	31
3.2.1 Leaky Prediction.....	32
3.2.2 Partial Prediction	32
3.2.3 Adaptive Mode Selection.....	34
3.3 THE RFGS SYSTEM ARCHITECTURE.....	35
3.3.1 Functional Description	40
3.3.2 Leaky and Partial Prediction.....	40
3.3.3 Analysis of Error Propagation	42
3.3.4 High Quality Reference in Base Layer	46
3.3.5 Rate Control for the Enhancement Layer	48
3.4 THE SELECTION OF THE RFGS PARAMETERS	49
3.4.1 Selection of the Leaky Factor	49
3.4.2 The Number of Bitplanes	53
3.5 EXPERIMENT RESULT AND ANALYSES	54
3.5.1 The Testing Conditions.....	54
3.5.2 Performance Comparisons.....	56
3.5.3 Test for Error Recovery Capability.....	58
3.6 SUMMARY	61
CHAPTER 4 STACK ROBUST FINE GRANULARITY SCALABILITY (SRFGS)	63
4.1 INTRODUCTION	63
4.2 SIMPLIFIED RFGS PREDICTION SCHEME	64
4.3 ENHANCED PREDICTION ARCHITECTURE USING STACK CONCEPT.....	69
4.4 THE STACK RFGS SYSTEM ARCHITECTURE	72
4.4.1 Functional Description	72
4.4.2 Optimized macroblock-based alpha adaptation	77
4.4.3 Prediction scheme of B-frame	77
4.4.4 Stack RFGS with single-loop enhancement layer decoder.....	78
4.5 EXPERIMENT RESULTS AND ANALYSES	81
4.6 SUMMARY	85
CHAPTER 5 RELEVANCE TO THE H.264/AVC SVC	86
5.1 INTRODUCTION	86
5.2 RFGS IN H.264/AVC SVC	86
5.3 SRFGS IN H.264/AVC SVC.....	88
5.4 SUMMARY	91
CHAPTER 6 ROBUST SCALABLE VIDEO CODING	92
6.1 INTRODUCTION	92
6.2 THE RSVC SYSTEM ARCHITECTURE	93
6.3 SPATIAL SCALABILITY AND SNR SCALABILITY	95
6.3.1 Texture Prediction.....	95
6.3.2 Prediction-Information Prediction.....	96
6.3.3 Residue Prediction.....	97
6.3.4 Skip Mode	97
6.4 FINE GRANULARITY SCALABILITY (FGS).....	98
6.4.1 Entropy Coding	98
6.4.2 Leaky Prediction.....	102
6.5 TEMPORAL SCALABILITY	102
6.6 BITSTREAM EXTRACTION AND ERROR CONCEALMENT.....	104
6.6.1 Bitstream Extraction.....	104
6.6.2 Error Concealment.....	105
6.7 SIMULATION RESULTS.....	105
6.7.1 Spatial Scalability	108
6.7.2 SNR Scalability	108
6.7.3 Combined Scalability	110
6.8 SUMMARY	112

CHAPTER 7 CONCLUSION	113
APPENDIX A STREAMING VIDEO APPLICATION BASED ON H.264/AVC SVC FOR MOBILE WIMAX	116
A.1 INTRODUCTION	116
A.2 SYSTEM ARCHITECTURE	117
A.1.1 Overview of the System Architecture	117
A.1.2 The H.264/AVC SVC Streaming Server.....	118
A.1.3 The Mobile WiMAX Simulation Platform	120
A.3 SIMULATION RESULTS.....	121
A.1.4 Test Conditions	121
A.1.5 Simulation results	123
A.4 SUMMARY.....	126
BIBLIOGRAPHY	128
CURRICULUM VITAE	131



List of Figures

FIGURE 1.1. AN EXAMPLE OF SVC (A) APPLICATION SCENARIO (B) BIT STREAM EXTRACTION, AND (C) THE DECODED VIDEO	3
FIGURE 2.1. MPEG-4 FGS ENCODER STRUCTURE	11
FIGURE 2.2. H.264/AVC SVC ENCODER STRUCTURE WITH THREE SPATIAL/SNR LAYERS.....	13
FIGURE 2.3. TEMPORAL DECOMPOSITION.....	14
FIGURE 2.4. CONFIGURATION OF INTER-LAYER PREDICTION	21
FIGURE 2.5. PERFORMANCE COMPARISON BETWEEN H.264/AVC AND H.264/AVC SVC.....	28
FIGURE 3.1. PARTIAL INTER PREDICTION MODE FOR CODING THE BITPLANES AT THE ENHANCEMENT LAYER USING RFGS CODING FRAMEWORK. EACH FRAME HAS THE FLEXIBILITY TO SELECT THE NUMBER OF BITPLANES USED TO GENERATE THE HIGH QUALITY REFERENCE FRAME. FOR EXAMPLE, THE FIRST FRAME USES THREE BITPLANES TO COMPUTE THE HIGH QUALITY REFERENCE FRAME.	33
FIGURE 3.2. CHANNEL BANDWIDTH VARIATION PATTERN FOR THE DYNAMIC TEST DEFINED IN THE MPEG DOCUMENT M8002 [19].....	35
FIGURE 3.3. DIAGRAM OF THE RFGS ENCODER FRAMEWORK. THE SHADOWED BLOCKS ARE THE NEW MODULES FOR RFGS AS COMPARED TO MPEG-4 BASELINE FGS.....	37
FIGURE 3.4. DIAGRAM OF THE RFGS DECODER FRAMEWORK. THE SHADOWED BLOCKS ARE THE NEW MODULES FOR RFGS AS COMPARED TO MPEG-4 BASELINE FGS.....	38
FIGURE 3.5. ILLUSTRATION OF A TRANSMISSION SCENARIO WITH CORRUPTED OR LOST FRAME FOR A VIDEO STREAM OF N FRAMES, WHERE THE ENHANCEMENT LAYER OF THE I -TH FRAME IS ASSUMED TO BE LOST.	45
FIGURE 3.6 THE VISUAL QUALITIES OF THE RECONSTRUCTED PICTURES USING THE PROPOSED RFGS RATE CONTROL SCHEME. WE PROVIDE THE QUALITY OF THE FIRST 60 FRAMES OF THE FOREMAN BITSTREAM. THE BASE LAYER BITSTREAM IS ENCODED WITH A BITRATE OF 256KBPS. THE ENHANCEMENT LAYER BITSTREAM IS TRUNCATED AT SEVERAL BITRATES TO UNDERSTAND THE VARIATION IN PSNR FOR VARIOUS CHANNEL BANDWIDTHS. THE RESULTS SHOW THAT THE PSNR VARIATION IS SMALLER THAN 2 dB AT VARIOUS BITRATE.	48
FIGURE 3.7. THE LINEAR DEPENDENCY BETWEEN NEAR-OPTIMAL LEAK FACTOR AND THE PICTURE QUALITY IN PSNR OF THE BASE LAYER. THE FRAMES WITHIN FIVE GOVs, WHERE EACH HAS 60 FRAMES, ARE USED FOR THE SIMULATIONS WITH THE FOUR SEQUENCES, NAMELY AKIYO, CARPHONE, FOREMAN, AND COASTGUARD.....	50
FIGURE 3.8. PSNR VERSUS BITRATE COMPARISON BETWEEN FGS, RFGS AND SINGLE LAYER CODING SCHEMES FOR THE Y COMPONENT OF THE FOREMAN SEQUENCE, WHERE β IS 3. WE USE THREE DIFFERENT CODING SCHEMES INCLUDING ‘RFGS1’, ‘RFGS2_NearOpt’, AND ‘RFGS2_LM’ IN THE EXPERIMENTS. ‘RFGS1’ USE THE RFGS ALGORITHM FOR THE ENHANCEMENT LAYER ONLY. ‘RFGS2’ USES THE RFGS ALGORITHM FOR BOTH THE ENHANCEMENT AND BASE LAYERS. ‘NearOpt’ MEANS THE RESULT OF THE NEAR-OPTIMAL APPROACH AND ‘LM’ MEANS THE RESULTS USING THE PROPOSED LINEAR MODEL.	51
FIGURE 3.9 PSNR VERSUS BITRATE COMPARISON BETWEEN FGS, RFGS AND SINGLE LAYER CODING SCHEMES FOR THE Y COMPONENT OF THE COASTGUARD SEQUENCE, WHERE β IS 3. WE USE THREE DIFFERENT CODING SCHEMES INCLUDING ‘RFGS1’, ‘RFGS2_NearOpt’, AND ‘RFGS2_LM’ IN THE EXPERIMENTS. ‘RFGS1’ USE THE RFGS ALGORITHM FOR THE ENHANCEMENT LAYER ONLY. ‘RFGS2’ USES THE RFGS ALGORITHM FOR BOTH THE ENHANCEMENT AND BASE LAYERS. ‘NearOpt’ MEANS THE RESULT OF THE NEAR-OPTIMAL APPROACH AND ‘LM’ MEANS THE RESULTS USING THE PROPOSED LINEAR MODEL.	52
FIGURE 3.10. PSNR VERSUS BITRATE COMPARISON BETWEEN FGS, RFGS AND SINGLE LAYER CODING	

SCHEMES FOR THE Y COMPONENT OF THE AKIYO SEQUENCE, WHERE β IS 3. WE USE THREE DIFFERENT CODING SCHEMES INCLUDING ‘RFGS1’, ‘RFGS2_NEAROPT’, AND ‘RFGS2_LM’ IN THE EXPERIMENTS. ‘RFGS1’ USES THE RFGS ALGORITHM FOR THE ENHANCEMENT LAYER ONLY. ‘RFGS2’ USES THE RFGS ALGORITHM FOR BOTH THE ENHANCEMENT AND THE BASE LAYERS. ‘NEAROPT’ MEANS THE RESULT OF THE NEAR-OPTIMAL APPROACH AND ‘LM’ MEANS THE RESULTS USING THE PROPOSED LINEAR MODEL.	52
FIGURE 3.11. PSNR VERSUS BITRATE COMPARISON BETWEEN VARIOUS VALUES OF RFGS PARAMETER β FOR THE Y COMPONENT OF THE FOREMAN SEQUENCE, WHERE THE LEAK FACTOR α IS SELECTED WITH THE PROPOSED LINEAR MODEL.	53
FIGURE 3.12 PSNR VERSUS BITRATE COMPARISON BETWEEN RFGS AND PFGS FOR THE Y COMPONENT OF THE COASTGUARD AND FOREMAN SEQUENCES IN CIF FORMAT USING THE TEST CONDITION A IN THE MPEG DOCUMENT M6779 [18]. FOR RFGS, β IS 3.	55
FIGURE 3.13 PSNR VERSUS BITRATE COMPARISON BETWEEN RFGS AND PFGS FOR THE Y COMPONENT OF THE COASTGUARD AND FOREMAN SEQUENCES IN CIF FORMAT USING THE TEST CONDITION B FROM THE MPEG DOCUMENT M6779 [18]. FOR THE RFGS, β IS 3.	56
FIGURE 3.14. SAMPLE BANDWIDTH PROFILE TO TEST THE ERROR RECOVERY CAPABILITY OF THE RFGS TECHNIQUE.	58
FIGURE 3.15. THE ERROR ATTENUATION IN PSNR FOR THE Y COMPONENT OF THE AKIYO SEQUENCE UNDER DIFFERENT α IN THE RFGS1 FRAMEWORK, WHERE THE PAIR OF THE VALUES INDICATES THE PREDICTION MODE PARAMETERS (α, β)	59
FIGURE 3.16. THE ERROR ATTENUATION IN PSNR FOR THE Y COMPONENT OF THE FOREMAN SEQUENCE USING THE RFGS2_LM FRAMEWORK. ALL THE CURVES DENOTE TRUNCATION OF THE ENHANCEMENT LAYER BITSTREAM AT 1024KBPS. FOR THE CURVE LABELED ‘RFGS2_LM DROP 1’, THE FIRST FRAME OF EACH GOV IS DROPPED. FOR THE CURVE LABELED ‘RFGS2_LM DROP 7’, THE FIRST SEVEN FRAMES OF EACH GOV ARE DROPPED. FOR THE CURVE LABELED ‘RFGS2_LM NONE DROP’, NO FRAME IS DROPPED. THE CURVE LABELED ‘BASELINEFGS_DROP=7’ IS THE BASELINE FGS WITH THE FIRST 7 FRAMES OF EACH GOV DROPPED.	60
FIGURE 3.17. THE RELATIONSHIP BETWEEN THE LEAK FACTOR α AND THE TIME CONSTANT τ FOR THE ERROR ATTENUATION. FOR EACH CURVE, β IS 3.	61
FIGURE 3.18. THE COMPARISON OF VISUAL QUALITY IN PSNR BETWEEN FGS AND SINGLE LAYER APPROACHES WITH THE DYNAMIC TEST CONDITION AS DEFINED IN THE MPEG DOCUMENT M8002 [19].	62
FIGURE 4.1 THE ORIGINAL RFGS ENCODER.	66
FIGURE 4.2 THE SIMPLIFIED RFGS ENCODER.	68
FIGURE 4.3 SRFGS PREDICTION CONCEPT.	71
FIGURE 4.4 DIAGRAM OF THE SRFGS ENCODER FRAMEWORK.	73
FIGURE 4.5 DIAGRAM OF THE SRFGS DECODER FRAMEWORK.	75
FIGURE 4.6 THE SRFGS ENHANCEMENT LAYER BITSTREAM FORMAT.	76
FIGURE 4.7 DIAGRAM OF THE SRFGS SINGLE-LOOP ENHANCEMENT LAYER DECODER FRAMEWORK.	80
FIGURE 4.8 PSNR VERSUS BITRATE COMPARISON BETWEEN SRFGS, RFGS AND AVC CODING SCHEMES FOR THE Y COMPONENT.	83
FIGURE 6.1. RSVC ENCODER STRUCTURE WITH THREE SPATIAL/SNR LAYERS.	94
FIGURE 6.2. PROBABILITY DISTRIBUTION OF THE RESIDUE VALUE CAN BE APPROXIMATE BY A LAPLACIAN MODEL, WHERE SMALLER VALUE HAS LARGER PROBABILITY.	100
FIGURE 6.3. HIERARCHICAL PREDICTION STRUCTURE IMPLEMENTATION.	103
FIGURE 6.4. SIMULATION RESULTS FOR SPATIAL SCALABILITY.	107
FIGURE 6.5. SIMULATION RESULTS FOR CGS AND FGS ENTROPY IN RSVC.	109
FIGURE 6.6. SIMULATION RESULTS FOR SNR SCALABILITY WITH FGS.	110
FIGURE 6.7. SIMULATION RESULTS FOR COMBINED SCALABILITY.	111
FIGURE A.1 SVC VIDEO STREAMING ARCHITECTURE.	118
FIGURE A.2 THE TRANSMITTED GOPS IN EACH REPORT PERIOD.	119
FIGURE A.3 THE SDU FAILURE RATE IN 1-CONNECTION SERVICE.	123
FIGURE A.4 THE DATA RATE IN 1-CONNECTION SERVICE.	123
FIGURE A.5 THE SDU FAILURE RATE IN 2-CONNECTION SERVICE.	125
FIGURE A.6 THE DATA RATE IN 2-CONNECTION SERVICE.	125
FIGURE A.7 THE PSNR RESULTS OF THE STREAMING SERVICES.	126

List of Tables

TABLE 3.1. TERMINOLOGY OF THE RFGS CODING FRAMEWORK.....	39
TABLE 4.1 THE VALUE OF (A, B) USED IN THE SIMULATION.....	82
TABLE A.1 THE AVERAGE BITRATE OF THE SVC BITSTREAM AT VARIOUS SPATIAL-SNR AND TEMPORAL RESOLUTIONS	121



List of Notations

$(\cdot)_{mc}$	Operation of motion compensation
α	The leaky factor that multiply on the enhancement layer information
β	The number of enhancement layer bitplane used for inter prediction
i	The time index of the image.
$BLPI$	Predicted base layer frame that is generated by motion compensation from the base layer frame buffer.
$ELPI$	Predicted frame of the enhancement layer that is generated by motion compensation from the enhancement layer frame buffer.
$MCFD_{BL}$	Motion compensated frame difference of the base layer, which is the difference between $BLPI$ and the original image.
$MCFD_{EL}$	Motion compensated frame difference of the enhancement layer which the difference between $ELPI$ and the original image.
$HQRI$	High quality reference image, which is stored in the enhancement layer frame buffer to generate the high quality prediction image $ELPI$.
$ELRI$	Enhancement layer reconstructed image, which is the summation of $ELPI$, \hat{B} , and \hat{D} . $ELRI$ will be processed by the leaky factor to generate the $HQRI$.
F	The original image before encoding.
B	The base layer reconstructed image, which is the summation of $BLPI$ and \hat{B} . B will be stored in the base layer frame buffer.
D	The final residual used at the enhancement layer prediction loop

in the encoder. $(B + \alpha D)$ will be stored at the enhancement layer frame buffer of the encoder.

\hat{B} Coded DCT coefficients of frame $MCFD_{BL}$. The \hat{B} before de-quantization will be compressed as the base layer bitstream.

\hat{D} Difference signal between $MCFD_{EL}$ and \hat{B} for P -pictures or $MCFD_{BL}$ and \hat{B} for I -pictures and B -pictures. \hat{D} will be compressed as the enhancement layer bitstream.

\check{D} The received \hat{D} in the decoder side. Since there may be truncation or error during the transmission of enhancement layer bitstream, \hat{D} and \check{D} may be different.

$\Delta\hat{D}$ The difference between \hat{D} and \check{D} .

\tilde{D} The reconstructed D in the decoder side. $(B + \alpha\tilde{D})$ will be stored at the enhancement layer frame buffer of the decoder.

QE Quantization error.



CHAPTER 1

Introduction

1.1 Overview of Dissertation

The delivery of multimedia information to mobile device over wireless channels and/or Internet is a challenging problem because multimedia transportation suffers from bandwidth fluctuation, random errors, burst errors and packet losses [10]. Thus, the MPEG-4 committee has adopted various techniques to address the issue of error-resilient delivery of video information for multimedia communications. However, it is even more challenging to simultaneously stream or multicast video over Internet or wireless channels to a wide variety of devices where it is impossible to optimize video quality for a particular device, bitrate and channel conditions. The compressed video information is lost due to congestion, channel errors and transport jitters. The temporal predictive nature of most compression technology causes the undesirable effect of error propagation.

To address the broadcast or Internet multicast applications, the ideas of Scalable Video Coding (SVC) is proposed. The SVC provides a single bitstream that can be easily adapted to support various bandwidths and clients. It can be used for various applications such as multi-resolution content analysis, content adaptation, complexity adaptation and bandwidth adaptation. For example, when the video is transported over error-prone channels with fluctuated bandwidth for Internet or wireless visual

communications, the clients, consisting of various devices, requires different processing power and spatio-temporal resolutions. To serve diversified clients over heterogeneous networks, the SVC allows on-the-fly adaptation in the spatio-temporal and quality dimensions according to the network conditions and receiver capabilities. During transmission, the server or router truncates the bit stream to match the available bandwidth. Moreover, the client can skip parts of the received bit stream to match its capability in execution cycles and display dimension.

Figure 1.1 illustrates an application scenario for SVC. In Figure 1.1 (a), the system contains 3 devices including server, router, and wireless access point with different connection speeds. Multiple clients are connected to the networks. The SVC bit stream has 1) 2 spatial resolutions: Standard Definition (SD, 704x576) and Common Intermediate Format (CIF, 352x288); 2) 3 temporal resolutions: 60 frames per second (fps), 30 fps, and 15 fps; and 3) 3 Signal-to-Noise-Ratio (SNR) layers for each spatial resolution. Figure 1.1 (b) shows the bit stream structure for each connection. The bit stream consists of multiple pictures and each picture contains several spatial and quality resolutions. Initially, the video server retains only the first three SNR layers at the CIF resolution and the first and part of the second SNR layers at the SD resolution to match the 4 Mbps bandwidth between the video server and the router. To match the 3Mbps bandwidth between the router and the wireless access point, the router discards the bit stream for the second SNR layer at the SD resolution and the additional temporal resolutions for 60 fps. Similarly, the two wireless clients of lower complexity and display resolution are supported with further truncation. The spatio-temporal pyramid is illustrated in Figure 1.1 (c).

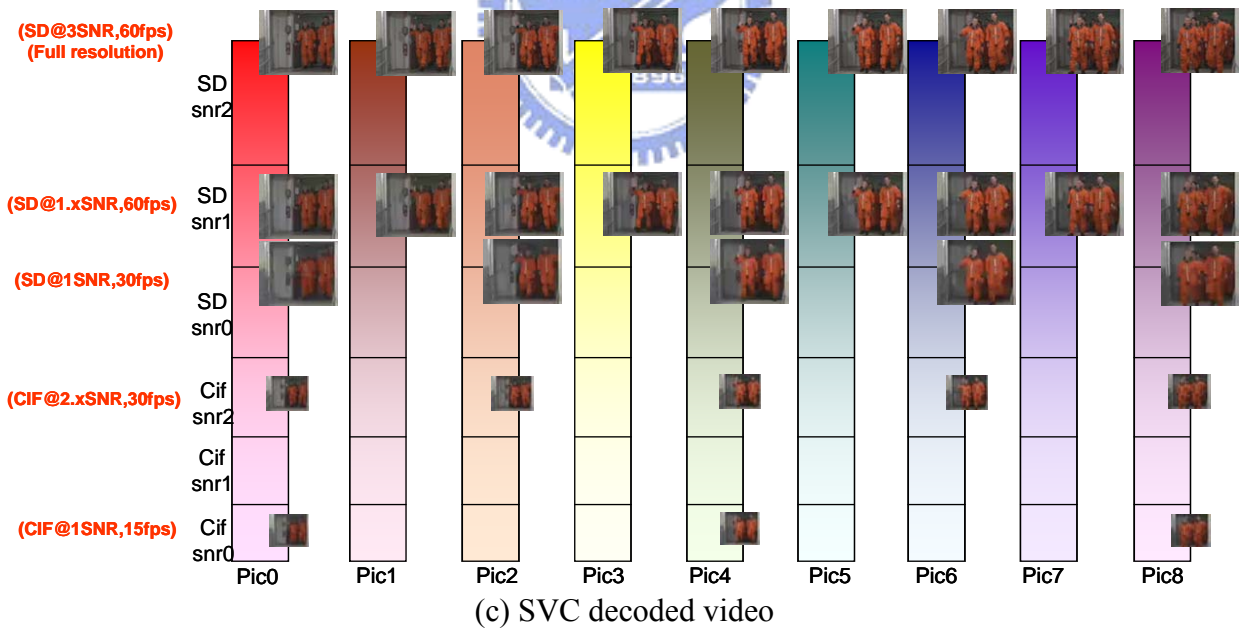
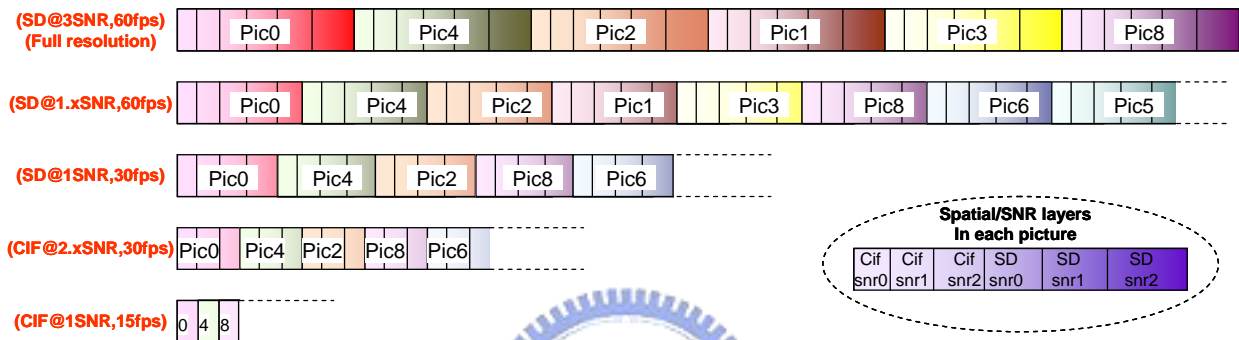
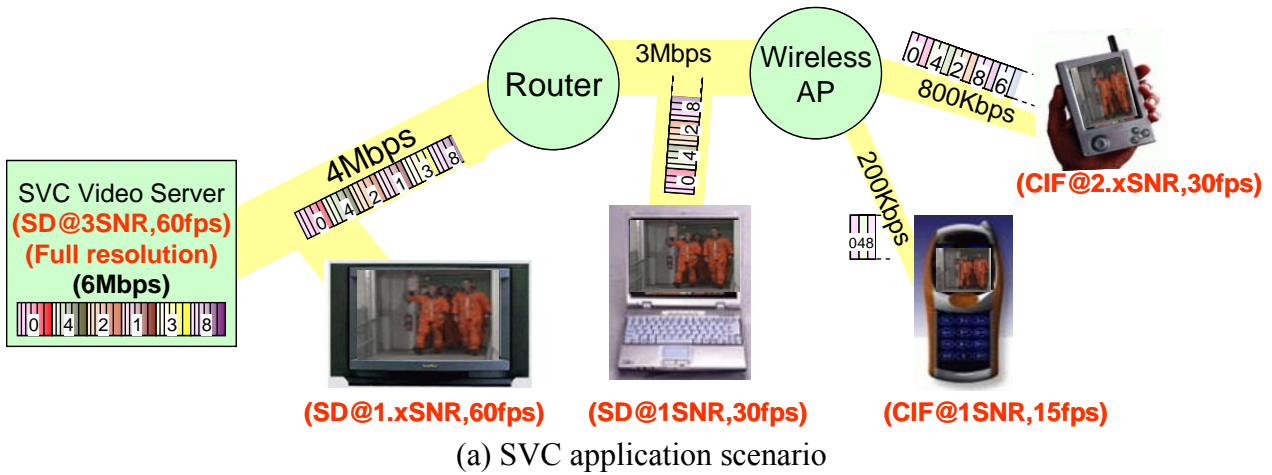


Figure 1.1. An example of SVC (a) application scenario (b) bit stream extraction, and (c) the decoded video

1.1.1 Scalable Video Coding Standard

There are two scalable video coding standards developed in these years. The ISO/IEC MPEG-4 committee defined the Fine Granularity Scalability (FGS) that provides a DCT-based scalable approach in a layered fashion. The base layer is coded by a non-scalable MPEG-4 advanced simple profile (ASP) while the enhancement layer is intra coded with embedded bit plane coding to achieve fine granular scalability. The lack of temporal prediction at the FGS enhancement layer leads to inherent robustness at the expense of coding efficiency.

To further improve the coding efficiency of SVC, recently the ISO/IEC MPEG and ITU-T VCEG form the Joint Video Team (JVT) to develop the scalable video coding amendment of the H.264/AVC standard [1][2][3] (refer to as “H.264/AVC SVC” in this dissertation). The H.264/AVC SVC technology consists of hierarchical-B structure with leaky prediction. To enhance coding efficiency among coding layers, it adopts adaptive inter-layer prediction techniques including intra texture, motion, and residue predictions. The constrained inter-layer prediction is used for reduced decoder complexity. A cyclic block coding is used for SNR scalability with better subjective quality.

1.1.2 Robust Fine Granularity Scalability (RFGS)

The lack of temporal prediction at the MPEG-4 FGS enhancement layer leads to inherent robustness at the expense of coding efficiency. Our goal is constructing a prediction structure that utilizes the enhancement layer information to improve the prediction efficiency, while still maintaining the robustness when the enhancement layer bitstream is truncated.

We proposed the Robust FGS (RFGS) that utilize the leaky prediction concept to improve the temporal prediction efficiency while keeping the features of fine granularity and robustness of MPEG-4 FGS. RFGS multiplies the enhancement layer

temporal prediction information by a leaky factor α , where $0 \leq \alpha \leq 1$. With utilizing the enhancement layer information, the prediction efficiency improved significantly. When error occurs in the enhancement layer, it is multiplied with the leaky factor α every time when forming the temporal prediction frames. After several iterations, the error is attenuated to zero and no longer drift. RFGS further provides another factor β to control the number of bit planes used in the enhancement layer prediction loop. These parameters α and β can be selected for each frame to provide tradeoffs between coding efficiency and error drift. To further improve the coding efficiency, RFGS can also use the enhancement layer information to predict the base layer.

Our experimental results show over 4 dB improvements in coding efficiency using the MPEG-4 testing conditions.

1.1.3 Stack Robust Fine Granularity Scalability (SRFGS)

In the RFGS approach, Larger β leads to more enhancement layer information used for temporal prediction. With the removal of more temporal redundancy, larger β provides better performance when all the reference bit planes are fully reconstructed. However, larger β may lead to larger drifting error at lower bitrate as less amount of required reference information is available for motion compensation. On the contrary, smaller β reduce the drift at lower bitrate at the expense of coding efficiency because the bit planes after β effectively become intra-coded with less coding performance.

We propose the Stack RFGS (SRFGS) to solve the problem. In SRFGS, the RFGS architecture is extended to multi-layer stack architecture. Each layer has its own prediction loops. The error in a layer will not affect the data in other layers. This error localization feature reduces the drifting error because when the higher enhancement layer information is truncated, the lower enhancement layer still can be decoded

correctly. The simulation results show that SRFGS can improve the performance of RFGS by 0.4 to 3.0 dB in PSNR.

1.1.4 Relevance to H.264/AVC SVC

Although the RFGS and SRFGS framework were originally developed based on the MPEG-4 FGS structure, the same prediction structure can also be applied for H.264/AVC SVC. In H.264/AVC SVC, the RFGS prediction structure is adopted and extended to adapt the leaky factor at the coefficient level. The SRFGS prediction structure is adopted and modified to reduce the decoder complexity. The simulation results show that the RFGS and SRFGS prediction structure have up to 4dB and 2dB PSNR improvement in the H.264/AVC SVC, respectively.

1.1.5 Robust Scalable Video Coding (RSVC)

Based on the proposed leaky prediction and stack structure, we further propose the Robust Scalable Video Coding (RSVC) to support spatio-temporal and SNR scalability simultaneously. To remove the inter-layer redundancy, a flexible inter-layer prediction with limited overhead is proposed to for the spatial and SNR scalability. For SNR scalability, both coarse granularity scalability (CGS) and FGS are supported. The H.264/AVC CABAC is extended to support the bitplane coding and FGS. A lower Decoded Picture Buffer (DPB) requirement method is used to implement the temporal scalability. The simulation results show we have -0.7dB to +0.8dB PSNR difference comparing with the developing H.264/AVC SVC.

1.1.6 Streaming Video Application

To demonstrate the application scenario of SVC, we further establish a video streaming architecture based on H.264/AVC SVC for mobile WiMAX. The performance of SVC and non-SVC using both single and multiple connection WiMAX services are studied.

1.2 Organization and Contribution

In this thesis, we propose the Robust FGS (RFGS) to improve the coding efficiency of the MPEG-4 FGS. We further develop the Stack RFGS (SRFGS) to improve the RFGS performance. The utilization of these techniques in the H.264/AVC SVC is also described. We then develop the Robust Scalable Video Coding (RSVC) to support spatio-temporal and SNR scalability simultaneously. The details of each part are organized as follows:

- Chapter 2 introduces the MPEG-4 FGS and the H.264/AVC SVC.
- Chapter 3 discusses the problem in the MPEG-4 FGS and details the RFGS architecture. RFGS utilize enhancement layer information and leaky prediction technique to improve the coding efficiency while maintaining the fine granularity and error robustness of MPEG-4 FGS. Our contributions of this works are:
 - We construct the prediction structure that utilizes the leaky prediction to control the drifting error. The structure offers a general and flexible framework that allows further optimization.
 - We provide an adaptive technique to select the parameter α and β , which yields an improved performance as compared to that of fixed parameters.
 - We also applied the enhancement layer information in the prediction of the base layer to further improve the coding efficiency.
 - Our experimental results show over 4 dB PSNR improvements in coding efficiency using the MPEG-4 testing conditions.
 - The RFGS paper has been cited more than 40 times in Google Scholar.

- Chapter 4 describes the SRFGS architectures. It uses a multiple-loop stack structure to improve the performance of RFGS. The contributions in SRFGS are:
 - We firstly simplified the RFGS structure to reduce the complexity and to reveal the nature of RFGS prediction concept.
 - We then extend the RFGS architecture into multi-layer stack architecture. The SRFGS can be optimized at several operating points to meet the requirement for various applications, while maintaining the fine granularity and error robustness of RFGS.
 - We extend the leaky factor adaptation into macroblock level. An optimized macroblock-based leaky factor adaptation scheme is proposed to improve the coding efficiency.
 - A single-loop enhancement layer decoding scheme is proposed to reduce the decoder complexity.
 - The simulation results show that SRFGS can improve the performance of RFGS by 0.4 to 3.0 dB in PSNR.
 - The SRFGS has been reviewed by the MPEG committee and ranked as one of the best algorithms according to the subjective testing in the Report on Call for Evidence on Scalable Video Coding

- Chapter 5 shows the application scenarios of the RFGS and SRFGS techniques based on H.264/AVC SVC. The applications include:
 - The RFGS leaky prediction structure is used for the anchor pictures with a modification that adapts the leaky factor at coefficient level.
 - The SRFGS stack structure is also utilized for the anchor pictures

with modifications to reduce the decoder complexity.

- Chapter 6 describes the RSVC architectures. Based on the leaky prediction and stack structure, RSVC support spatio-temporal and SNR scalability simultaneously. The contributions in RSVC are:
 - We extend the stack structure to support spatial scalability. A flexible inter-layer prediction with limited overhead is proposed to adaptively remove the inter-layer redundancy.
 - We extend the H.264/AVC CABAC to support bitplane coding and FGS.
 - We efficiently implement the hierarchical temporal prediction structure in H.264/AVC to support temporal scalability with limited Decoded Picture Buffer (DPB) requirement.
 - Our simulation results show that as compared to the current H.264/AVC SVC the RSVC has -0.2 to +0.8dB PSNR gain at spatial scalability, 0.7dB PSNR gain at SNR scalability, and -0.7 to +0.3dB PSNR gain at combined scalability.
- Chapter 7 concludes the thesis.
- Appendix A shows a streaming video application for SVC.
 - We establish a video streaming architecture to show an application scenario of SVC. A streaming server is developed to adapt the H.264/AVC SVC bitstream for the mobile WiMAX.

CHAPTER 2

Scalable Video Coding Standard

In this chapter, we introduce two scalable video coding standards that related to the thesis. The works of the thesis are originally developed based on the MPEG-4 FGS standard, but can also be used on the H.264/AVC. Recently, the H.264/AVC SVC is developing and has utilized the proposed ideas in this thesis. Both of these two standards are introduced in this section.

2.1 MPEG-4 FGS

To address the broadcast or Internet multicast applications, the MPEG-4 committee develops the Fine Granularity Scalability (FGS) Profile [6] that provides a scalable approach for streaming video applications. As shown in Figure 2.1, the MPEG-4 FGS representation starts by separating the video frames into two layers with identical spatial resolutions, which are referred to as the base layer and the enhancement layer. The bitstream at base layer is coded by a non-scalable MPEG-4 advanced simple profile (ASP) while the enhancement layer is obtained by coding the difference between the original DCT coefficients and the coarsely quantized base layer coefficients in a bitplane-by-bitplane fashion [10]. The FGS enhancement layer can be truncated at any location, which provides fine granularity of reconstructed video quality proportional to the number of bits actually decoded. There is no temporal prediction for the FGS

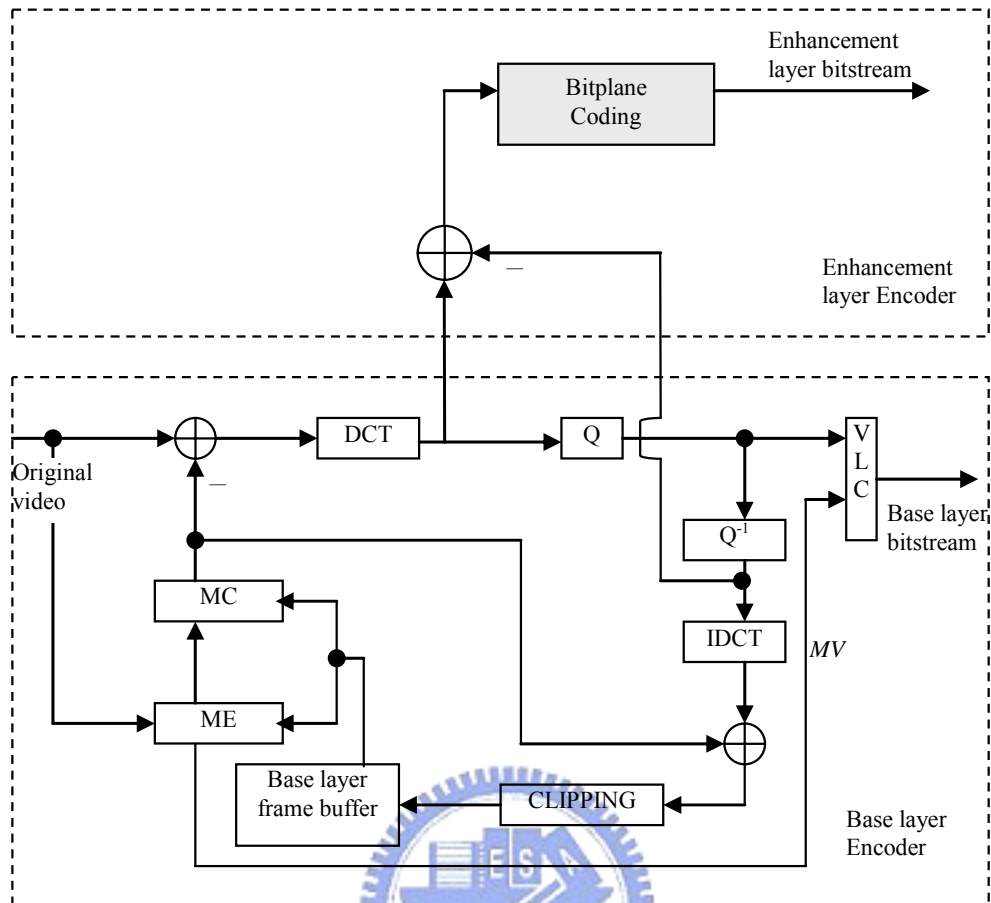


Figure 2.1. MPEG-4 FGS encoder structure

enhancement layer, which provides an inherent robustness for the decoder to recover from any errors. However, the lack of temporal dependency at the FGS enhancement layer decreases the coding efficiency as compared to that of the single layer non-scalable scheme defined in [11].

2.2 H.264/AVC SVC

2.2.1 Overview

To further improve the coding efficiency of SVC and achieve flexible visual content adaptation for multimedia communications, the ISO/IEC MPEG and ITU-T

VCEG form the Joint Video Team (JVT) to develop a scalable video coding standard based on the H.264/AVC standard [1][2][3] (referred to as H.264/AVC SVC in the following). The H.264/AVC SVC standard receives worldwide industrial support and will be elevated to Final Draft International Standard in January 2007.

The H.264/AVC SVC technology consists of hierarchical-B structure with leaky prediction. To enhance coding efficiency among coding layers, it adopts adaptive inter-layer prediction techniques including intra texture, motion, and residue predictions. The constrained inter-layer prediction is used for reduced decoder complexity. A cyclic block coding is used for SNR scalability with better subjective quality.

In this section, we will provide an overview of these technologies and a comparison of coding efficiency between H.264/AVC and H.264/AVC SVC. The rest of this paper is organized as follows: Section 2.2.2 describes the encoder structure of H.264/AVC SVC. Sections 2.2.3 through 2.2.5 examines temporal, SNR, and spatial scalability. Section 2.2.6 and 2.2.7 illustrates the on-going interlaced representation and bit-stream adaptation. Section 2.2.8 compares the coding efficiency between non-scalable H.264/AVC and H.264/AVC SVC. Section 2.2.9 gives a summary of H.264/AVC SVC.

2.2.2 Overall Encoder Structure

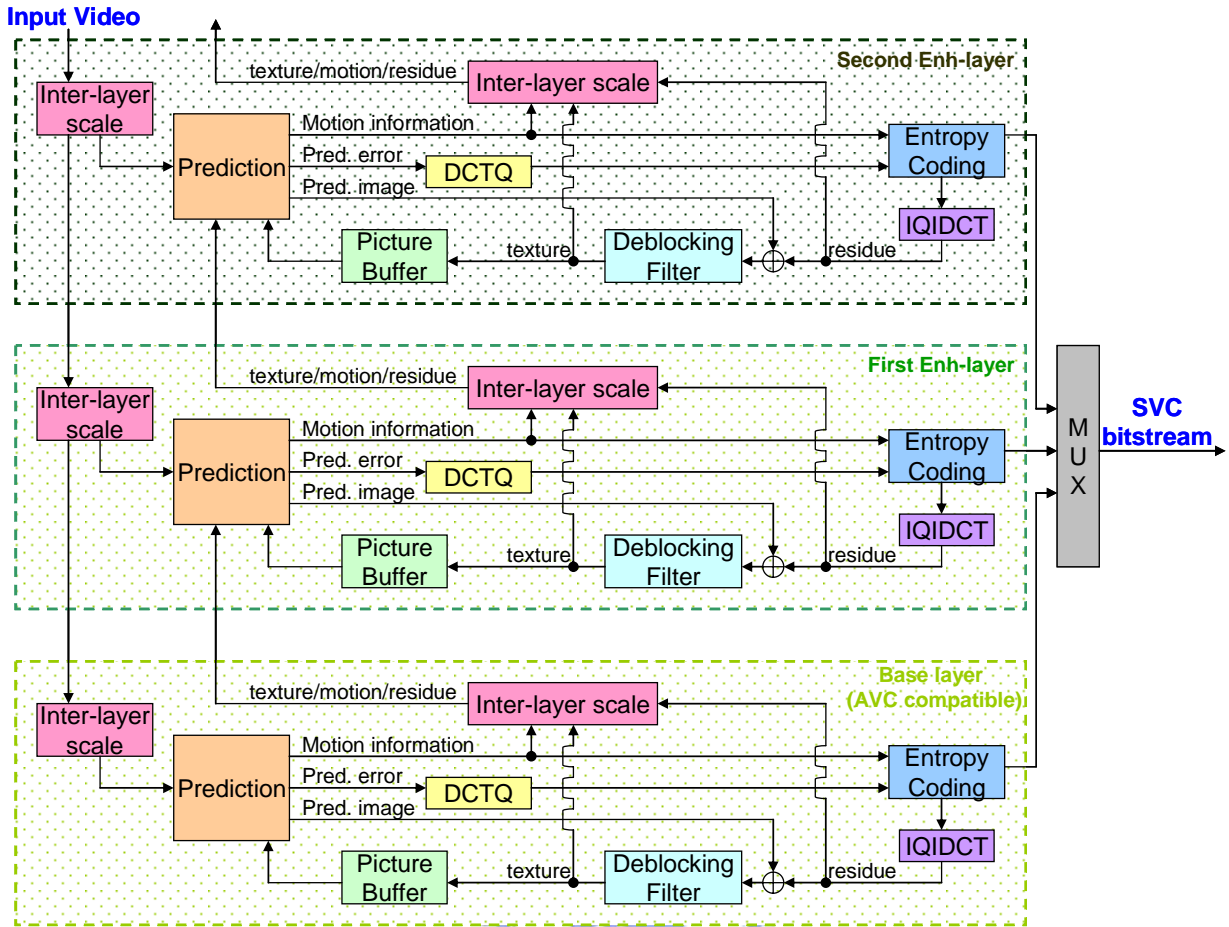
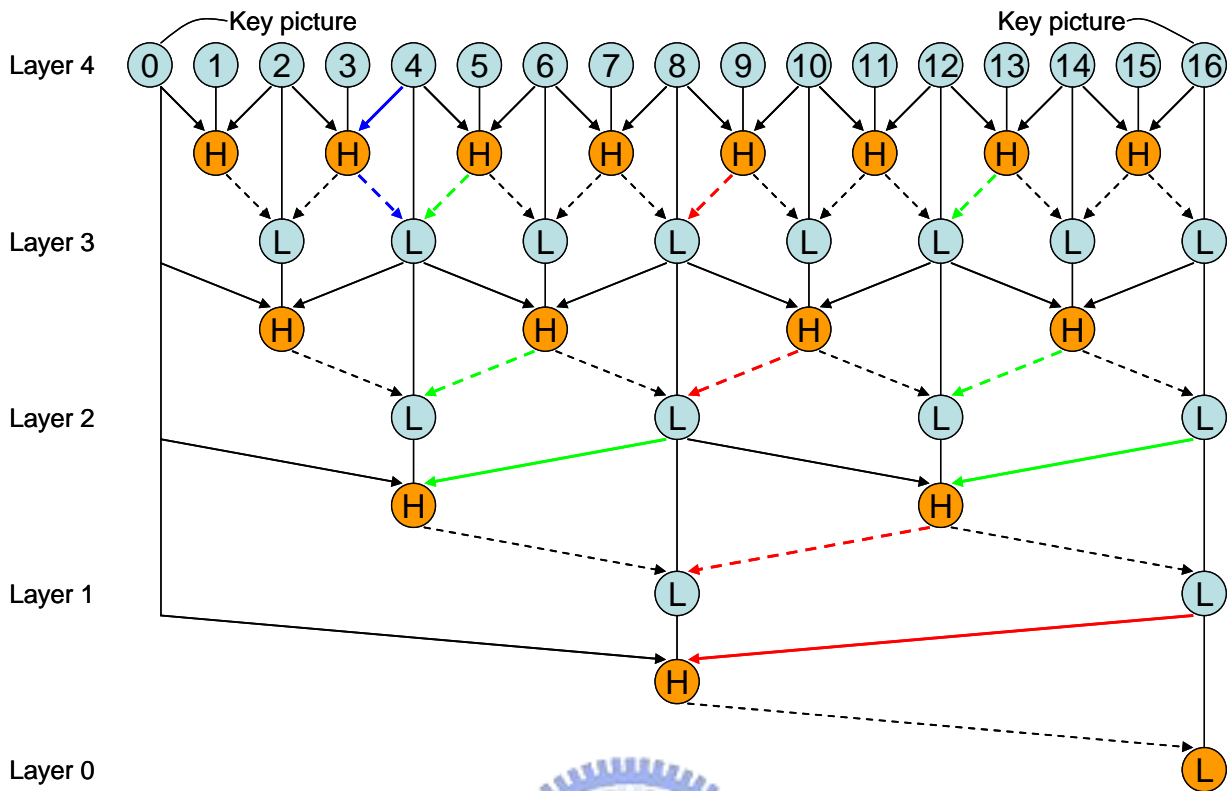


Figure 2.2. H.264/AVC SVC encoder structure with three spatial/SNR layers

In this section, we present an overview of the encoder structure of H.264/AVC SVC. The H.264/AVC SVC encodes the video into multiple spatial, temporal, and SNR layers¹ for combined scalability. Figure 2.2 shows a generic structure of H.264/AVC SVC encoder with three spatial layers (or SNR layers). The input video is spatially decimated to support various spatial resolutions, which is coded with separated encoders as shown in dotted boxes of Figure 2.2.

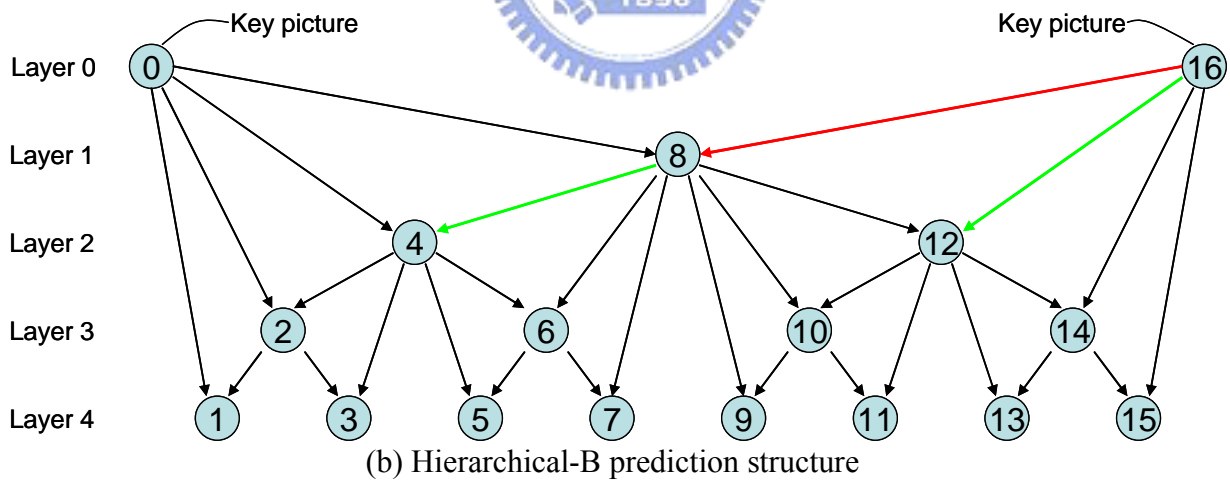
¹ In this chapter, we use “SNR layer” instead of “quality layer” to indicate the layers at the same resolution but with different quality. This is to prevent the ambiguity with the “quality layer” technique that is used for the bit stream adaptation, which will be described in section 2.2.7.2.





 prediction update

(a) MCTF prediction structure



(b) Hierarchical-B prediction structure

Figure 2.3. Temporal decomposition

For each spatial layer, temporal scalability of multiple levels is supported with hierarchical-B structure [4], and motion compensated temporal filtering (MCTF) structure can be used as a pre-processing tool for better coding efficiency. The two

prediction structures are illustrated in Figure 2.3 and more detail will be given in Section 2.2.3.

Since the information of different layers contains correlations, an inter-layer prediction scheme reuses the texture, motion, and prediction information of the lower layers to improve the coding efficiency at the enhancement layer. When each layer has different spatial resolution, the prediction needs to perform interpolation. Note that H.264/AVC SVC also support non-dyadic spatial resolution ratio among spatial layers. After the inter-layer prediction module, the residues of each spatial layer are encoded with either an embedded coder for fine granularity scalability (FGS), or a non-scalable coder for coarse granularity scalability (CGS). However, the entropy coding is restricted to non-scalable mode when it is the first SNR layer of a spatial layer (also refer to as “SNR base layer” in this article). The lower layers do not refer to higher layers for prediction so that the removal of enhancement layers does not affect the decoding of lower layers. In the following Sections, we will describe the detail for temporal, SNR and spatial scalability.

2.2.3 Temporal Scalability

The temporal scalability is implemented with hierarchical B-pictures, while Motion Compensated Temporal Filtering (MCTF) can be used as a pre-processing tool for better coding efficiency.

2.2.3.1 Motion Compensated Temporal Filtering

The MCTF is a temporal decomposition technique that adaptively performs the wavelet decomposition and reconstruction along the motion trajectory using Haar and 5/3 wavelets, which can be implemented with lifting schemes with only one prediction/update step. Particularly, the lifting scheme of 5/3 wavelet is realized by

traditional bi-directional prediction. In Figure 2.3 (a), the layer 4 contains the full resolution and the 5/3 wavelet is used for most predictions. For temporal decomposition, the odd-indexed pictures are predicted from the adjacent even-indexed pictures to produce the high-pass pictures. The even-indexed pictures are updated to generate low-pass pictures using combination of the adjacent high-pass pictures.

When the Haar wavelet is selected, the uni-directional prediction is formed. As illustrated in Figure 2.3 (a), the prediction and update path of Picture 3 shown with blue color are removed. Particularly, uni-directional prediction can be either forward or backward prediction. In addition, the selection of uni-/bi-directional prediction (i.e., the selection of Haar and 5/3 wavelet) is adaptive for each block. To remove the temporal redundancy, motion compensation is conducted before the prediction and update steps.

For temporal scalability of multiple levels, wavelet decomposition is recursively applied on the low-pass pictures of different layers. Using n decomposition stages, up to n levels of temporal scalability can be achieved. The video of lower frame rate consists of the low-pass pictures at lower layer [5].

The MCTF structure requires memory buffer and coding delay equal to the whole GOP size. To reduce the complexity, some backward prediction/update path can be removed. As illustrated in Figure 2.3, removal of the red (and green) prediction/update path reduces the memory requirement and coding delay to half (or quarter) of the GOP size.

2.2.3.2 Hierarchical-B Structure

In MCTF, the un-compressed pictures are employed for prediction leading to an open-loop control. With such control, the encoder provides better prediction since original pictures has higher quality. However, it causes mismatch error between encoder and decoder in the presence of quantization error. Furthermore, the update step doubles

the complexity and increases memory requirement.

To investigate the performance of loop control and justify the complexity increase of the update step, several studies have shown that the closed-loop structure without update step outperforms the open-loop MCTF structure in most of the testing conditions [4]. The update step can be replaced by a simpler noise reduction filter and it can be disabled at decoder side without incurring significant degradation of subjective quality. However, the update step at encoder side does reduce the quality variation of decoded pictures. After these studies, a closed-loop control at encoder side replaces the open-loop control and the update step is now removed from the normative parts. This new temporal decomposition structure is known as “hierarchical-B” or “pyramid-B” prediction structure as shown in Figure 2.3 (b). To support closed-loop encoding, the pictures at lower layers are encoded first such that the pictures at higher layers can refer to the reconstructed pictures at lower layers. Another advantage is that such a prediction scheme is already supported by the syntax of H.264/AVC [1]. To reduce the memory requirement and coding delay, the similar concept used in MCTF can be applied to hierarchical-B structure.

2.2.3.3 Adaptive Reference Fine Granularity Scalability

In the hierarchical-B structure, the key pictures get temporal prediction only from the base layer of the previously coded key pictures but the non-key pictures include both the base and SNR enhancement layers for temporal prediction. Since the base layer has low bit rate and thus poor quality, the key pictures generally have poor prediction efficiency. To improve coding efficiency, the prediction of key pictures should incorporate the SNR enhancement layers. However, drift occurs as the enhancement layer may be truncated. The same problem also exists in the non-key pictures but the

hierarchical-B structure significantly constrains the length of the prediction path and propagation of drift. The drift problem of key pictures was also extensively discussed during the development of MPEG-4 FGS [6]. In MPEG-4 FGS, the enhancement layer is only predicted from the base layer with poor quality, leading to poor coding efficiency. Several works employ the enhancement layer for prediction with various drift control mechanism [7][8]. In particular, RFGS [8] uses leaky prediction to improve coding efficiency while constraining drifting errors. The predict data from the enhancement layer is multiplied with a leaky factor, which is smaller than one, in each prediction loop. When the predicted data from the enhancement layer are truncated, the drift is decayed by the leaky factor in each prediction loop leading to 3 to 4 dB improvement [8]. The stack robust FGS (SRFGS) further incorporates multiple prediction loops to improve R-D performance over a wide range of bit rates [9].

In H.264/AVC SVC, the adaptive reference FGS (ARFGS) approach adaptively selects the leaky factor at transform coefficient level for improving the coding efficiency of key pictures. The ARFGS prediction process is performed in the transform domain. For each coefficient at the enhancement layer, the ARFGS reference coefficient is constructed from both the co-located coefficient at the reconstructed base layer and the predicted coefficient at the enhancement layer from the previous frame. Depending on whether the co-located residue at the base layer is zero or not, the ARFGS reference coefficient is set equal to a weighted average of the two sources. After generating the ARFGS reference coefficients, they are inverse transformed to spatial domain to obtain the ARFGS reference block. If all the collocated residues in the base layer are zeros, the derivation of ARFGS reference block is simplified to the weighted average of the two sources in the spatial domain, and the transform domain prediction process is skipped. In addition, the multi-loop prediction in SRFGS is also implemented in H.264/AVC

SVC. A single enhancement layer loop decoding method can be used to reduce complexity with some degradation of the coding efficiency improvement of multi-loop prediction.

2.2.4 SNR Scalability

The SNR scalability consists of Coarse Grain Scalability (CGS) and Fine Grain Scalability (FGS). The former encodes the transform coefficients in a non-scalable way while the latter can be truncated at any location.

2.2.4.1 Coarse Grain Scalability (CGS)

The CGS layer data can only be decoded as an integral part. Each CGS layer has its own motion information and temporal prediction. There is inter-layer prediction for CGS to re-uses information from the lower layers but it does not require spatial interpolation as all layers have identical resolution. Further, it does not use motion vector refinement (*quarter-pel refinement mode*) as in spatial scalability.

2.2.4.2 Fine Grain Scalability (FGS)

The FGS layer arranges the transform coefficients as an embedded bit stream which allows truncation at any arbitrary point. The cyclical block coding is proposed to achieve embedded representation. Each FGS layer is coded in two passes: significant and refinement passes. The significant pass first encodes the insignificant coefficients (zeros) in the subordinate FGS layers. Then, the refinement pass refines the significant coefficients with data from -1 to +1. During the significance pass, the transform coefficients are coded in a cyclical, block-interleaved manner. Each coding cycle in a block includes an End-of-Block (EOB) symbol, a Run index (number of consecutive zeros), and a non-zero quantization index. The EOB symbol is coded first to signal

whether there are non-zero coefficients to be coded in a cycle. Then, the Run index represented by several significance bits further locates the non-zero coefficient. In the refinement pass, the significant coefficients are refined in a subband-by-subband fashion. The significant coefficients of low-frequency subbands are refined before those of high-frequency subbands. With block-interleaved coding order in both coding passes, the decoded video can have more uniform quality when the bit stream of FGS layers is truncated. To further reduce the bit rate, each symbol can be coded by CABAC or CAVLC. In both entropy coding modes, the spatial correlations are employed by constructing the context model. For example, for the coding of a significance bit, the significance status of the co-located coefficients in the neighboring blocks is referred.

Besides using different entropy coder, each FGS slice (*progressive refinement slice* or *PR slice*) provides one more flag (*motion_refinement_flag*) to select prediction process. When this flag is set to 0, the motion information will not be refined in a FGS slice. The FGS layer simply re-uses the motion information of the previous SNR layer and successively refines the prediction residue of the previous SNR layer. When the flag is set to 1, it has its own motion and the residue is adaptively predicted from the previous SNR layer. The motion refinement provides more than 1 dB gain, which is more noticeable when the base layer is coded at low bit rate or the FGS layers cover a wide range of bit rates. With motion refinement, FGS also provides similar coding efficiency as the CGS.

2.2.5 Spatial Scalability

Similar to the MPEG-2/4 approach, the spatial scalability is achieved by decomposing the original video into spatial pyramid. As shown in Figure 2.2, each spatial layer is coded independently while the motion and temporal prediction are

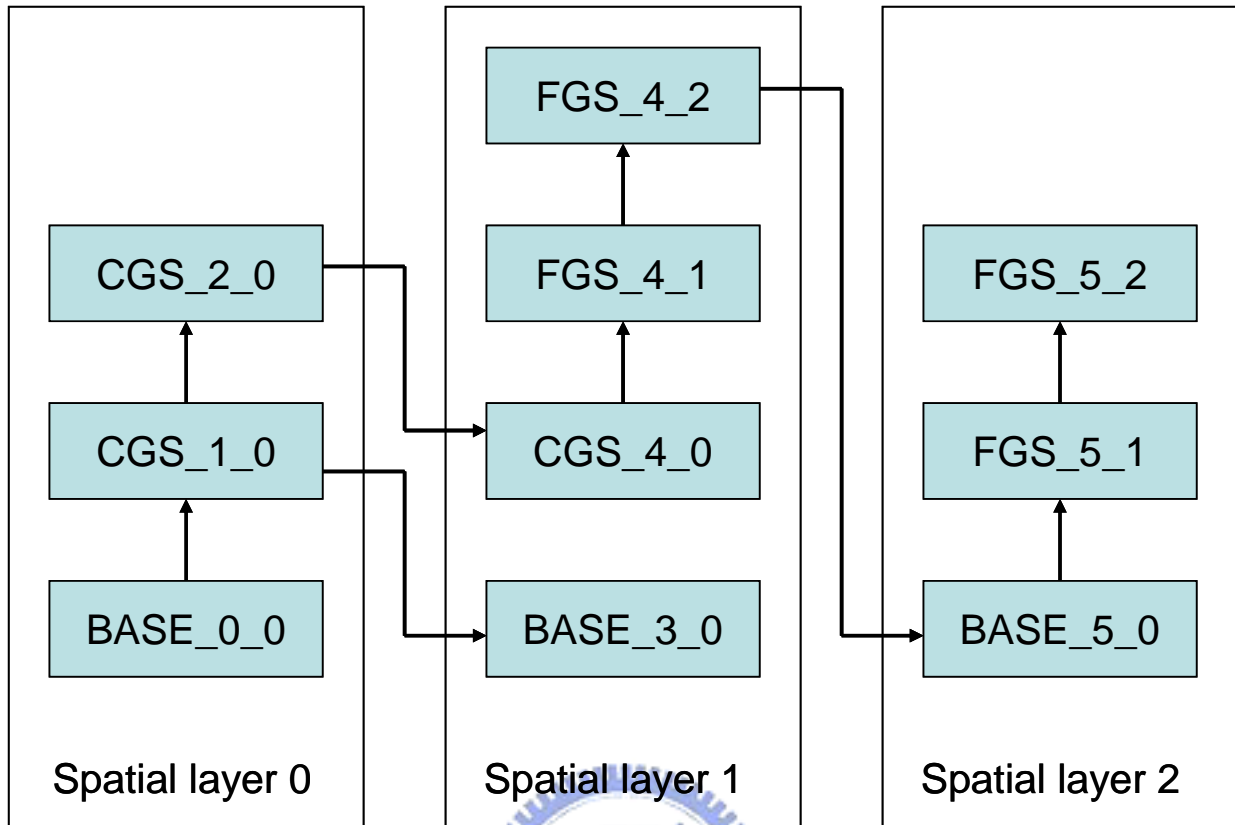


Figure 2.4. Configuration of inter-layer prediction

derived from the reference pictures at the same layer. To remove the redundancy among layers, significant inter-layer prediction is used for motion, residue, and texture.

2.2.5.1 Inter-layer Prediction Structure

The inter-layer prediction is dependent on the types of layers used. The spatial and CGS layers can flexibly select the reference layer from any lower layers while the FGS layer must be predicted from the previous SNR layer at the same resolution.

As shown with an example in Figure 2.4, each rectangle specifies a coding layer of a picture using the notation of X_Y_Z, where the symbol X denotes coding method of the layer including BASE (the SNR base layer), CGS, and FGS. The second symbol Y and third symbol Z specify the *dependency_id* and *quality_level* for a spatial or a SNR layer, where the *dependency_id* is incremented by 1 for the successive spatial layers or

CGS layers and the *quality_level* is incremented by 1 for the successive FGS layers. Both parameters are used by the decoder to identify a coding layer. The BASE_0_0 is the lowest layer that is compatible with H.264/AVC. On top of the BASE_0_0, CGS_1_0 and CGS_2_0 layers are the CGS layers, which are predicted from BASE_0_0 and CGS_1_0, respectively. In the second column, BASE_3_0 is the base layer of the second spatial layer. With flexible selection of the reference layer, BASE_3_0 refers to CGS_1_0 while CGS_4_0 refers to CGS_2_0 instead of BASE_3_0. In this example, CGS_4_0 is decodable even when BASE_3_0 is corrupted by errors. The rule for the FGS layer is different for CGS/spatial layer. The FGS layer can only refer to previous SNR layer of the same resolution. With the configuration shown in Figure 2.4, some layers are redundant for the decoding of certain layer. For instance, the CGS_2_0 is redundant for decoding BASE_3_0. Similarly, BASE_3_0 is redundant for decoding CGS_4_0. Such flexibility is left for further Rate-Distortion performance optimization. The inter-layer prediction information is categorized as intra texture, motion, and residue predictions [3].

2.2.5.2 Intra Texture Prediction

Intra texture prediction uses the reconstructed image of the reference layer to predict an enhancement layer. As the inter-layer prediction of a block refers to an inter-block in the reference layer, or refers to an intra-block in the reference layer that predicted from its neighboring inter-blocks, the motion compensation will be performed at the reference layer to generate the prediction. When multiple spatial layers are coded, such a process may be invoked multiple times leading to significant complexity.

To reduce the complexity, the constrained inter-layer prediction is used to allow only intra texture prediction from an intra-block at the reference layer. Moreover, the referred intra-block can only be predicted from another intra blocks (i.e., the reference

layer re-use of “constrained intra prediction” in H.264/AVC). In this way, the motion compensation is invoked only at the highest layer. Such a constraint is also referred to as “single loop decoding”. However, it should be noted that the key pictures can still be configured as multiple loop decoding while the non-key pictures are restricted to the single loop decoding. Before the prediction, the reconstructed image in the reference layer will be firstly de-blocked and spatially interpolated by the 6-tap half-pixel filter.

2.2.5.3 Motion Prediction

Motion prediction is used to remove the redundancy of motion information, including macroblock partition, reference picture index, and motion vector, among layers. In addition to the macroblock modes available in H.264/AVC, H.264/AVC SVC creates two additional modes for the inter-layer motion prediction. The first mode (*base layer mode*) reuses the motion information of the reference layer without spending extra bits. The second mode (*quarter-pel refinement mode*) refines the motion vector to quarter-pixel precision. The allowable offset of refinement is -1 or 1. If neither one is selected, independent motion is encoded. Note that the motion vectors and macroblock partition of the reference layer may be interpolated before the prediction.

2.2.5.4 Residue prediction

Residue prediction is used to reduce the energy of residues after temporal prediction. A similar idea was proposed in PFGS [7], where the DCT coefficients of the enhancement layer are predicted from those of the base layer. In H.264/AVC SVC, the residue prediction is performed in spatial domain. Due to the inter-layer motion prediction, consecutive spatial layers may have similar motion information. Thus, the residues of consecutive layers may exhibit strong correlations. However, it is also possible that consecutive layers have independent motion and thus residues of two consecutive layers become uncorrelated. Therefore, the residue prediction in

H.264/AVC SVC is done adaptively at macroblock level. Like the motion prediction, the residues at the reference layer are interpolated with a bilinear filter before the prediction. Spatially, each macroblock is interpolated separately and the filtering process cannot cross the macroblock boundary.

2.2.6 Interlaced Coding

While the H.264/AVC SVC has considered progressive video so far, the interlaced coding tools are necessary when applying the scalability among several common video formats. The H.264/AVC SVC needs to consider a scenario where the base layer is coded with progressive mode while the enhancement layer is coded by interlaced format, and vice versa. Thus, an ad-hoc group (AHG) was established to develop interlaced coding tools for H.264/AVC SVC. However, none of the proposals has been adopted so far. In the following, we briefly summary the techniques that have been proposed for interlaced coding.

In the interlaced coding, the main issue for H.264/AVC SVC is the inter-layer prediction since two successive layers may be coded by different modes. Some proposals utilize a “two-steps” approach: one step deals with the inter-layer prediction between different modes (frame or field), but with the same resolution. Another step handles the inter-layer prediction between different resolutions, but with the same mode. The first step is applied on the base layer to generate a “virtual layer” while the second step is applied further on the “virtual layer” to produce the final inter-layer prediction. For example, the inter-layer prediction between a progressive CIF sequence and an interlaced 4CIF sequence is considered. A 4CIF virtual layer is constructed from a progressive CIF and it is followed by the frame to field inter-layer prediction at the same resolution. Due to the possible phase shift of the frame and field between the

successive layers, the re-sampling (down-/up-sampling) process needs some adaptations.

2.2.7 Bit stream Extraction and Adaptation

The H.264/AVC SVC bit stream contains a set of predefined spatio-temporal and quality resolutions. An extractor can be used to extract the bit stream for the prescribed resolution. There are two extraction methods namely simple truncation and quality layers extraction.

2.2.7.1 Simple Truncation

For simple truncation [3], the extractor determines all the reference layers required for decoding the base layer of the requested spatio-temporal resolutions. Because of causality in encoding, the lower layers have higher priority in the extraction process. The higher layer is excluded first if the requested bit rate only allows partial layers to be transmitted. If more bandwidth is available, the SNR layers of the requested spatio-temporal resolutions are then transmitted. If CGS is used for SNR scalability, the bit stream is truncated at the layer boundary. If FGS is used, every picture is equally truncated according to the target bit rate.

2.2.7.2 Quality Layer Adaptation

The concept of quality layer is to add side information in the NAL units that encapsulates FGS layers so as to provide better bit stream adaptation. The *quality layer id* is sent as side information with each NAL unit to signal the importance of each unit. The extractor can drop a packet according to the *quality layer id*, i.e., the packet of least importance will be dropped first.

Bit stream extraction, similar to simple truncation method, keeps the required

reference layers from lower layers to higher layers until the base layer of the requested spatio-temporal resolution is reached. At the requested spatio-temporal resolution, the extractor firstly computes the bit rate of each quality layer and then removes the NAL units according to the *quality layer id*. If the target bit rate can not cover all the NAL units of a quality layer, all the NAL units with this *quality layer id* will be equally truncated. From the simulation results, the concept of quality layer provides up to 0.5dB PSNR improvement versus simple truncation.

2.2.8 Performance Comparison between H.264/AVC and H.264/AVC SVC

In this section, we compare the coding efficiency of H.264/AVC and H.264/AVC SVC. For the simulation, the JM10.1 and the JSVM with the tag JSVM_4_6 are used. In addition, both H.264/AVC and H.264/AVC SVC have the same GOP size, which is 64, and all the key pictures are intra coded. Without any particular statements, the other configurations are the same as those in [4].

The comparison mainly contains three parts: H.264/AVC SVC with spatial scalability only, H.264/AVC SVC with SNR scalability only, and H.264/AVC SVC with combined scalability (, i.e., simultaneously enable spatial, temporal, and SNR scalability). Temporal scalability is not compared separately because it is already supported in H.264/AVC by the hierarchical-B structure. The sequence Crew is used.

2.2.8.1 H.264/AVC SVC with Spatial Scalability Only

In this comparison, the bit stream contains three spatial layers: QCIF, CIF, and 4CIF. The SNR scalability is disabled; thus the RD-points of different bit rates are generated by multiple encoding. Moreover, the input videos of different resolutions are

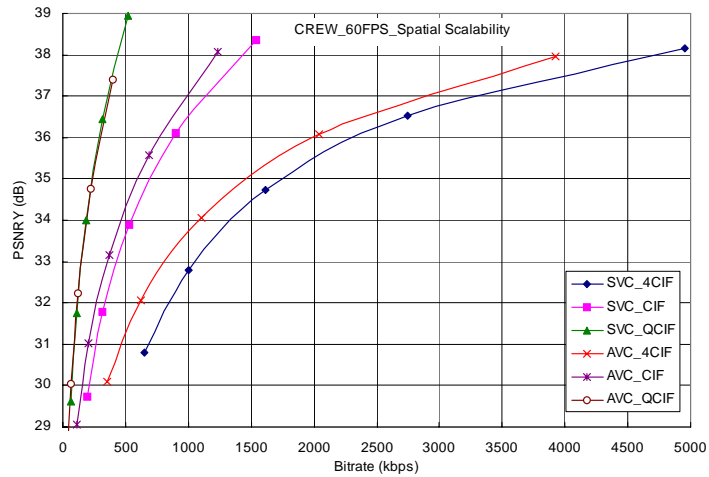
all coded at 60 fps. As shown in Figure 2.5 (a), the QCIF layer, which is H.264/AVC compatible, has identical performance as the H.264/AVC. At CIF layer, there is 0.5dB loss compared with H.264/AVC. At 4CIF layer, the loss is up to 1.0dB at low bit rate and around 0.5dB at high bit rate. As expected, scalability is gained at minor loss of coding efficiency.

2.2.8.2 H.264/AVC SVC with SNR Scalability Only

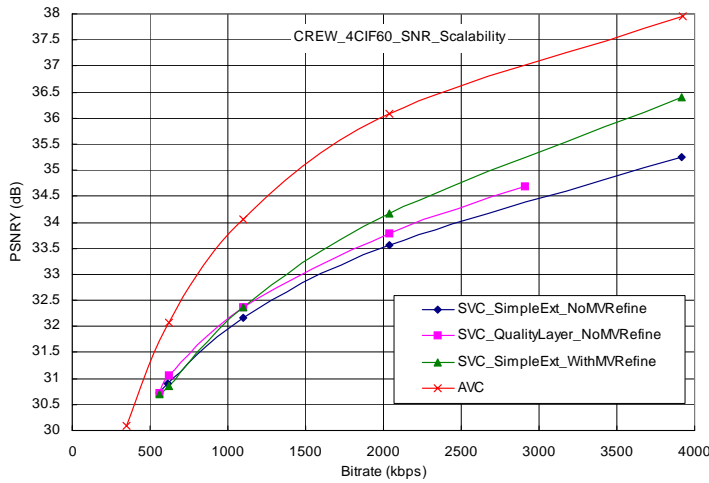
In this comparison, the bit stream supports SNR scalabilities with FGS. Both the simple extraction and the quality layer methods are tested. The performance of motion refinement is also tested. Note that the combination of motion refinement and quality layer is still not supported in the current JSVM software, so the related curve is not shown. The input video of 4CIF is coded at 60 fps. As shown in Figure 2.5 (b), the H.264/AVC SVC with quality layer truncation has 0.5dB improvement compared with the simple extraction. Furthermore, motion refinement offers 1.0dB improvement at high bit rate. However, as compared to H.264/AVC, H.264/AVC SVC still has 1.8dB PSNR loss. The performance degradation can be further reduced by enabling both quality layer and motion refinement in H.264/AVC SVC.

2.2.8.3 H.264/AVC SVC with Combined Scalability

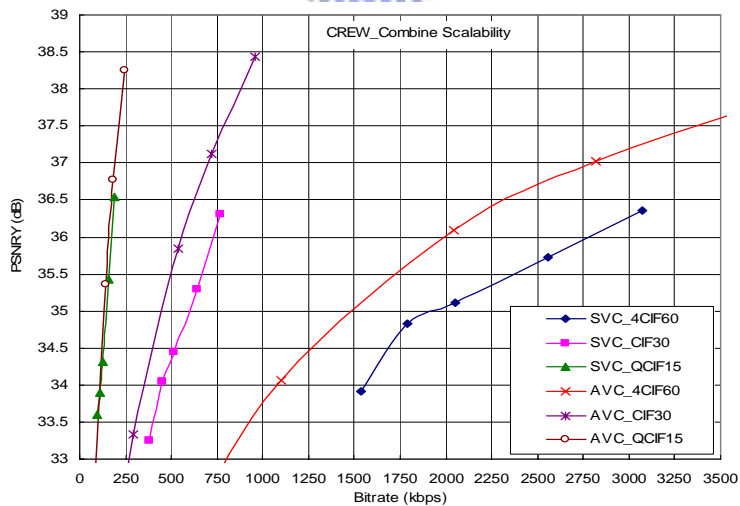
In this comparison, the bit stream supports spatial, temporal, and SNR scalabilities. For the SNR scalability, we use FGS with motion refinement and simple truncation. Both the H.264/AVC and H.264/AVC SVC is encoded with 60fps at 4CIF, 30fps at CIF, and 15fps at QCIF. The GOP size is 64/32/16 for 4CIF/CIF/QCIF, respectively. As shown in Figure 2.5 (c), H.264/AVC SVC has PSNR loss from 0.5dB to 1.2dB as compared to H.264/AVC.



(a) H.264/AVC SVC with spatial scalability only



(b) H.264/AVC SVC with SNR scalability only

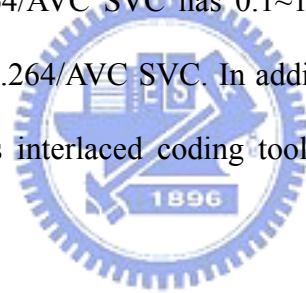


(c) H.264/AVC SVC with combined scalability

Figure 2.5. Performance comparison between H.264/AVC and H.264/AVC SVC

2.2.9 Summary

In this section, we have reviewed the fundamentals of H.264/AVC SVC. As an extension of H.264/AVC, current H.264/AVC SVC algorithm provides a H.264/AVC-compatible base layer and a fully scalable enhancement layer that supports spatial, temporal, and SNR scalability. For spatial scalability, the pyramid structure is used with improved inter-layer prediction. For temporal scalability, the hierarchical-B structure is adopted and may sometimes improve the coding efficiency. For SNR scalability, both CGS and FGS are supported with successive quantization. To assist the bit stream adaptation process, priority information can be embedded in the NAL units. As expected, scalability is gained at the cost of coding efficiency. As compared to H.264/AVC, H.264/AVC SVC has 0.1~1.8dB PSNR loss. Thus, coding efficiency is still an issue for H.264/AVC SVC. In addition, there are many other open problems to be solved such as interlaced coding tools, error resilience/concealment, encoder optimizations, and etc.



CHAPTER 3

Robust Fine Granularity Scalability (RFGS)

3.1 Introduction

The lack of temporal dependency at the FGS enhancement layer decreases the coding efficiency as compared to that of the single layer non-scalable scheme defined in [11]. To improve the MPEG-4 FGS, a motion compensation based FGS technique (MC-FGS) with high quality reference frame was proposed to remove the temporal redundancy for both the base and enhancement layers [12]. The advantage of MC-FGS is that it can achieve high compression efficiency close to that of the non-scalable approach in an error-free transport environment. However, the MC-FGS suffers from the disadvantage of error propagation or drift when part of the enhancement layer is corrupted or lost.

Similarly, the PFGS [7] improves the coding efficiency of FGS and provides means to alleviate the error drift problems simultaneously. To remove the temporal redundancy, the PFGS adopts a separate prediction loop that contains a high quality reference frame where a partial temporal dependency is used to encode the enhancement layer video. Thus, the PFGS trades coding efficiency for certain level of error robustness. In order to address the drift problem, the PFGS keeps a prediction path from the base layer to the highest bitplanes at the enhancement layer across several frames to make sure that the coding schemes can gracefully recover from errors over a

few frames. The PFGS suffers from loss of coding efficiency whenever a lower quality reference frame is used. Such disadvantageous situation occurs when only a limited number of bitplanes are used or a reset of the reference frame is invoked.

To prevent the error propagation due to packet loss in a variable bitrate channels, the leaky prediction technique was used for the interframe loop in DPCM and subband coding systems [13]-[15]. Based on a fraction of the reference frame, the prediction is attenuated by a leak factor of value between zero and unity. The leaky prediction strengthens the error resilience at the cost of coding efficiency since only part of the known information is used to remove the temporal redundancy. For a given picture activity and bit error rate (BER) there exists an optimal leak factor to achieve balance between coding efficiency and error robustness [14]. In this chapter, we propose a flexible FGS framework that allows encoder to select a tradeoff that simultaneously improves the coding efficiency and maintains adequate video quality for varying bandwidth or error prone environments.

The rest of this chapter will be organized as follows. Section 3.2 introduces the basic idea of the Robust FGS (RFGS) framework. In Section 3.3, we show the encoder and decoder structures based on the RFGS scheme. The rate control scheme in the streaming server is explained. The approaches for selecting the optimized parameters are described in Section 3.4. Section 3.5 shows the performance and robustness of the RFGS algorithm based on several typical channel transmission scenarios. Finally, a summary is given in Section 3.6.

3.2 Prediction Techniques of the Enhancement Layer

The MPEG-4 FGS compresses the enhancement layer with only the prediction that

comes from the base layer of the current frame. Therefore, truncation of the enhancement layer does not cause error propagation. While providing flexibility in adapting the bandwidth variations and providing robustness to packet loss and errors, the MPEG-4 FGS is worse in coding efficiency as compared to the traditional two-layer Signal-to-Noise-Ratio (SNR) scalable scheme because the SNR scalable approach uses a high quality reference frame. Such an improved coding efficiency comes with a penalty in error propagation whenever there is a loss at the enhancement layer. The picture quality will drift until the next intra-coded frame [7]. Thus, the MPEG-4 FGS approach offers the best error robustness while the SNR scalable approach provides the best coding efficiency. We will describe a novel and flexible framework, which is referred to as RFGS that aims to strike a balance between these two approaches. The RFGS focuses on constructing a better reference frame based on two motion compensated (MC) prediction techniques: leaky and partial predictions.



3.2.1 Leaky Prediction

The leaky prediction [14] technique scales the reference frame by a factor α , where $0 \leq \alpha \leq 1$, as the prediction for the next frame. The leak factor is used to speed up the decay of error energy in the temporal directions. In RFGS, we use the leak factor to scale a picture that is constructed based on the concept of partial prediction as detailed in the next subsection.

3.2.2 Partial Prediction

As described in **Figure 3.1**, the RFGS is constructed with two prediction loops for the base and enhancement layers. The base layer loop is coded with a non-scalable approach for all frames F_i . The enhancement layer loop uses an improved quality

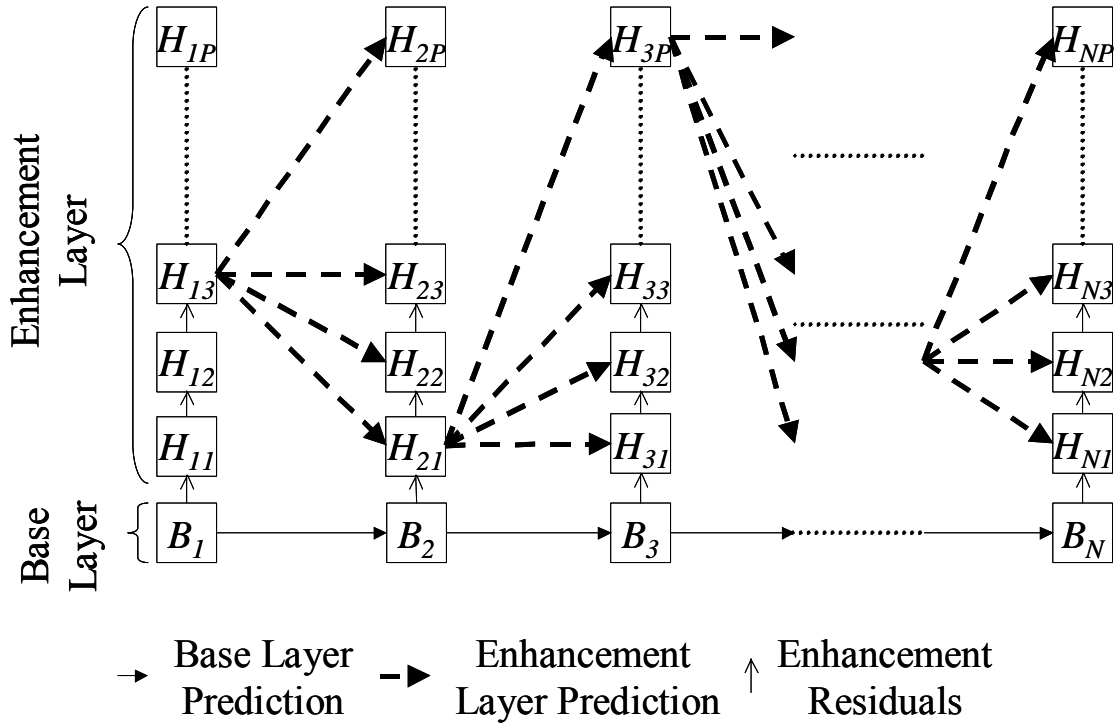


Figure 3.1. Partial inter prediction mode for coding the bitplanes at the enhancement layer using RFGS coding framework. Each frame has the flexibility to select the number of bitplanes used to generate the high quality reference frame. For example, the first frame uses three bitplanes to compute the high quality reference frame.

reference frame that combines the base layer reconstructed image and partial enhancement layer. Thus, the enhancement layer loop can be built with an adaptive selection of number of bitplanes for the reference picture. The combinations of selections for each frame constitute multiple prediction paths.

Let's assume that each frame has P maximal number of bitplanes for the enhancement layer. As the number of bitplanes (denoted as β) used is increased, the residuals will be decreased that translates into improved coding efficiency. On the other hand, the reconstruction errors will accumulate and propagate if the bitplanes used for the reference frame are not available at the decoder. Thus, the parameter β can be used to control the tradeoff between coding efficiency and error robustness.

Combining the concepts of partial and leaky predictions, the first β bitplanes will

be scaled by a leak factor. Consequently, if any information at the first β bitplanes is lost, the error will be attenuated by α times for each frame at the enhancement layer. Since the value of α is smaller than unity, the drift will be eliminated in a few frames. Thus, the RFGS is implemented by defining a set of the parameters for each frame i :

$$\{M_i(\alpha, \beta)\}, \quad i = 0, \dots, (N-1) \quad (3.1)$$

, where the parameter α denotes the leak factor and the parameter β denotes the number of the bitplanes used to construct the reference frame. The symbol N is the total number of frames in the video sequence. As compared to the PFGS [7], the periodic reset of the reference frames can be simulated with a periodic selection of the parameter α as zeros. The MPEG-4 FGS is equivalent to the case of setting α to zero through the whole sequence. As compared to the MC-FGS [12], the use of high quality reference frames can be simulated with α equals to unity for all reference frames. Thus, the RFGS provides a flexible MC prediction scheme that can be adapted to achieve various tradeoffs as proposed by PFGS and MC-FGS [7][12].

3.2.3 Adaptive Mode Selection

We can easily construct a trellis of predictions based on the selected parameters α and β for each frame. The RFGS leaves great flexibility to optimize the selection of (α, β) to achieve adequate performance in terms of coding efficiency and error robustness. The design is constrained by several parameters such as average bitrate, average bit error rate (BER) and desired video quality. For instance, we have a sample traffic pattern that has significant variation in bandwidth and occasional packet loss as illustrated in **Figure 3.2**. If a specific traffic pattern is known beforehand, the optimal set of β should match the instantaneously available bandwidth and the drift is

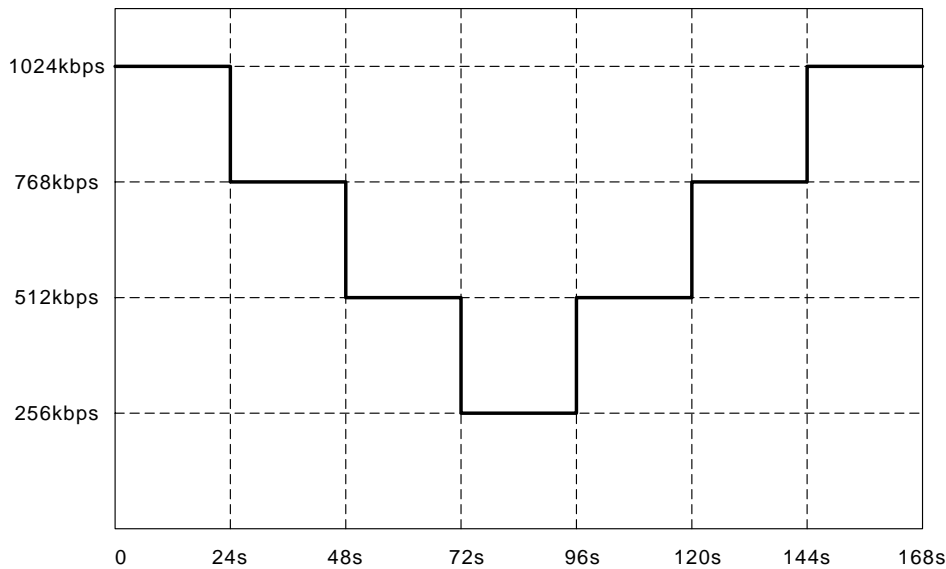


Figure 3.2. Channel bandwidth variation pattern for the dynamic test defined in the MPEG document m8002 [19].

nonexistent. However, it is unrealistic to know this traffic pattern so this solution will not be optimal for other traffic patterns. Thus, the RFGS need to select a set of parameters $\{M_i(\alpha, \beta)\}$, $i = 0, \dots, (N-1)$ that maximizes the average coding efficiency over a range of channel bandwidth.

3.3 The RFGS System Architecture

Based on the concepts of leaky and partial predictions, the RFGS encoder and decoder are constructed as illustrated in **Figure 3.3** and **Figure 3.4** with all the symbols defined in **Table 3.1**. As compared to the MPEG-4 FGS [6], the RFGS has added only a few modules including motion compensation, DCT/IDCT and a reference frame buffer to store the high quality reference frame that is constructed based on the base and enhancement layers. The concept of leaky and partial predictions can be applied to both the base and enhancement layers. We will explain how to realize the leaky prediction at the enhancement layer in detail from section 3.3.1 through 3.3.3. The identical steps can

be applied for the base layer except that the predicted frames of both layers are stored in two distinct frame buffers.



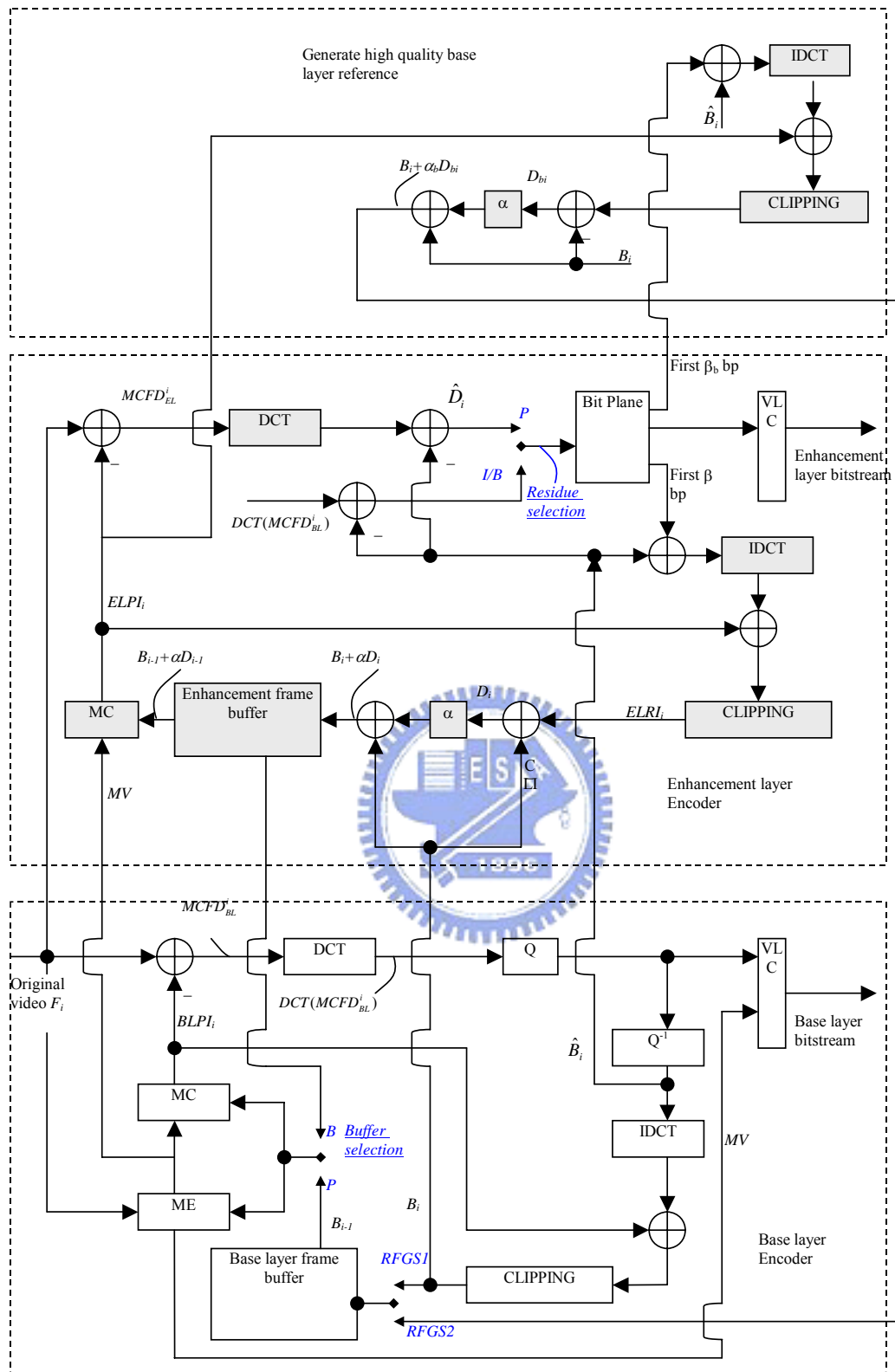


Figure 3.3. Diagram of the RFGS encoder framework. The shadowed blocks are the new modules for RFGS as compared to MPEG-4 baseline FGS.

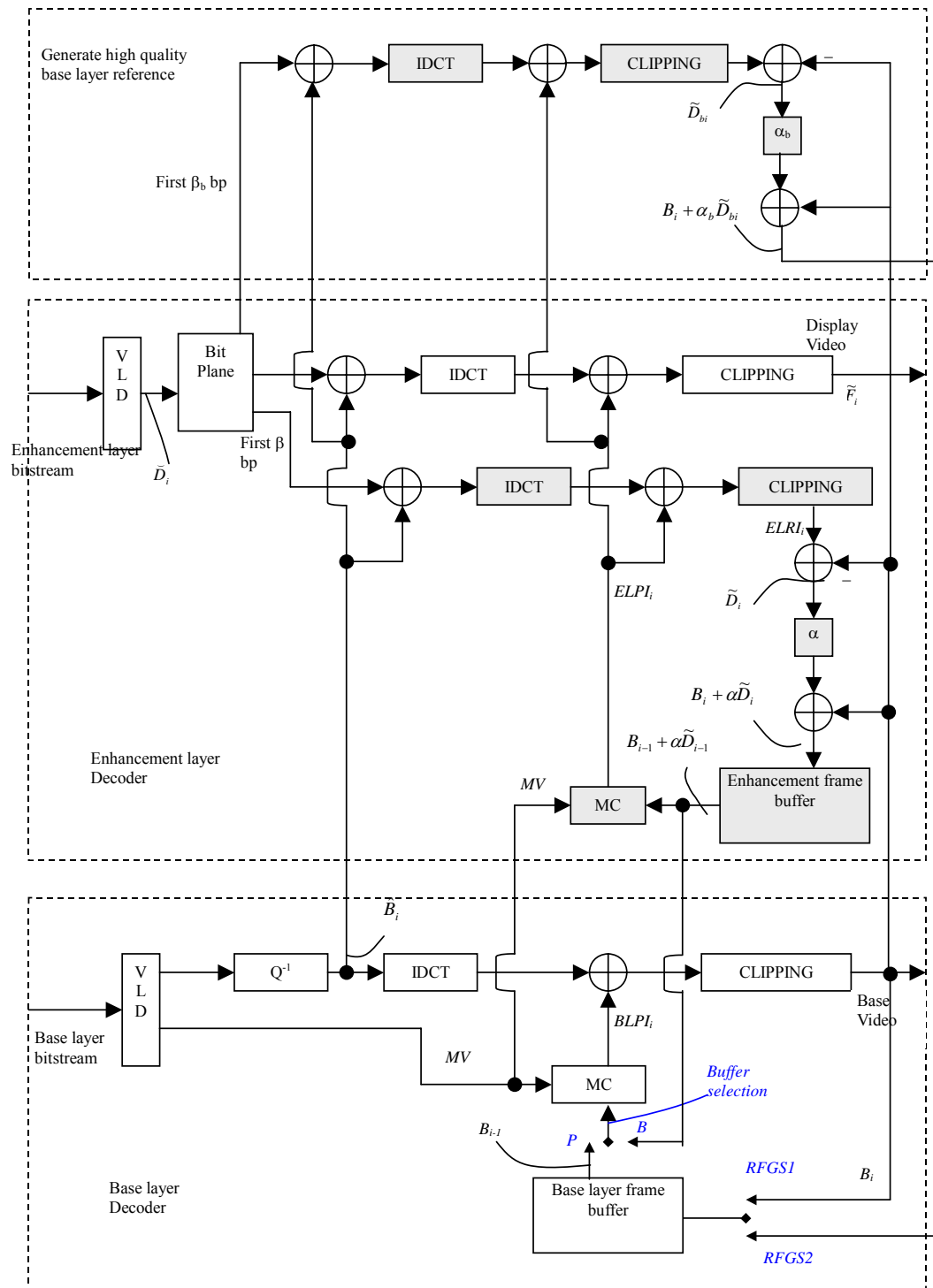


Figure 3.4. Diagram of the RFGS decoder framework. The shadowed blocks are the new modules for RFGS as compared to MPEG-4 baseline FGS.

Table 3.1. Terminology of the RFGS coding framework.

Notation	Definitions
F	The original image
$BLPI$	Predicted base layer frame that is generated by motion compensation from the base layer frame buffer.
$MCFD_{BL}$	Motion compensated frame difference of the base layer, which is the difference between $BLPI$ and the original image.
\hat{B}	Coded DCT coefficients of frame $MCFD_{BL}$. The \hat{B} before de-quantization will be compressed as the base layer bitstream.
B	The base layer reconstructed image, which is the summation of $BLPI$ and \hat{B} . B will be stored in the base layer frame buffer.
$ELPI$	Predicted frame of the enhancement layer that is generated by motion compensation from the enhancement layer frame buffer.
$MCFD_{EL}$	Motion compensated frame difference of the enhancement layer which the difference between $ELPI$ and the original image.
\hat{D}	Difference signal between $MCFD_{EL}$ and \hat{B} for P -pictures or $MCFD_{BL}$ and \hat{B} for I -pictures and B -pictures. \hat{D} will be compressed as the enhancement layer bitstream.
D	The final residual used at the enhancement layer prediction loop in the encoder. $(B + \alpha D)$ will be stored at the enhancement layer frame buffer of the encoder.
\tilde{D}	The received \hat{D} in the decoder side. Since there may be truncation or error during the transmission of enhancement layer bitstream, \hat{D} and \tilde{D} may be different.
$\Delta\hat{D}$	The difference between \hat{D} and \tilde{D} .
\tilde{D}	The reconstructed D in the decoder side. $(B + \alpha\tilde{D})$ will be stored at the enhancement layer frame buffer of the decoder.

3.3.1 Functional Description

The base layer is encoded with the advanced simple profile (ASP) using a modification of the B -pictures. The B -picture is encoded with a high quality reference frame at the enhancement layer. There is no drift because B -picture is not used for prediction. The enhancement layer is encoded with the MPEG-4 FGS syntax but with the new prediction schemes. The enhancement layer uses the same motion vectors from the base layer. The motion compensation module uses the base layer motion vectors and the high quality reference frames to generate the high quality predictions $ELPI$ as shown in **Figure 3.3**. The difference signal $MCFD_{EL}$ for the enhancement layer is obtained by subtracting $ELPI$ from the original signal F . For the P -pictures, the signal \hat{D} is computed by subtracting \hat{B} from the enhancement layer difference signal $MCFD_{EL}$. As for the I -pictures and B -pictures, the signal \hat{D} is computed by subtracting \hat{B} from the base layer difference signal $MCFD_{BL}$. Finally, the signal \hat{D} is encoded with the MPEG-4 FGS syntax to generate the enhancement layer bitstream.

3.3.2 Leaky and Partial Prediction

Now we will describe the technique to generate the high quality reference image using the leaky and partial predictions. The first β bitplanes of the difference signal \hat{D} are combined with the reconstructed base layer DCT coefficients \hat{B} . The resultant signal is transformed back to the spatial domain using IDCT and is added to the enhancement layer motion compensated prediction $ELPI$. The difference between the high quality reference frame and the base layer reconstructed signal B is computed and attenuated by a leak factor α . The base layer reconstructed signal B is added back before storing back into the frame buffer.

The encoding of *B*-pictures as shown in **Figure 3.3** uses the high quality reference frame as the extended base layer to form the prediction for both the base and enhancement layers. The base layer difference signal $MCFD_{BL}$ is first quantized to form the *B*-picture base layer, and the residual (quantization error) is coded as FGS enhancement layer using MPEG-4 FGS syntax. Since *B*-picture is not used as reference frame, there is no drift. Thus, we can increase the leak factor to achieve better coding efficiency. However, the inclusion of *B*-pictures at the enhancement layer requires an extra frame buffer to achieve the extra coding gain.

Since the difference between the high quality reconstructed signal and the low quality reconstructed signal is attenuated by a leak factor α , the attenuated difference and the low quality reconstructed signals will be summed together to form the high quality reference image for the next frame. Therefore, the drift or the difference between the encoder and decoder will be attenuated accordingly. If the leak factor is set to zero, the drift will be removed completely, which is exactly how the MPEG-4 FGS works.

The rationale for performing such a complicated and tricky attenuation process in the spatial domain is because in this way the errors can be recursively attenuated for all the past frames. If the attenuation process is only applied for the first few bitplanes of the current VOP, only the errors occurred in the current VOP are attenuated. The errors that occurred earlier are only attenuated once and can still be propagated to the subsequent frames without further attenuation. In our approach, not only the errors occurred in the current VOP are attenuated but also all the errors in the earlier frames are attenuated. After several iterations, the errors will be reduced to zero.

3.3.3 Analysis of Error Propagation

The RFGS framework is constructed based on the well-known concept of leaky prediction to improve the error recovery capability as proposed in several other video coding techniques such as the DPCM and the subband video coding in [13]-[15]. The major distinction in our approach is the technique to compute the reference frame and the final residual for transmission. In the RFGS framework, the high quality reference image (HQRI) consists of three components including the motion compensated base layer reconstructed frame, the quantized difference signal of the base layer and the attenuated final residual at the enhancement layer. Thus, we have the following relationship:

$$HQRI = B + \alpha \times D$$

, where B is the base layer reconstructed signal and D is the final residual used at the enhancement layer.

We now compute the reconstruction errors when only partial bitstream is available. As illustrated in **Figure 3.3**, we describe the technique to form the base and enhancement layers. For the current frame, the original frame at time i is denoted as F_i . At the base layer, the reconstructed frame of the previous time $i-1$ is denoted as B_{i-1} . The base layer motion compensated frame difference signal is denoted as $MCFD_{BL}^i$ at time i . Thus, the original frame at time i can be computed as

$$F_i = (B_{i-1})_{mc} + MCFD_{BL}^i \quad (3.2)$$

The subscript mc means that the $(B_{i-1})_{mc}$ is the motion compensated version of B_{i-1} . That is, the $(B_{i-1})_{mc}$ equals to the $BLPI_i$ as illustrated in **Figure 3.3**.

$$BLPI_i = (B_{i-1})_{mc} \quad (3.3)$$

The coded version of the based layer difference signal $MCFD_{BL}^i$ is denoted as frame \hat{B}_i . Let the quantization error after encoding be Q_i , the relationship

between $MCFD_{BL}^i$, \hat{B}_i , and Q_i is

$$MCFD_{BL}^i = \hat{B}_i + Q_i. \quad (3.4)$$

The quantized version of the difference signal $MCFD_{BL}^i$, which equals to the signal \hat{B}_i before de-quantization, is compressed as the base layer bitstream. In the MPEG-4 FGS coding scheme, the quantization error Q_i will be encoded to generate the enhancement layer bitstream.

For the enhancement layer, the base layer reconstructed frame B_{i-1} of the previous time $i-1$ and αD_{i-1} will be summed to create the high quality reference frame, where D_{i-1} is the actually used information from the enhancement layer of the previous frame at time $i-1$. After motion compensation, the $MCFD_{EL}^i$ is computed from

$$F_i = (B_{i-1} + \alpha D_{i-1})_{mc} + MCFD_{EL}^i. \quad (3.5)$$

, where the $(B_{i-1} + \alpha D_{i-1})_{mc}$ is the same as the $ELPI_i$ in **Figure 3.3**. That is,

$$ELPI_i = (B_{i-1} + \alpha D_{i-1})_{mc} \quad (3.6)$$

Assume that there is redundancy between $MCFD_{EL}^i$ and \hat{B}_i (the coded version of $MCFD_{BL}^i$), the frame \hat{B}_i is subtracted from the difference signal $MCFD_{EL}^i$ to remove such redundancy. The resultant difference is denoted as \hat{D}_i , which will be compressed for transmission at the enhancement layer. Thus, we have

$$\hat{D}_i = MCFD_{EL}^i - \hat{B}_i. \quad (3.7)$$

Substitute (3.7) into (3.5), the original image F_i can be reformulated as

$$F_i = (B_{i-1} + \alpha D_{i-1})_{mc} + \hat{B}_i + \hat{D}_i. \quad (3.8)$$

By grouping the base and enhancement layer information, (3.7) becomes

$$F_i = (B_{i-1})_{mc} + \hat{B}_i + (\alpha D_{i-1})_{mc} + \hat{D}_i \quad (3.9)$$

$$= B_i + D_i \quad (3.10)$$

,where

$$B_i = (B_{i-1})_{mc} + \hat{B}_i, \quad (3.11)$$

and

$$D_i = (\alpha D_{i-1})_{mc} + \hat{D}_i. \quad (3.12)$$

The signals B_i and D_i will be used for the prediction of next frame. It should be noted that for simplicity, we assume all of the bitplanes in \hat{D}_i are used at the enhancement layer prediction loop.

By expanding the recursive formula of D_i in (3.12), we can get

$$\begin{aligned} D_i &= (\alpha((\alpha D_{i-2})_{mc} + \hat{D}_{i-1}))_{mc} + \hat{D}_i \\ &= (\alpha((\alpha((\alpha D_{i-3})_{mc} + \hat{D}_{i-2}))_{mc} + \hat{D}_{i-1}))_{mc} + \hat{D}_i \\ &= \dots \end{aligned} \quad (3.13)$$

As demonstrated in (3.13), it is obvious that the any errors in final residual D_i will be attenuated in the RFGS framework. Assume there is a network truncation or error at the enhancement layer for frame F_{i-2} , we denote the received enhancement layer bitstream as \tilde{D}_{i-2} and the transmission error is denoted as $\Delta\hat{D}_{i-2}$. Thus, we have

$$\hat{D}_{i-2} = \tilde{D}_{i-2} + \Delta\hat{D}_{i-2}. \quad (3.14)$$

and the reconstructed version of D_{i-2} is denoted as \tilde{D}_{i-2} . Thus,

$$\begin{aligned} \tilde{D}_{i-2} &= (\alpha D_{i-3})_{mc} + \tilde{D}_{i-2} \\ &= (\alpha D_{i-3})_{mc} + \hat{D}_{i-2} - \Delta\hat{D}_{i-2}. \end{aligned} \quad (3.15)$$

Comparing (3.12) and (3.15), the difference between D_{i-2} and \tilde{D}_{i-2} is $\Delta\hat{D}_{i-2}$.

Now we trace back to the frame F_{i-1} . For simplicity, we assume that there is no error or bit truncation at the enhancement layer for frames F_{i-1} and F_i . Expanding (3.15), we have

$$\begin{aligned} \tilde{D}_{i-1} &= (\alpha\tilde{D}_{i-2})_{mc} + \hat{D}_{i-1} \\ &= (\alpha((\alpha D_{i-3})_{mc} + \hat{D}_{i-2} - \Delta\hat{D}_{i-2}))_{mc} + \hat{D}_{i-1} \end{aligned} \quad (3.16).$$

The difference between D_{i-1} and \tilde{D}_{i-1} is now $\alpha(\Delta\hat{D}_{i-2})$.

Now we move on to the frame F_i and get

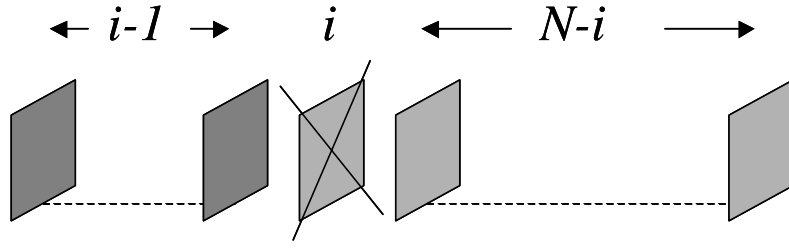


Figure 3.5. Illustration of a transmission scenario with corrupted or lost frame for a video stream of N frames, where the enhancement layer of the i -th frame is assumed to be lost.

$$\begin{aligned}\tilde{D}_i &= (\alpha \tilde{D}_{i-1})_{mc} + \hat{D}_i \\ &= (\alpha((\alpha((\alpha D_{i-3})_{mc} + \hat{D}_{i-2} - \Delta \hat{D}_{i-2}))_{mc} + \hat{D}_{i-1}))_{mc} + \hat{D}_i.\end{aligned}\quad (3.17)$$

The difference between D_i and \tilde{D}_i is now $\alpha^2(\Delta \hat{D}_{i-2})$.

From the above derivations, it is obvious that the errors occurred in the decoded bitstream at the enhancement layer will be attenuated by a factor of α for each iteration. After several iterations, the error will be attenuated to zero for α less than unity. Thus, the drift is removed from the system.

As an example shown in **Figure 3.5**, there is a video bitstream for N frames. Let's assume that only the i -th frame F_i is lost during transmission, the mean square error for the reconstructed enhancement layer frame of size $H \times M$ can be computed as

$$e_i^2 = \frac{1}{HM} \sum_{x=1}^H \sum_{y=1}^M (\hat{F}_i(x, y) - \hat{F}_i^e(x, y))^2 \quad (3.18)$$

, where the signal $\hat{F}_i(x, y)$ represents the reconstructed frame with all bitplanes and the $\hat{F}_i^e(x, y)$ represents the reconstructed frame where some bitplanes are lost. Consequently, the average video quality degradation of the reconstructed picture that is caused by the errors at frame F_i is

$$\Delta MSE_{\text{avg}} = \frac{(1 + \alpha^2 + \dots + \alpha^{2(N-i)})}{N} e_i^2 = \frac{1 - (\alpha^2)^{N-i+1}}{(1 - \alpha^2)N} e_i^2. \quad (3.19)$$

As α tends to unity, the average MSE accumulated through the prediction loop will accumulate as expected. For the leak factor less than unity, the degradation will be decreased exponentially as shown in **Figure 3.15**. The error attenuation can be approximated with an exponential function:

$$\Delta PSNR(\alpha) = K_1(\alpha)e^{-K_2(\alpha)t} = K_1(\alpha)e^{-\frac{t}{\tau(\alpha)}}, \quad (3.20)$$

where $K_1(\alpha)$ and $K_2(\alpha)$ are constants that vary as a function of α and can be computed using the least square approximation technique. The constant $K_2(\alpha)$ is a reciprocal of the time constant $\tau(\alpha)$ for an exponential function. It is expected that $K_2(\alpha)$ is increased as α is decreased because the errors are attenuated faster when α is decreased. As demonstrated in **Figure 3.17**, the time constant $\tau(\alpha)$ is reduced by half when the leak factor α is reduced to 0.9. Thus, the selection of the leak factor α is a critical issue to achieve a better balance between coding efficiency and error robustness. For α that is closed to unity, the coding efficiency is the best while the error robustness is the worst with longest attenuation time constant. On the other hand, for α that is close to zero, the error recovery property will be enhanced at the cost of less coding efficiency.

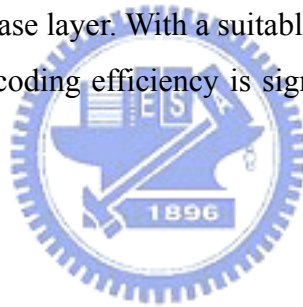
3.3.4 High Quality Reference in Base Layer

As mentioned in Section 3.3.1, the signal \hat{D} , which is transmitted at the enhancement layer, is computed by subtracting \hat{B} from the enhancement layer difference signal $MCFD_{EL}$. Such a differencing reduces the energy of the residuals but increases the dynamic range of the signal \hat{D} , which is particularly inefficient for bitplane coding [16]. Thus, there is room for further improvement. Additionally, there is redundancy that exists between the high quality reference image for the enhancement layer and the base layer difference signal $MCFD_{BL}$. To decrease the fluctuation of \hat{D} and remove the said redundancy, a higher quality reference image for the base layer is used. As compare to the signal B , the statistic characteristics of the higher quality reference for the base layer is closer to that of the high quality reference image for the enhancement layer. Therefore the dynamic range of \hat{D} is reduced and the temporal

redundancy between the high quality reference image for the enhancement layer and the signal $MCFD_{BL}$ is also reduced.

In **Figure 3.3** and **Figure 3.4**, we illustrate how the high quality reference is generated for the base layer. Part of the enhancement layer is duplicated in the part “generate high quality base layer reference” to form the high quality reference image for the base layer. The derivation of the high quality reference image for the base layer is identical to that for the enhancement layer except that the base layer has its own RFGS parameters, which are denoted as α_b and β_b , respectively. The resultant high quality reference image will replace the signal B and is stored in the base layer frame buffer.

Although the use of a high quality reference image for the base layer can achieve a better coding efficiency, it suffers from drift problem at low bitrate [12]. The drift at the base layer cannot be removed because the base layer reference image is not attenuated by α . To strike a balance between the coding efficiency and the error drift, a small α should be used for the base layer. With a suitable selection of α_b , the drift at low bitrate can be reduced and the coding efficiency is significantly enhanced for medium and high bitrates.



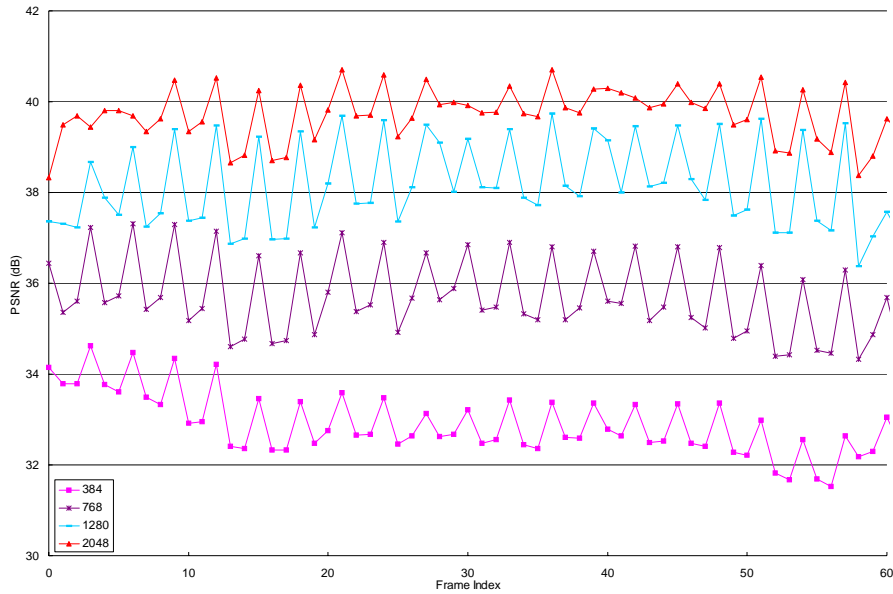


Figure 3.6 The visual qualities of the reconstructed pictures using the proposed RFGS rate control scheme. We provide the quality of the first 60 frames of the Foreman bitstream. The base layer bitstream is encoded with a bitrate of 256kbps. The enhancement layer bitstream is truncated at several bitrates to understand the variation in PSNR for various channel bandwidths. The results show that the PSNR variation is smaller than 2 dB at various bitrate.



3.3.5 Rate Control for the Enhancement Layer

For the MPEG-4 FGS, the rate control is not an issue since there is no temporal dependency among frames at the enhancement layer. However, the rate control is relevant in the case of the RFGS, especially when the expected range of bandwidth in operation is widely varied. The server can adaptively determine the number of bits to be sent frame by frame. When the expected channel bandwidth is small, the bitplanes that are used to construct the high quality reference frame may not be available mostly. Since only the *I*-picture and *P*-pictures are used as the reference frames, the limited bandwidth should be allocated to those anchor frames at low bitrate [12]. The *B*-pictures will also be improved because better anchor frames are used for interpolation. When the average bitrate becomes higher, additional bits should be allocated to *B*-pictures, where

bits can be spent on the most significant bitplanes for more improvements. By allocating more bits to the P -pictures the overall coding efficiency is improved but the PSNR values vary significantly between the adjacent P -picture and B -picture, especially at medium bitrate, where most of the bitplanes in P -pictures have been transmitted but only a few bitplanes for B -pictures are transmitted. The maximal PSNR difference may be up to 4 dB in our simulation. To achieve better visual quality, as shown in **Figure 3.6**, the proposed rate control scheme reduces the variance of the PSNR values of the adjacent pictures at the cost of decreasing the overall quality by about 0.5 dB in PSNR. Since the RFGS scheme provides an embedded and fully scalable bitstream, the proposed rate control can occur at server, router, and decoder. In this chapter, we perform the rate control at the server side for all simulations.

3.4 The Selection of the RFGS parameters

3.4.1 Selection of the Leaky Factor

In order to find an algorithm that computes the optimized α , we perform a near optimal exhaustive search for the parameters by dividing every sequence into several segments that contain a Group of Video Object Planes (GOV). In our simulation, each GOV has 60 frames. The “near optimal” scenario is defined based on the proposed criterion of the “average weighted difference” (AWD), which is the weighted sum of the PSNR differences between the RFGS and the single layer approaches for a given bitrate range. Thus,

$$AWD = \sum_{BR} W(BR) \times D(BR) \quad (3.21)$$

, where BR is a set of evenly spaced bitrates for a given bitrate range. The symbol $W(BR)$ is the weighting function for the bitrate set BR . $D(BR)$ is a set of the PSNR differences between the RFGS and single layer approaches for every bitrate from the set BR . In our

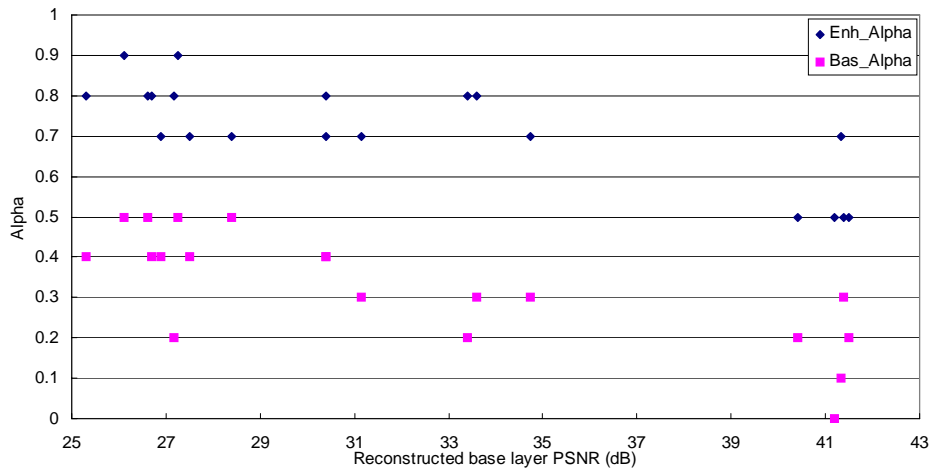


Figure 3.7. The linear dependency between near-optimal leak factor and the picture quality in PSNR of the base layer. The frames within five GOVs, where each has 60 frames, are used for the simulations with the four sequences, namely Akiyo, Carphone, Foreman, and Coastguard.

simulations, the set BR is defined by

$$BR = \{256,512,768,1024,1280,1536,1792,2048,2304\} \text{ kbps},$$

and the weighting function is

$$W(\cdot) = \{2,2,2,2,1,1,1,1\}$$

, where the importance of the PSNR differences at low bitrate is stressed.

To observe the influence of the leak factors on the coding efficiency, the bitplane numbers for both layers are fixed at three bitplanes. The parameters α_e and α_b are scanned from 0.0 to 0.9 with a step size of 0.1. All the combinations of α_e and α_b are employed for each GOV within the sequence and the pair of α_e and α_b with minimal AWD is selected. Thus, we can get a near-optimal combination of α_e and α_b for each GOV. The results would be optimal if we adapt α_e and α_b at frame level but the complexity is prohibitive.

In **Figure 3.7**, we show the relationship between the near-optimal combinations of α_e and α_b and the base layer PSNR values with the experimental results using four

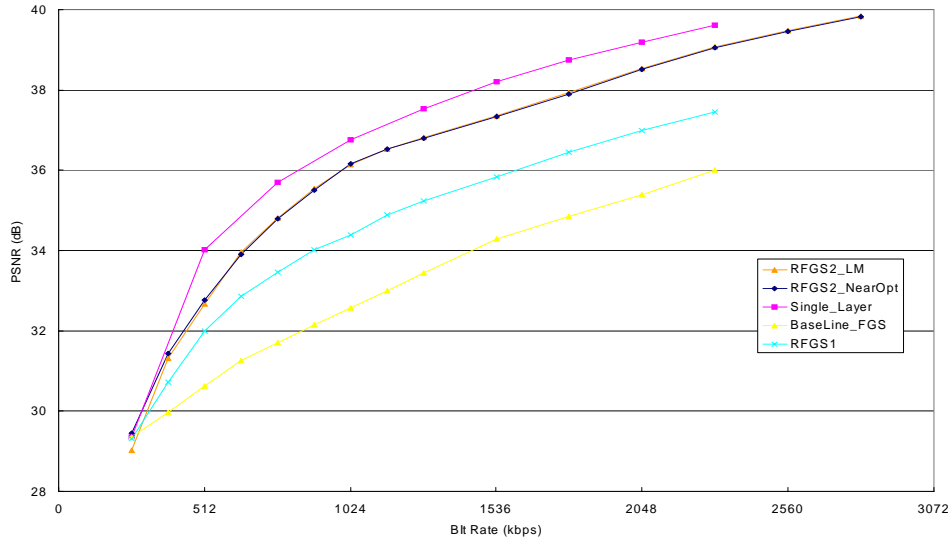


Figure 3.8. PSNR versus bitrate comparison between FGS, RFGS and single layer coding schemes for the Y component of the Foreman sequence, where β is 3. We use three different coding schemes including ‘RFGS1’, ‘RFGS2_NearOpt’, and ‘RFGS2_LM’ in the experiments. ‘RFGS1’ use the RFGS algorithm for the enhancement layer only. ‘RFGS2’ uses the RFGS algorithm for both the enhancement and base layers. ‘NearOpt’ means the result of the near-optimal approach and ‘LM’ means the results using the proposed linear model.

sequences based on the GOV-based scheme. As the PSNR value of the base layer reconstructed frame is decreased, the near optimal α tends to be increased accordingly. Their relationship is almost linear if we eliminate several outliers, which provides a linear model for computing the near optimal α based on the PSNR value of the base layer. For each frame, we first get the base layer PSNR values after encoding. Based on the derived PSNR value per frame and the proposed linear model, we compute both α_e and α_b and encode every frame at the enhancement layer. From **Figure 3.10** to **Figure 3.9** we find that the RFGS using the linear model has almost identical PSNR values as the RFGS based on the near optimal exhaustive search, which has at maximum 0.2 dB differences. The performance of the RFGS based on the proposed linear model is much superior to the RFGS with fixed α_e and α_b found empirically.

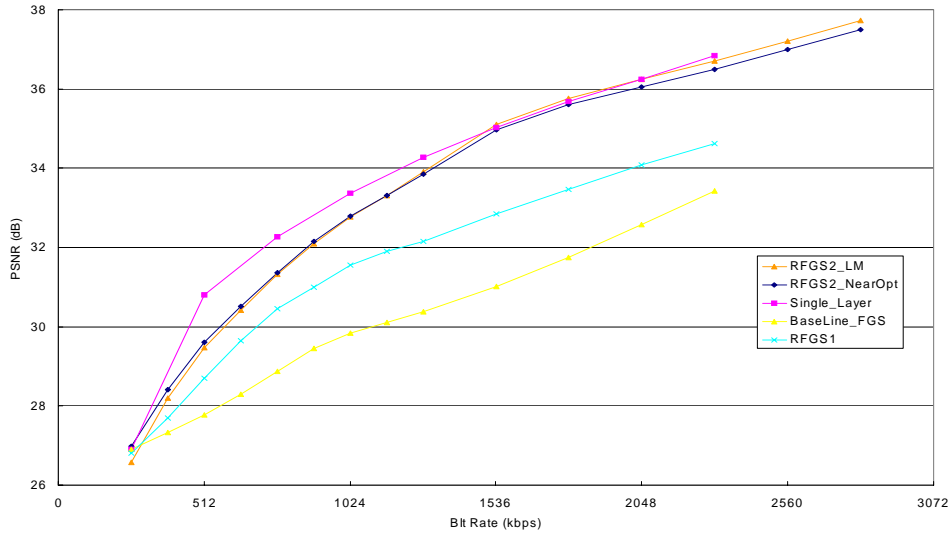


Figure 3.9 PSNR versus bitrate comparison between FGS, RFGS and single layer coding schemes for the Y component of the Coastguard sequence, where β is 3. We use three different coding schemes including ‘RFGS1’, ‘RFGS2_NearOpt’, and ‘RFGS2_LM’ in the experiments. ‘RFGS1’ use the RFGS algorithm for the enhancement layer only. ‘RFGS2’ uses the RFGS algorithm for both the enhancement and base layers. ‘NearOpt’ means the result of the near-optimal approach and ‘LM’ means the results using the proposed linear model.

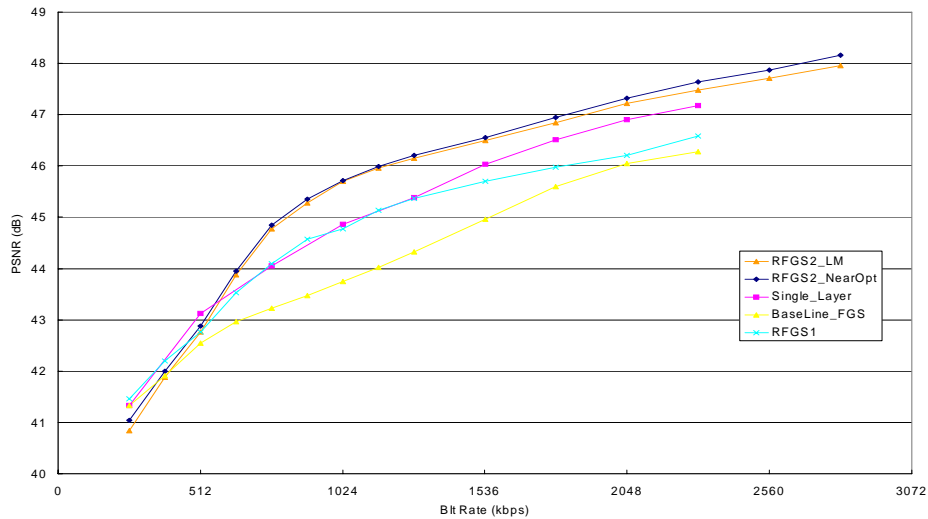


Figure 3.10. PSNR versus bitrate comparison between FGS, RFGS and single layer coding schemes for the Y component of the Akiyo sequence, where β is 3. We use three different coding schemes including ‘RFGS1’, ‘RFGS2_NearOpt’, and ‘RFGS2_LM’ in the experiments. ‘RFGS1’ uses the RFGS algorithm for the enhancement layer only. ‘RFGS2’ uses the RFGS algorithm for both the enhancement and the base layers. ‘NearOpt’ means the result of the near-optimal approach and ‘LM’ means the results using the proposed linear model.

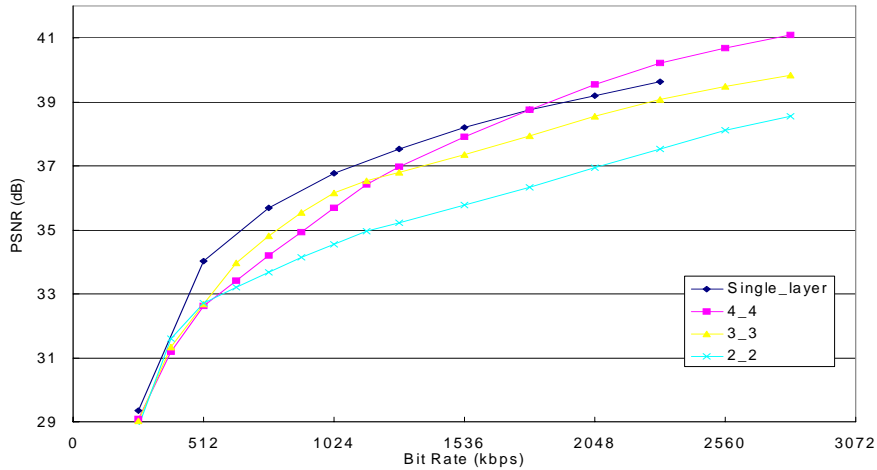


Figure 3.11. PSNR versus bitrate comparison between various values of RFSG parameter β for the Y component of the Foreman sequence, where the leak factor α is selected with the proposed linear model.

3.4.2 The Number of Bitplanes

Similarly, we can encode video sequences using different combinations of enhancement layer β and base layer β (denoted as β_e and β_b , respectively), where α_e and α_b are computed with the proposed linear model. Empirically, we find that performance is better when 2 to 4 bitplanes are used for coding. By applying all possible combination of β_e and β_b within a specified range to the whole sequence, we found that the coding efficiency with identical β for both layers is better than that with distinct β for each layer. The optimal β can be selected based on the range of the target bandwidth. When the target bandwidth is smaller than 512 kbps, the experiments in **Figure 3.11** show that the RFGS with $\beta=2$ has the best performance. When the bandwidth is from 256 kbps to 1.2 Mbps, the RFGS with $\beta=3$ provides the maximal gain in PSNR for most bitrates. When the bandwidth is even higher, the RFGS takes 4 bitplanes to achieve the optimal average coding efficiency. Thus, the number of bitplanes is selected based on the target range of the channel bandwidths. Our framework provides a flexible support for all of them.

3.5 Experiment Result and Analyses

Extensive experiments have been performed to demonstrate the performance of the proposed RFGS coding technique. From **Figure 3.10** to **Figure 3.9**, the coding efficiency of the RFGS is compared with those of the baseline FGS coding ('Baseline_FGS') and the single layer non-scalable coding schemes ('Single_layer'). These two techniques are considered as the lower and upper bounds for the performance. There are 3 different coding schemes for the RFGS. The scheme, labeled as 'RFGS1', uses the RFGS algorithm for the enhancement layer only. The other schemes, denoted as "RFGS2_NearOpt" and "RFGS2_LM", adopt the RFGS algorithm for both the enhancement and the base layers simultaneously as mentioned in section 3.3.4. The 'RFGS2_NearOpt' provides the near-optimal results and the 'RFGS2_LM' denotes the results by selecting the parameters based on the proposed linear model in the Section 4.1. In **Figure 3.12** and **Figure 3.13**, we compare the performance of the RFGS that selects the leak factor based on the proposed linear model with that of the macroblock-based PFGS [18]. All performance comparisons among the FGS, PFGS, RFGS and single layer coding schemes are based on the reconstructed video quality in PSNR for the given bitrate.

3.5.1 The Testing Conditions

From **Figure 3.10** to **Figure 3.9**, we adopt the testing condition B of the core experiments as specified by the MPEG-4 committee [17] and the MPEG-4 reference encoder with the Advanced Simple Profile for the base layer. In these experiments, the three sequences including Akiyo, Foreman, and Coastguard of CIF format are used for testing. For each sequence, every GOV has size of 60 frames that consist of one *I*-picture, 19 *P*-pictures, and two *B*-pictures between each pair of *P*-pictures. To derive the motion vectors for *P*-pictures and *B*-pictures, a simple half-pixel motion estimation

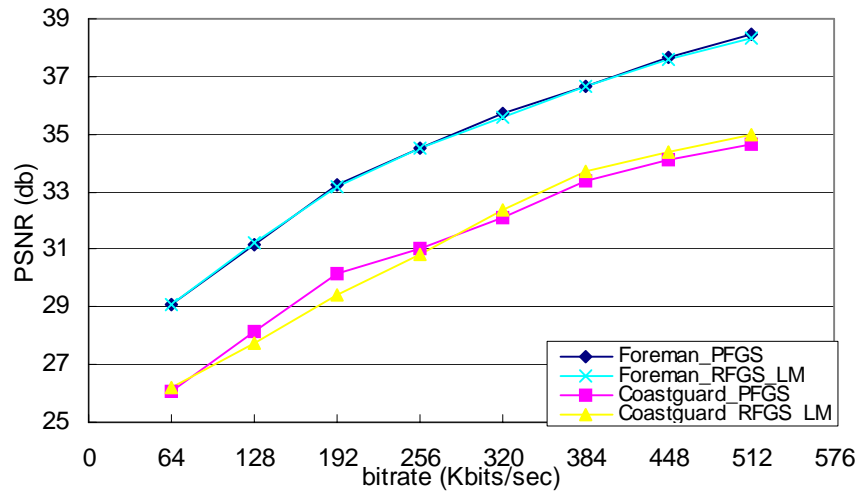


Figure 3.12 PSNR versus bitrate comparison between RFGS and PFGS for the Y component of the Coastguard and Foreman sequences in CIF format using the test condition A in the MPEG document m6779 [18]. For RFGS, β is 3.

scheme using linear interpolation is used. The search range of the motion vectors is set to ± 31.5 pixels. The bitrate of the base layer is 256 kbps with TM5 rate control, and the frame rate is 30 Hz. To simulate the possible channel bandwidth variation, the total bitrate of the enhancement layer bitstream is truncated to bitrate ranging from 0 to 2048 kbps with an interval of 128 kbps. In each category, a simple frame-level bit allocation with a truncation module is used in the streaming server to obtain optimized quality for the given bandwidth.

For **Figure 3.12** and **Figure 3.13**, we follow the testing condition A and B as described in [18]. The Foreman and Coastguard sequences of CIF format are used for simulation, where only one GOV and no B-picture are used. For the testing condition A, the bitrate of the base layer is 64 kbps and the TM5 rate control is adopted with frame rate of 5 Hz. The enhancement layer bitstream is truncated to the bitrates ranging from 0 kbps to 448 kbps with an interval of 64 kbps. For the testing condition B, the bitrate of the base layer is 128 kbps and TM5 rate control with frame rate of 10 Hz. The

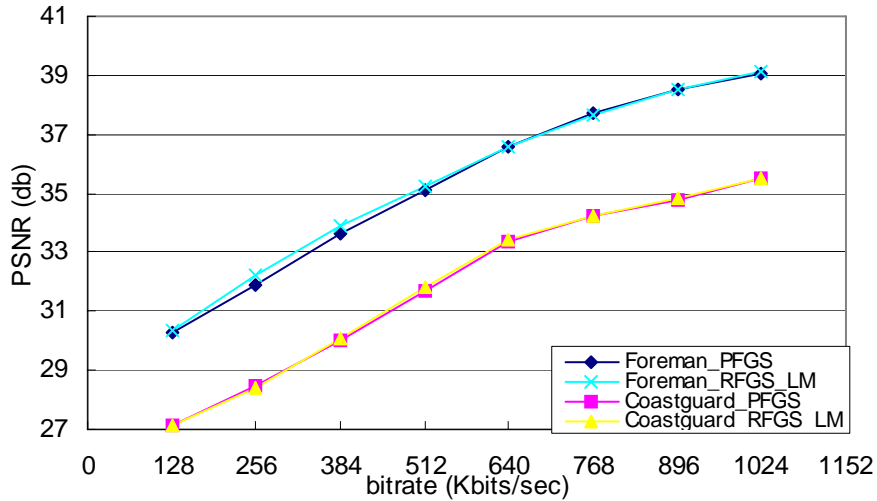


Figure 3.13 PSNR versus bitrate comparison between RFGS and PFGS for the Y component of the Coastguard and Foreman sequences in CIF format using the test condition B from the MPEG document m6779 [18]. For the RFGS, β is 3.

enhancement layer bitstream is truncated to bitrates ranging from 0 kbps to 896 kbps with an interval of 128 kbps.

3.5.2 Performance Comparisons

For the three specified test sequences, we first show the performance of the RFGS schemes with the GOV structure with *B*-pictures. From **Figure 3.10** to **Figure 3.9**, as compared to baseline FGS, our results show that the RFGS has improved by about 2 dB in PSNR for the fast motion sequences such as Foreman and Coastguard and improves up to 1.1 dB for the slow motion sequence such as Akiyo over the baseline FGS. When the RFGS method labeled as ‘RFGS2_LM’ is applied for both layers, there are up to 3.6 dB and 4.1 dB gain in PSNR over the baseline FGS for the Foreman and Coastguard sequences, respectively. For the Akiyo sequence, the RFGS also has 2.0 dB gain in PSNR over the baseline FGS. To compare with the single layer approach, the RFGS scheme has a 0.6 to 1.3 dB loss under the various bitrates for the Foreman sequence. For the Coastguard sequence, as compared to the single layer approach, the RFGS has

1.4 dB loss in PSNR at low bitrate and the almost identical PSNR values at medium and high bitrates. Additionally, the RFGS for the Akiyo sequence is actually better than the single layer approach by around 0.3 to 0.9 dB at medium and high bitrates.

It is interesting that the RFGS2 outperforms the single layer at high bitrate for the slow motion sequences. For the single layer approach only one VLC table is used and it can't be optimal for a wide range of bitrates. In the FGS approach, however, the different bitplanes have their own VLC tables that can approach to the entropy of the DCT coefficients at both low bitrate and high bitrate. The RFGS2 algorithm removes most of the temporal redundancy and reduces the dynamic range of the residuals. It can encode more efficiently using better VLC tables designed for the high bitrate.

When only the base layer bitstream is decoded for the extremely low bitrate case, all the three sequences have the PSNR values worse than the PSNR by the single layer by about 0.3 to 0.5 dB because the RFGS2 uses the enhancement layer information for the base layer prediction. Since there is no leaky factor applied for the base layer, we have error drift even when α_b is small. Considering the significant improvement at the medium and high bitrates, the modest loss of PSNR value at the base layer is acceptable.

Now we compare the results of the RFGS with the macroblock-based PFGS [18] based on the GOV structure without the use of *B*-pictures and the rate control scheme defined in section 3.3.5. The experiments show that the error drift for RFGS2 is more serious since all the frames are *P*-pictures and all of their errors are propagated. Therefore α_b should be set as zero to eliminate the drift at low bitrate. For **Figure 3.12** and **Figure 3.13**, the frame based RFGS results are quite close to the macroblock based PFGS [18]. It should be mentioned that identical linear model of the enhancement layer are used to compute α_e .

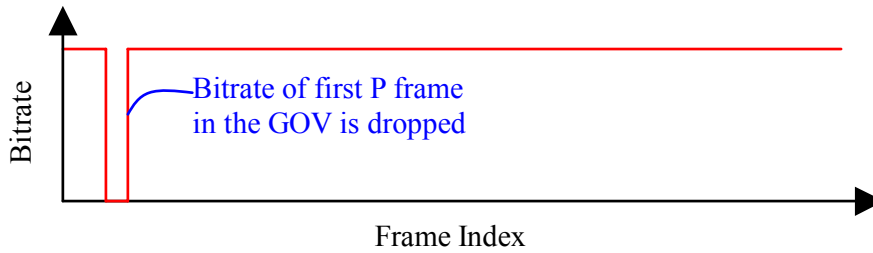


Figure 3.14. Sample bandwidth profile to test the error recovery capability of the RFGS technique.

3.5.3 Test for Error Recovery Capability

To verify the error recovery capability of the RFGS, a simple experiment is performed to demonstrate the worst-case scenario when there is bandwidth variation that can result in maximal effect of drift. We assume the network bandwidth is sharply dropped for every first P -picture transmitted of each GOV and the bit budget for the other frames is set as 1024 kbps. Such a bandwidth scenario is illustrated in **Figure 3.14**. Since only the first P -picture for the enhancement layer is lost and the degradation of the subsequent frames will be caused only by the errors from this P -picture. The same testing conditions and the video sequences are used as in [17]. To verify the error attenuation of RFGS mentioned in the Section 3.3.3, we first examine the RFGS1 method about the speed of the error recovery for various α . In all the simulations, β is set as 3 and α equals to one of the four predefined values, 0.5, 0.75, 0.9, and 1.0. As shown in **Figure 3.15**, the error attenuation capability of the RFGS framework is strongly affected by the value of α used. At the worse case scenario that no enhancement bit is received, the PSNR loss is more than 5 dB as compared to the PSNR under an error-free condition. For a small α of 0.5, the error is attenuated very fast. For example, in **Figure 3.15**, after fourth P -pictures within the first GOV, the PSNR differences are reduced to about 0.1 to 0.3 dB. When α equals to unity, as shown in the fourth GOV in **Figure 3.15**, the drift lasts for a long time. We provide the performance

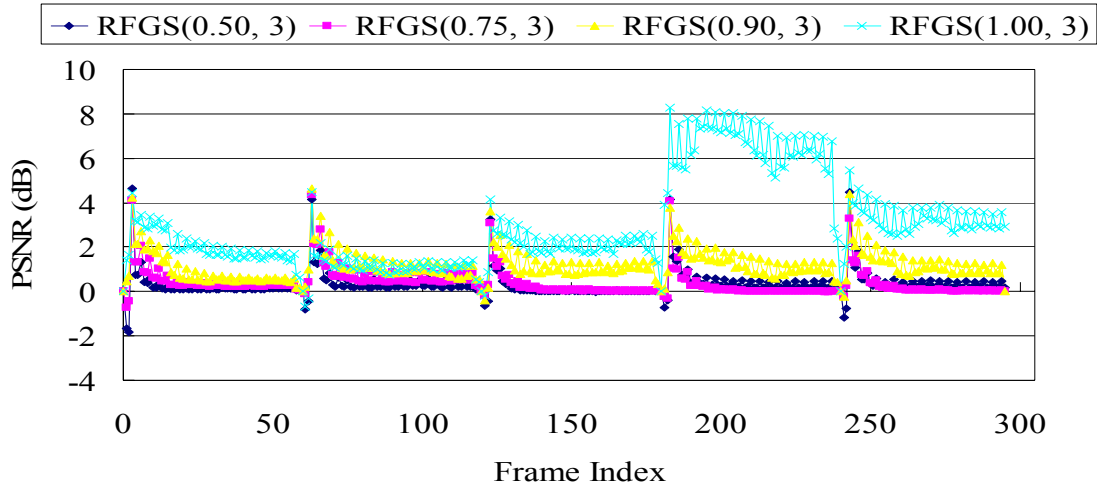


Figure 3.15. The error attenuation in PSNR for the Y component of the Akiyo sequence under different α in the RFGS1 framework, where the pair of the values indicates the prediction mode parameters (α, β) .

of RFGS2_LM under the burst error in **Figure 3.16**. We simulate the burst error with a loss of the first few frames in every GOV. Two burst lengths of one frame and seven frames are used for simulation. By applying the RFGS method for both the enhancement and base layers, the error drift is more serious as compared the drift for the RFGS1. However, the visual quality can still be fast recovered from the burst errors.

We also perform the dynamic test following the channel bandwidth variation pattern as defined in [19] to demonstrate the performance of RFGS. The bandwidth pattern as illustrated in **Figure 3.2** are as follows. The total bandwidth is switched in a step size of 256 kbps that decreases from 1024 kbps to 256kbps and increases back to 1024 kbps. The instantaneous bitrate is held for 24 seconds (or 720 frames with frame rate 30). Other test conditions are identical to those described in Section 3.5.1 and as defined in [17]. In the simulation, the Novel sequence in CIF format and with the frame rate of 30 fps were used. The first 5040 frames of the sequence are used for testing and the base layer is coded at 256 kbps. During transmission, we use the 2-Level Priority

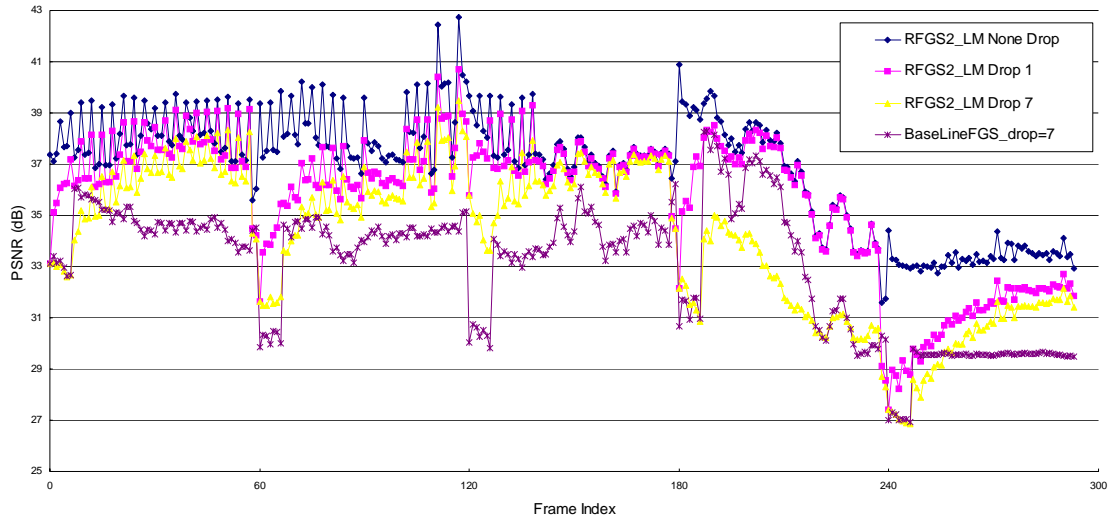


Figure 3.16. The error attenuation in PSNR for the Y component of the Foreman sequence using the RFGS2_LM framework. All the curves denote truncation of the enhancement layer bitstream at 1024kbps. For the curve labeled ‘RFGS2_LM Drop 1’, the first frame of each GOV is dropped. For the curve labeled ‘RFGS2_LM Drop 7’, the first seven frames of each GOV are dropped. For the curve labeled ‘RFGS2_LM None Drop’, no frame is dropped. The curve labeled ‘BaseLineFGS_drop=7’ is the baseline FGS with the first 7 frames of each GOV dropped.

Network, where the FGS base-layer is set at high priority. When the bandwidth is small, the base layer will be sent first. For the single layer approach, we encode the bitstream with 256kbps, 512kbps, 768kbps, and 1024kbps and dynamically select the appropriate bitstreams for the target bitrates as defined in [19].

Figure 3.18 shows the simulation results. As compared with the results based on the single layer and the baseline FGS approaches, the results show that the RFGS2 with the linear model can adaptively select the suitable α offline to achieve similar performance as that of the single layer approach for given dynamic bandwidths and different scene over a long sequence.

As for the error recovery speed for different sequences, as shown in **Figure 3.17**, it is observed that the error recovery is also related to the temporal dependency between the successive frames of the same sequence. For the fast moving sequences like Coastguard and Foreman, the current frame only refers to a fraction of information from

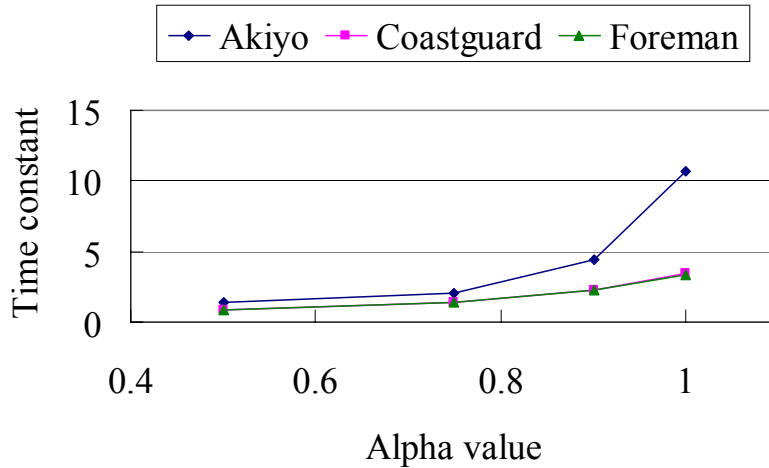


Figure 3.17. The relationship between the leak factor α and the time constant τ for the error attenuation. For each curve, β is 3.

the reference frame, which allows limited error propagation. Thus, the errors vanish even with a larger leak factor α . For the slow motion sequences such as Akiyo, most of the frames consist of static areas such that there exist strong dependencies between the consecutive frames of the sequence. The dependencies can improve the coding efficiency but suffers from more drift when the transmission bandwidth is insufficient. Therefore, the RFGS with a small α (about 0.5) is recommended for the slow motion video sequences to improve the error robustness.

3.6 Summary

In this chapter, we proposed a novel FGS coding technique RFGS. The RFGS is a flexible framework that incorporates the ideas of leaky and partial predictions. Both techniques are used to provide fast error recovery when part of the bitstream is not available. The RFGS provides tools to achieve a balance between coding efficiency, error robustness and bandwidth adaptation. The RFGS covers several well-know techniques such as MPEG-4 FGS, PFGS and MC-FGS as special cases. Because the RFGS uses a high quality reference, it can achieve improved coding efficiency. The

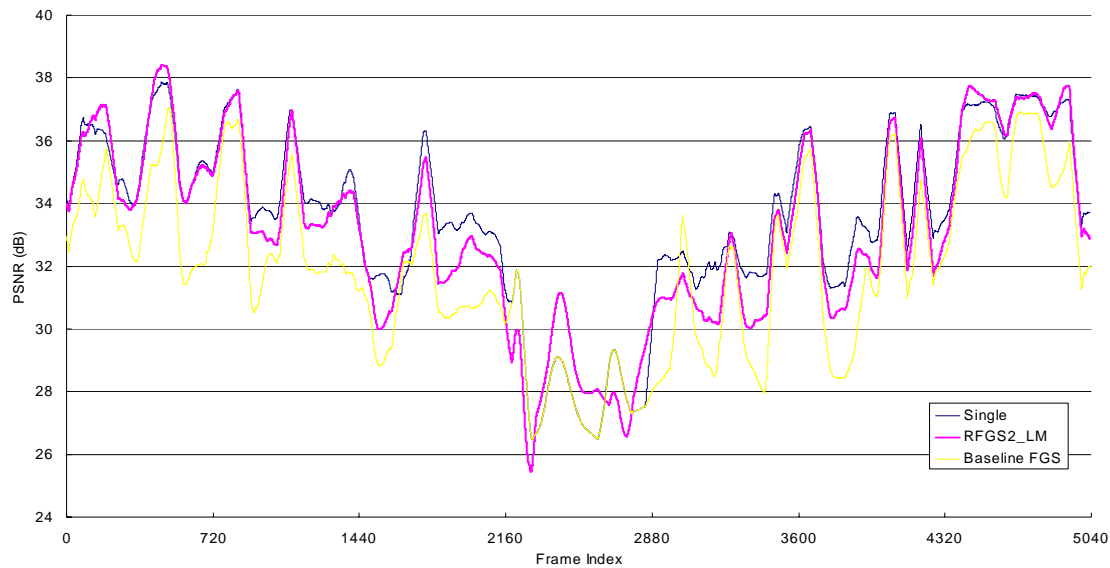


Figure 3.18. The comparison of visual quality in PSNR between FGS and single layer approaches with the dynamic test condition as defined in the MPEG document m8002 [19].

adaptive selection of bitplane number can be used to allow the tradeoff between coding efficiency and error robustness. The coding efficiency is maximized for a range of the target channel bandwidth. The enhancement layer information is scaled by a leak factor α , where $0 \leq \alpha \leq 1$ before adding to the base layer image to form the high quality reference frame. Such a leak factor is also used to alleviate the error drift.

Our experimental results show that the RFGS framework can improve the coding efficiency up to 4 dB over the MPEG-4 FGS scheme in terms of average PSNR. The error recovery capability of RFGS is verified by dropping the first few frames of a GOV at the enhancement layer. It is also demonstrated that tradeoff between coding efficiency and error attenuation can be controlled by the leak factor α . We also provide an approach to select the parameters and its performance approaches that of a near-optimal exhaustive search of parameters. Such a technique provides a good balance between coding efficiency and error resilience.

CHAPTER 4

Stack Robust Fine Granularity Scalability (SRFGS)

4.1 Introduction

Several research works are proposed to improve the temporal prediction efficiency while keeping the features of fine granularity and robustness of MPEG-4 FGS, as discussed in [8]. Among these approaches, the Robust FGS (RFGS) that described in Chapter 3 multiplies the temporal prediction information by a leaky factor α , where $0 \leq \alpha \leq 1$, to strengthen the error resilience and leads to good tradeoff between coding efficiency and error robustness.

To verify the improvement of the new SVC techniques after the MPEG-4 FGS [6], the MPEG committee issued a Call for Evidence on Scalable Video Coding (CFE on SVC) [20]. In the CFE on SVC, we proposed the Stack Robust Fine Granularity Scalability (SRFGS) to improve the temporal prediction efficiency of RFGS and provides temporal and SNR scalability. The SRFGS was reviewed by the MPEG committee in [21] and ranked as one of the best algorithms according to the subjective test in [23].

In this chapter, we describe the SRFGS technique in detail. In Section 4.2, we propose a simplified RFGS architecture. It significantly reduces the complexity of the RFGS architecture while maintaining the same performance. It leads to easier understanding on the basic prediction concept used in the RFGS enhancement layer.

Based on the simplified architecture, in Section 4.3, the prediction concept of SRFGS is introduced. Section 4.4 shows the detailed encoder and decoder structures of SRFGS. To optimize the coding efficiency of SRFGS, a novel macroblock-based alpha adaptation and the prediction architecture for the B frames are discussed. Single-loop enhancement layer decoder architecture is proposed to reduce the complexity of SRFGS decoder. In Section 4.5, the simulation results demonstrate the improvement of SRFGS as compared to RFGS. The comparison with AVC is also shown. Finally, the summary is given in Section 4.6.

4.2 Simplified RFGS Prediction Scheme

Figure 4.1 shows the original RFGS encoder architecture as proposed in [8] and [24]. The enhancement layer bitstream is generated with the following process. The motion compensation module of the enhancement layer uses the base layer motion vectors and the high quality reference image *HQRI* stored in the enhancement layer frame buffer to generate the high quality prediction image *ELPI*. The enhancement layer motion compensated frame difference $MCFD_{EL}$ is computed by subtracting *ELPI* from the original signal *F*:

$$MCFD_{EL,i} = F_i - ELPI_i = F_i - (HQRI_{i-1})_{mc} \quad (4.1)$$

, where the subscripts *i* and *i-1* mean the current frame time *i* and the previous frame time *i-1*, respectively. The subscript *mc* means that $(y)_{mc}$ is the motion compensated version of *y*. The signal \hat{D} is computed by subtracting the reconstructed base layer DCT coefficients \hat{B} from the $MCFD_{EL}$:

$$\hat{D}_i = MCFD_{EL}^i - \hat{B}_i \quad (4.2)$$

The signal \hat{D} is entropy encoded to generate the enhancement layer bitstream. Note that for simplicity and also due to the linearity of DCT, in this chapter we use same notation for the symbol in spatial and transform domain.

The high quality reference image $HQRI$ at the enhancement layer is generated as follows. The first β bit planes of the difference signal \hat{D} is summed up with \hat{B} . The resultant signal is converted back to the spatial domain using the IDCT transform and summed up with $ELPI$ to get the enhancement layer reconstructed image $ELRI$.

$$ELRI_i = (HQRI_{i-1})_{mc} + \hat{B}_i + \hat{D}_i \quad (4.3)$$

It should be noted that for simplicity we assume all of the bit planes in \hat{D}_i will be used in the enhancement layer prediction loop. The base layer reconstructed signal B will be subtracted from the signal $ELRI$ to get the signal D with only enhancement layer information. The signal D will be attenuated by a leak factor α and added back the signal B before storing into the enhancement layer reference frame buffer. Thus, we have the following relationship:



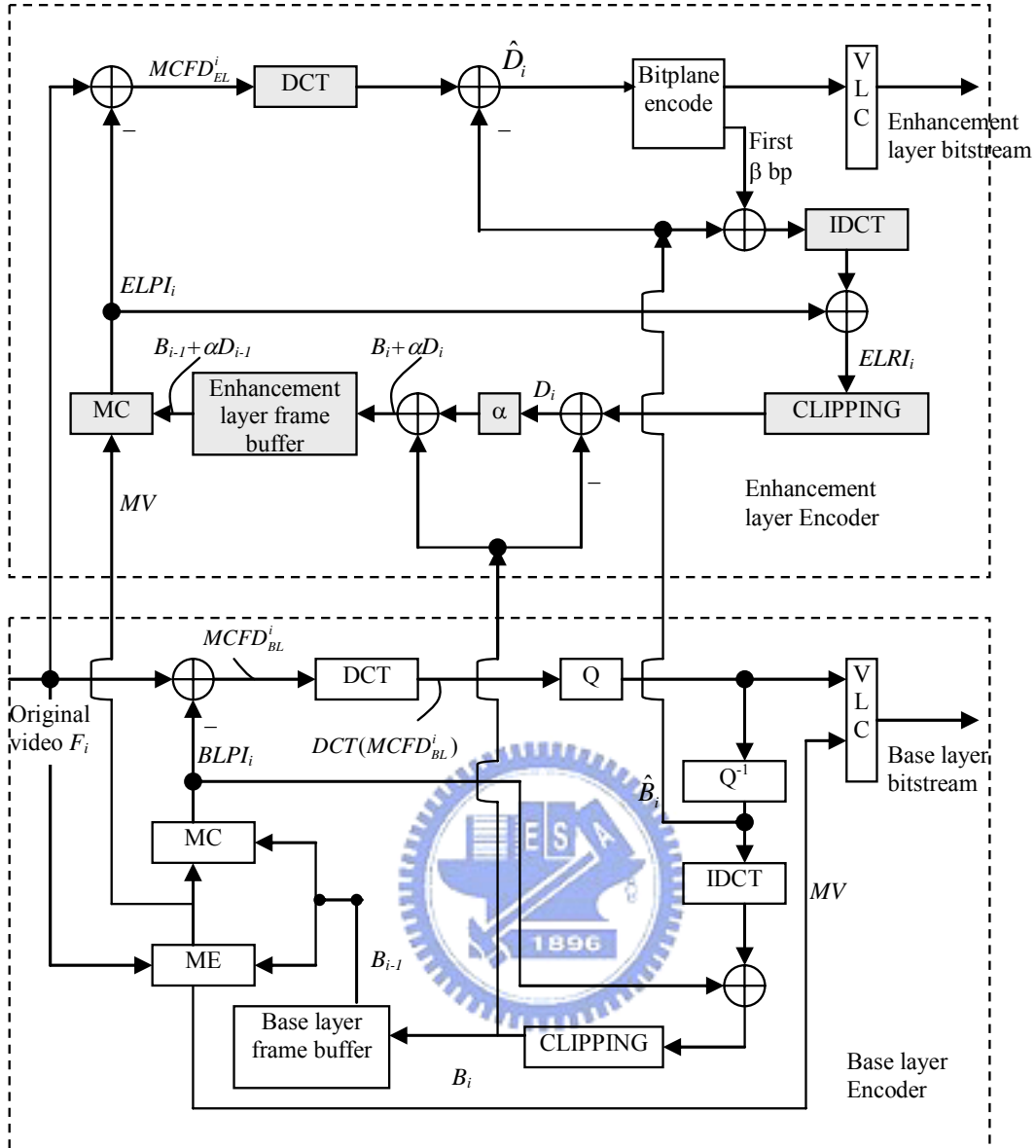


Figure 4.1 The original RFGS encoder

$$HQRI_i = B_i + \alpha D_i \quad (4.4)$$

The rationale for performing the attenuation process on the signal D is that we want the errors to be attenuated for all the past frames recursively. If the attenuation process is only applied to the first few bit planes of \hat{D} , only the errors occurred in the current frame are attenuated. The errors occurred earlier are still accumulated for the subsequent frames through the motion prediction loop without attenuation.

Although the RFGS prediction architecture efficiently reduce the drift error, it is quite complex. The base layer needs to store the reconstructed DCT coefficient \hat{B} . The enhancement layer firstly subtracts \hat{B} from the prediction error $MCFD_{EL}$ to reduce the entropy in the signal \hat{D} , and then it uses \hat{B} to form the ELRI. The enhancement layer further accesses the base layer reconstructed image B to generate the signal D with only the enhancement layer information and to generate the $HQRI$ stored in the enhancement layer frame buffer. This prediction scheme increases requirement for both memory and memory access bandwidth. Further, with this complex prediction architecture, the prediction concept of RFGS is difficult to grasp and make new improvements.

Thus, we will simplify the prediction scheme while maintaining the same coding efficiency. From equation (4.3) and (4.4), we can get the following relationship:

$$ELRI_i = (B_{i-1} + \alpha D_{i-1})_{mc} + \hat{B}_i + \hat{D}_i \quad (4.5)$$

By grouping the base layer information and the enhancement layer information, equation (4.5) becomes

$$ELRI_i = (B_{i-1})_{mc} + \hat{B}_i + (\alpha D_{i-1})_{mc} + \hat{D}_i = B_i + D_i \quad (4.6)$$

, where

$$B_i = (B_{i-1})_{mc} + \hat{B}_i \quad (4.7)$$

and

$$D_i = (\alpha D_{i-1})_{mc} + \hat{D}_i. \quad (4.8)$$

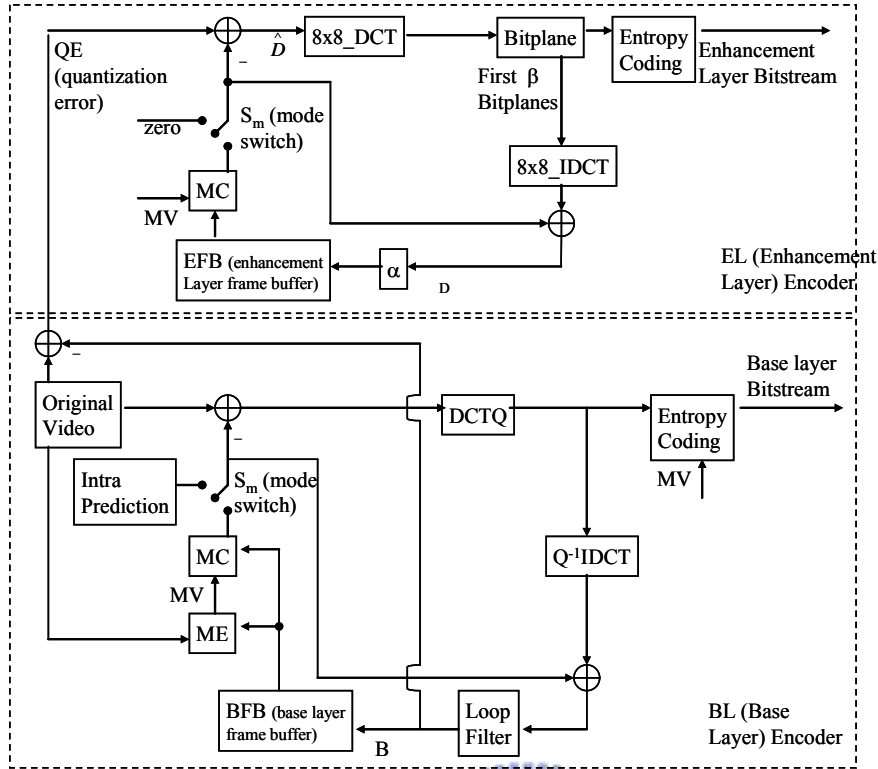


Figure 4.2 The simplified RFGS encoder

From (4.8) we know that the residue D can be derived simply from accumulating the signal \hat{D} in all the previous frames. From equations (4.1) and (4.4), we can re-write the derivation of the signal \hat{D}_i in (4.2) as:

$$\begin{aligned}
 \hat{D}_i &= MCFD_{EL-i} - \hat{B}_i \\
 &= F_i - (HQRI_{i-1})_{mc} - \hat{B} \\
 &= F_i - (B_{i-1} + \alpha D_{i-1})_{mc} - \hat{B}
 \end{aligned} \tag{4.9}$$

Again, by grouping the base layer information and the enhancement layer information, equation (4.9) becomes

$$\hat{D}_i = F_i - (B_{i-1})_{mc} - \hat{B} - (\alpha D_{i-1})_{mc} = F_i - B_i - (\alpha D_{i-1})_{mc} \tag{4.10}$$

The difference between the original frame F and the base layer reconstructed image B is actually the quantization error QE at the base layer,

$$QE_i = F_i - B_i \tag{4.11}$$

Thus, equation (4.10) becomes

$$\hat{D}_i = QE_i - (\alpha D_{i-1})_{mc} \quad (4.12)$$

From (4.8) and (4.12), we realize that the only signal that the enhancement layer acquires from the base layer is the base layer quantization error QE , all the other signals can be generated by the enhancement layer itself. With this analysis, we can derive a simplified RFGS prediction scheme as shown in **Figure 4.2**, and it still provides identical functionality with the original RFGS prediction scheme as shown in **Figure 4.1**. In the simplified architecture, the base layer quantization error QE will be predicted with the reference frame stored in the enhancement layer frame buffer EFB. This step performs the equation (4.12) in **Figure 4.1**. The prediction error \hat{D}_i will be transformed and bit plane coded as FGS bitstreams. The first β bit planes will be inversely transformed and added back with the prediction to generate the signal D . This step performs the equation (4.8) in **Figure 4.1**. The resultant signal D will multiply by α for leaky prediction before it is stored in the frame buffer. The simplified RFGS architecture significantly reduces the complexity of the RFGS. The base layer encoder needs not store the reconstructed base layer DCT coefficient \hat{B} . The enhancement layer encoder needs not access and perform the computation with the base layer signal \hat{B} and B . The enhancement layer encoder architecture is just like the base layer encoder replacing the original signal from F with the base layer quantization error QE .

4.3 Enhanced Prediction Architecture Using Stack Concept

With the simplified RFGS architecture, it is also easier to understand the prediction concept within the RFGS structure. In the RFGS structure, the base layer quantization error QE , which is intra coded in the MPEG-4 FGS scenario, is temporally

predicted by the previous enhancement layer information to remove the temporal redundancy. The leaky factor α is used to attenuate the drift error at decoder side when only partial enhancement layer reference information is reconstructed. Smaller leaky factor α leads to less amount of drift. However, smaller α leads to less performance when all of the reference enhancement layer information is received but only partial information is used for removing temporal redundancy. The other factor β , which denotes the number of bit planes used in the enhancement layer prediction loop, plays a key role in the RFGS structure, too. Larger β leads to more enhancement layer information used for temporal prediction. With the removal of more temporal redundancy, larger β provides better performance when all the reference bit planes are fully reconstructed. However, larger β may lead to larger drift error at lower bitrate as less amount of required reference information is available for motion compensation. In summary, smaller β reduce the drift at lower bitrate at the expense of coding efficiency because the bit planes after β effectively become intra-coded with less coding performance.

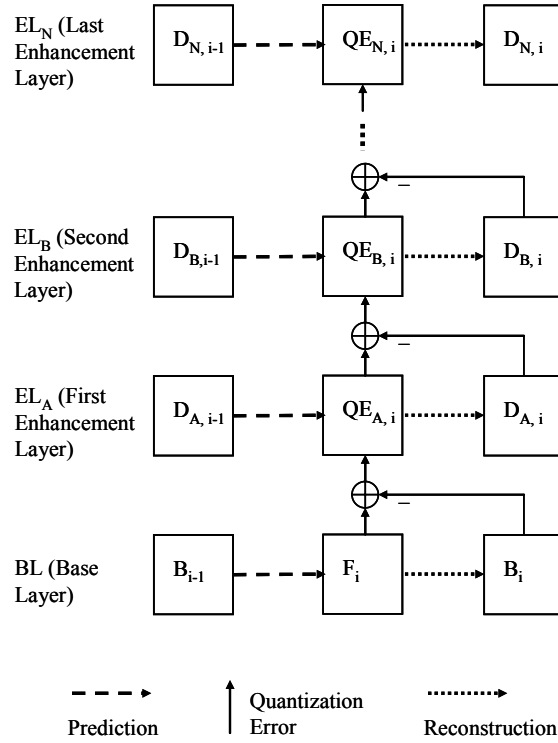


Figure 4.3 SRFGS prediction concept

To address the temporal redundancy removal and the drift reduction, a novel architecture, namely Stack RFGS (SRFGS), is proposed. In the SRFGS, the prediction scenario is generalized from that of RFGS as follows: The quantization error of the previous layer is temporally predicted by the reconstructed frame in the previous time instance of the current layer. We utilize this generalized prediction concept and further extend the architecture to multiple layers in SRFGS as illustrated in **Figure 4.3**. At time instant i , the original Frame F_i is predicted by the base layer reconstructed frame of time $i-1$, which is denoted as B_{i-1} . The quantization error $QE_{A,i}$ is computed as the difference between F_i and the reconstructed base layer B_i . The signal $QE_{A,i}$ is predicted by the first enhancement layer reconstructed frame at time instant $i-1$, which is $D_{A,i-1}$. At the second layer EL_B , the quantization error $QE_{B,i}$ is computed as the difference between $QE_{A,i}$ and the reconstructed first enhancement layer $D_{A,i}$. The signal $QE_{B,i}$ will be predicted by the second enhancement layer reconstructed frame at time $i-1$, which is $D_{B,i-1}$. With this

concept, the RFGS enhancement layer prediction scheme is generalized to multi-layer stack architecture. The coding performance of EL_A in SRFGS is the same as the first β bit planes in RFGS, since the temporal redundancy has been removed in both of them. However, the coding performance in EL_B (and all the following layers) of SRFGS is superior to the remaining bit planes of RFGS, because the temporal redundancy is only removed in SRFGS.

4.4 The Stack RFGS System Architecture

In this section we firstly describe the encoder and the decoder block diagrams of the SRFGS architecture. An optimized macroblock-based alpha adaptation is then introduced to increase the coding performance. The prediction scheme for the B-frame is described, too. We further propose a single-loop enhancement layer decoder architecture to reduce the SRFGS decoder complexity.

4.4.1 Functional Description

Based on the stack concept, the AVC-based SRFGS encoder in **Figure 4.4** is constructed. The prediction scheme at SRFGS base layer is the same as that in RFGS, except that there is no high quality base layer reference in SRFGS. The high quality base layer reference will not be used in the AVC-based SRFGS architecture to prevent drift at low bitrate. The first enhancement layer of SRFGS, as denoted as EL_A , is identical to that in RFGS except in two aspects. Firstly, only the first β_A bit planes are coded and written into the enhancement layer bitstream. Secondly, the multiplication of the leaky factor α_A is moved after the motion compensation module. All the enhancement layer loops have the identical architecture as that in EL_A , except the last enhancement layer loop EL_N . In EL_N , the entire residues are bit plane coded to achieve perfect reconstruction at the decoder.

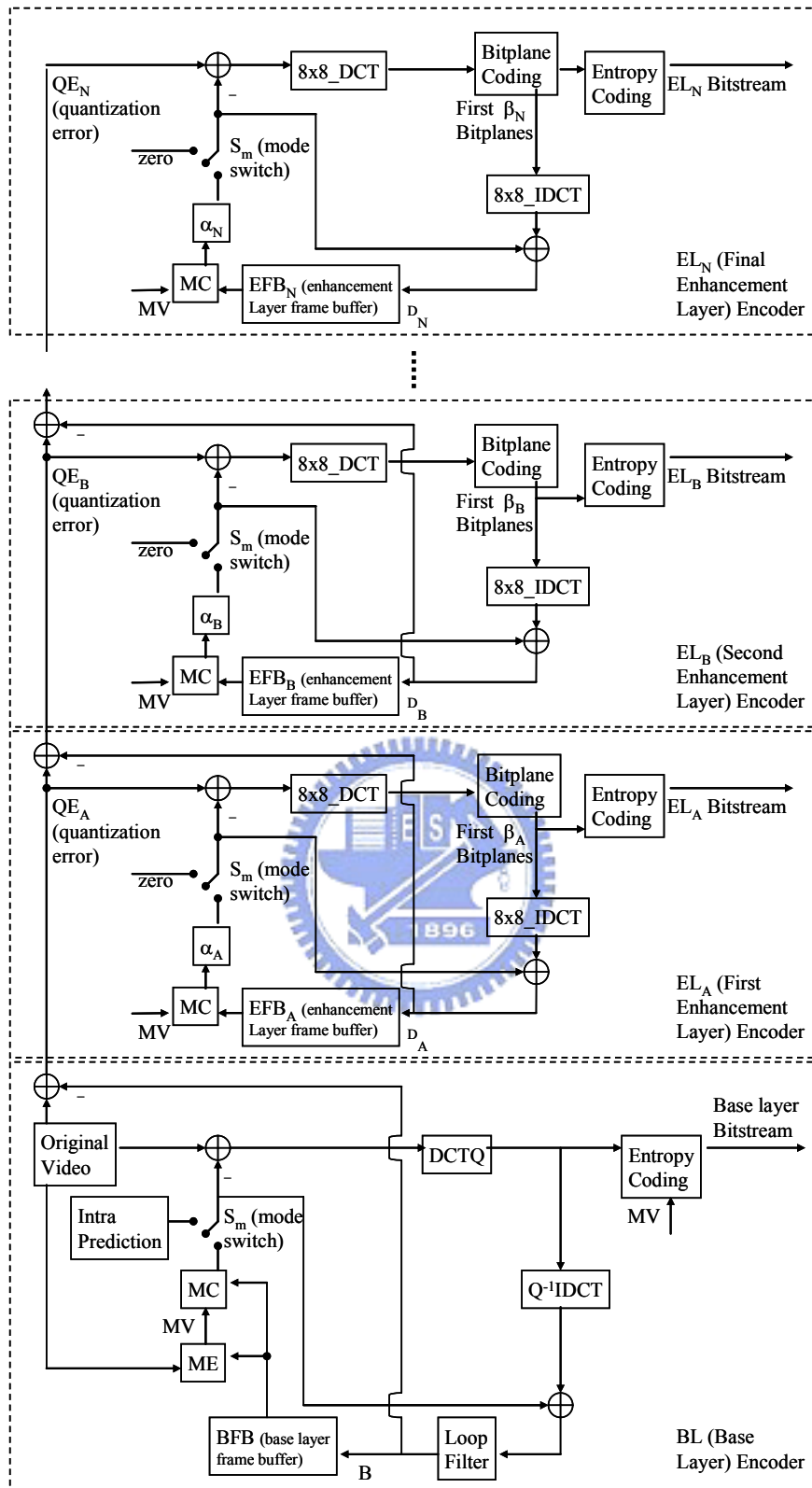


Figure 4.4 Diagram of the SRFGS encoder framework

A similar scheme as the improved motion estimation algorithm by He et al [25] is utilized in SRFGS. He et al. derive a motion vector that is adequate for both the base

and the enhancement layer information [25]. Based on this improved ME algorithm, the base and entire enhancement layer information is embedded into the stack architecture. With the derived motion vector through the improved ME module, the base layer mode decision module selects the best mode using the AVC mode decision algorithm. Consequently, the same coding mode and motion vector is used for the base and entire enhancement layer prediction loops.

At the decoder side, as shown in **Figure 4.5**, the received information of each loop will be decoded by its own loop and summed up with the base layer reconstructed image to construct the final image. For each loop, if only partial bitstream is received, the leaky factor α can attenuate the drift error as in the RFGS case. If there is no information received for a loop, the leaked motion compensated information will directly be stored back to the frame buffer. In the proposed framework, the information of each prediction loop is not used or affected by the information in the other loops. Consequently, if there is any error in a loop, it won't affect the data in the other loops. This intrinsic error localization property of SRFGS offers better performance in an error-prone environment.

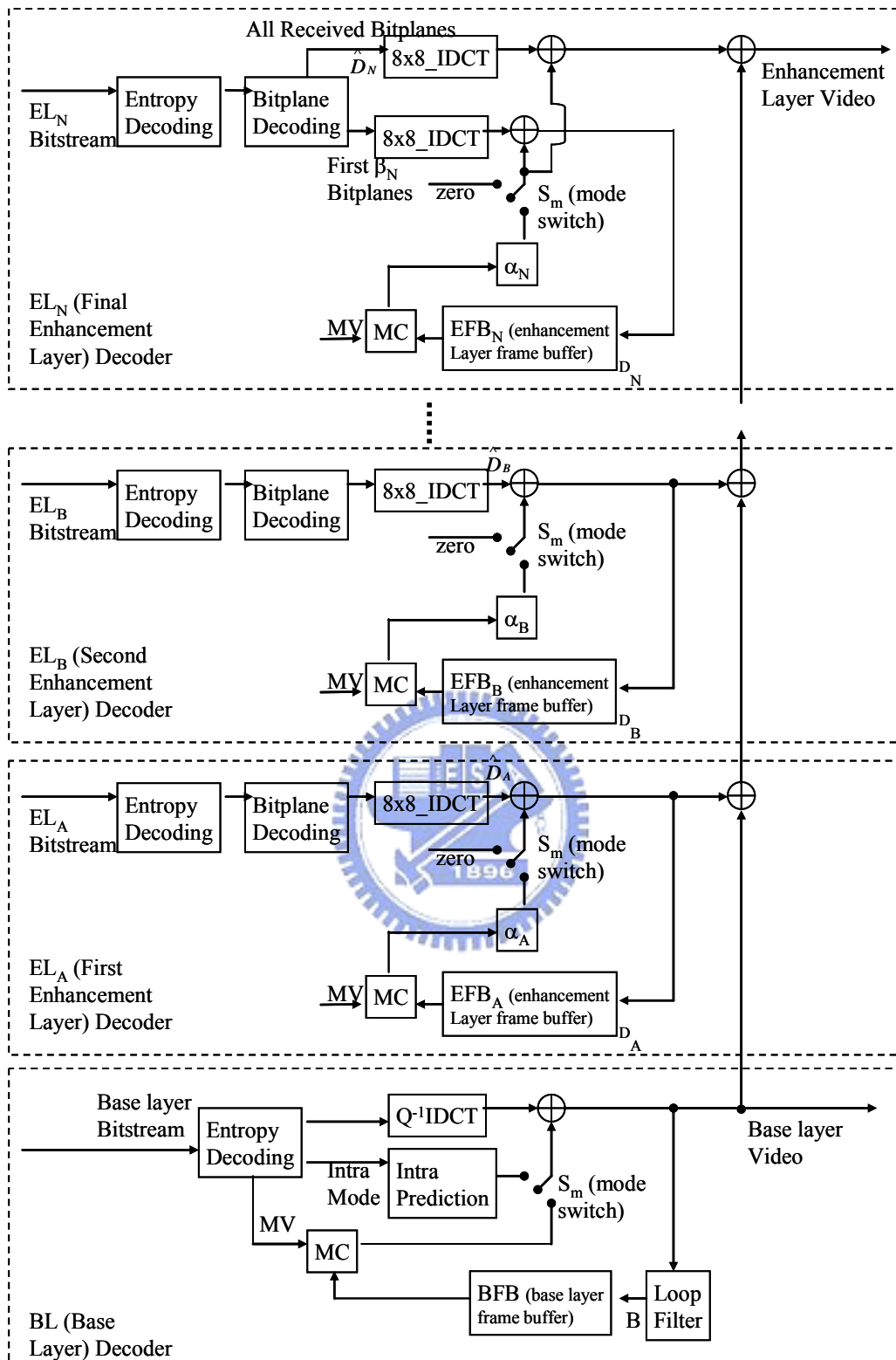


Figure 4.5 Diagram of the SRFGS decoder framework

More enhancement layer loops mostly lead to better coding performance. This sometimes may not be true because the temporal prediction not only reduces the energy

Frame Startcode	EL _A Header	EL _A 1 st BP	...	EL _A β_A^{th} BP	EL _B Startcode & Header	EL _B 1 st BP	...	EL _B β_B^{th} BP	EL _N Startcode & Header	EL _N 1 st BP	...	EL _N Last BP
-----------------	------------------------	------------------------------------	-----	--	------------------------------------	------------------------------------	-----	--	-------	------------------------------------	------------------------------------	-----	-------------------------

Figure 4.6 The SRFGS enhancement layer bitstream format

of quantization error but also increases the dynamic range with some extra sign bits. To overcome this overhead, the size of the enhancement layer loop should be large enough, such that the residue energy reduced from the temporal prediction is larger than the overhead. Note that usually the higher the enhancement layer, the more the random of the residue. To reduce the same amount of the residue energy from the temporal prediction, we need more reference data (larger β) at higher enhancement layer. Further, the static sequences have more temporal correlation and hence fewer reference data (smaller β) is enough to overcome the overhead. At this case (static sequence), smaller β also reduce the drifting error at low bitrate. After determine the size of a enhancement layer based on its position and the sequence characteristic, same process can be used to set the size of the next enhancement layer if the bitrate range of the target application is not fully covered yet.

In **Figure 4.6**, it shows the enhancement layer bitstream format of the SRFGS coding scheme in a frame. Assuming that there is N enhancement layer loops, the bitstream firstly stored all the β_A bit planes of EL_A , which is the most significant loop. After β_A , we include all the β_B bit planes of EL_B , which is the second most significant loop. The similar processes are applied to code the remaining enhancement layers except EL_N . In EL_N , which is the last significant loop, not only the first β_N bit planes but also all the remaining bit planes are stored in the bitstream. Within each loop, the bit planes are ordered from MSB to LSB. Thus, the SRFGS bitstream is ordered by the importance of the information. With the bitstream, the SFGS server, operating in similar

fashion as the MPEG-4 FGS and RFGS server, can truncate the bitstream at any point to provide the best performance for that bitrate.

4.4.2 Optimized macroblock-based alpha adaptation

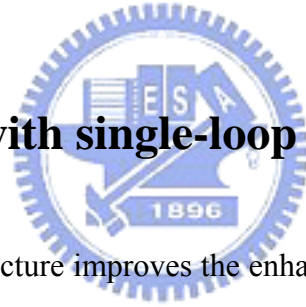
In the RFGS architecture [8] and [24], the value of α is adapted at frame level. Each macroblock in same frame use the same α . In this chapter, we generalize the α adaptation to macroblock level with simple optimization. The optimization is performed such that the handling macroblock has the least prediction error energy. As shown in **Figure 4.4**, the multiplication of α is placed after the motion compensation module. If the handling macroblock is selected as inter mode in the base layer mode decision module, the encoder will sweep the value of α between 0 and 1 to find the optimal value that minimizes the energy of the prediction error. Thus, we can find the best α for the handling macroblock in a very simple way. However, various values of α , coded in the macroblock header, cost significant overhead. In our approach, we further define a frame level α named **frame_** α . The **frame_** α is adapted at the frame level and uniquely coded at the header for each loop. Each macroblock can select the best α between 0 and **frame_** α . Thus for each macroblock, only one-bit flag is needed to indicate whether 0 or **frame_** α is used. In our simulation, this method provides a good tradeoff between energy and overhead reduction.

4.4.3 Prediction scheme of B-frame

The prediction scheme of B-frame in SRFGS is similar to that in RFGS. In RFGS, the base layer of B-frame is predicted by a high quality reference image that is the sum of the base and enhancement layer reconstructed images, denoted as $B+D$. In the SRFGS structure, the B-frame is predicted by the sum of the base and the entire

enhancement layer reconstructed images, which is $B+D_A+\dots+D_N$. The quantization error, which is the difference between the original and base layer reconstructed frames, is coded as the enhancement layer bitstream. There is no stack architecture in B-frame to reduce the complexity. Since no frame takes B-frame as reference, missing B-frame in the FGS server can support temporal scalability without any drift error for the following frames. The rate control algorithm allocates more bits for the P-frame at low bitrate to provide a better anchor frame. With this bit allocation, we can reduce the drift error of P-frame but also enhance the reference image quality of B-frame. The extra bits at high bitrate will be allocated to B-frames since the information carried by the MSB of B-frame is more important than that carried by the LSB in P-frame for averaged picture quality of reconstructed video.

4.4.4 Stack RFGS with single-loop enhancement layer decoder



Although the stack architecture improves the enhancement layer coding efficiency, it also significantly increases the complexity due to multiple loops. This is critical for a portable client device which is constrained by complexity and power. To address this issue, we propose a novel simplified SRFGS decoder that only requires single-loop enhancement layer decoding. Similar to equation (4.8), at each SRFGS enhancement layer decoder, the reconstructed information at that layer can be derived as:

$$D_{X,i} = (\alpha_{X,i-1} D_{X,i-1})_{mc(mv_{X,i-1})} + \hat{D}_{X,i} \quad (4.13)$$

,where X denotes the enhancement layer X . The signal $(y)_{mc(mv_{X,i-1})}$ denotes the motion compensated version of y using the motion vector $(mv_{X,i-1})$. In the current SRFGS structure, the motion vector of each layer is identical to that in the base layer. If we further constrain the encoder with the same leaky factor α for each layer, equation

(4.13) can be simplified as

$$D_{X,i} = (\alpha_{AllLayer,i-1} D_{X,i-1})_{mc(mv_{AllLayer,i-1})} + \hat{D}_{X,i} \quad (4.14)$$

That is, the signal D in each layer is attenuated with the same leaky factor $\alpha_{AllLayer}$, and then motion compensated by the same motion vector $(mv_{AllLayer,i-1})$. With this constraint, we need not separate the signal D for each layer and can merge them all. Thus, the equation (4.14) of multiple layers can be merged as:

$$(D_{A,i} + D_{B,i} + \dots + D_{N,i}) = (\alpha_{AllLayer,i-1} (D_{A,i-1} + D_{B,i-1} + \dots + D_{N,i-1}))_{mc(mv_{AllLayer,i-1})} + (\hat{D}_{A,i} + \hat{D}_{B,i} + \dots + \hat{D}_{N,i}) \quad (4.15)$$

This can be further simplified as:

$$D_{AllLayer,i} = (\alpha_{AllLayer,i-1} D_{AllLayer,i-1})_{mc(mv_{AllLayer,i-1})} + \hat{D}_{AllLayer,i} \quad (4.16)$$

, where

$$D_{AllLayer,i} = (D_{A,i} + D_{B,i} + \dots + D_{N,i}) \quad (4.17)$$

and

$$\hat{D}_{AllLayer,i} = (\hat{D}_{A,i} + \hat{D}_{B,i} + \dots + \hat{D}_{N,i}) \quad (4.18)$$

More precisely, for the latest enhancement layer N only the first β_N bit planes are combined with the information in other layers. In the above equation we have not shown this detail for simplicity. **Figure 4.7** shows this simplified SRFGS decoder. All the enhancement layers decoding loops are merged into a single loop. The entropy and bit plane decoding modules receive and decode the bitstreams for each layer, and merge them, except the bit plane after β_N in layer N , into one transform coefficient. These merged transform coefficients in each block are inversely transformed to the spatial domain. Since the IDCT is a linear process, merging the transform coefficient of each layer before the IDCT leads to identical results when the ordered is reversed. In this way, we only need one IDCT for all enhancement layers. The resultant spatial domain image is summed up with the attenuated prediction image of all enhancement layers to generate the reconstructed signal of all layers. The output signal is the sum of the base

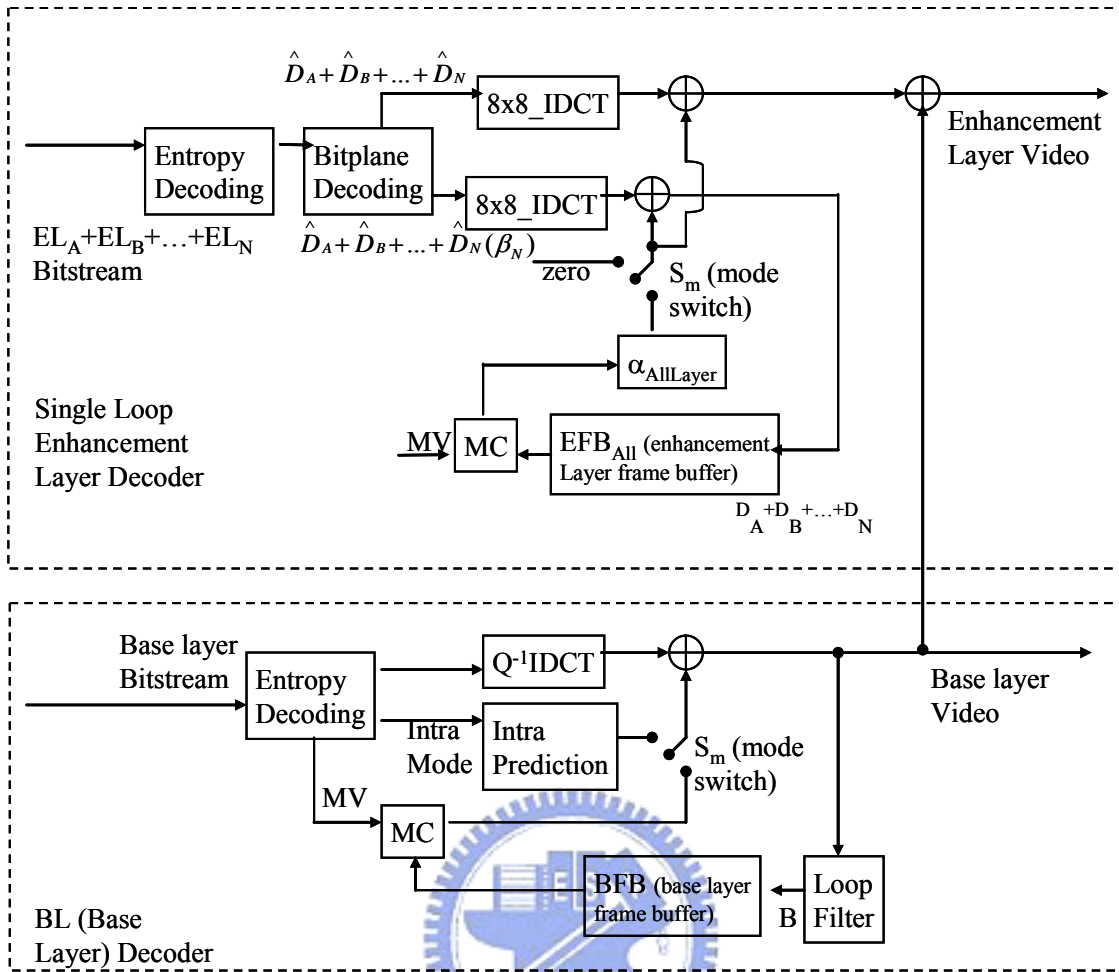


Figure 4.7 Diagram of the SRFGS single-loop enhancement layer decoder framework

layer reconstructed signal and the entire enhancement layer reconstructed signal.

Obviously, the single-loop enhancement layer decoder significantly reduces the decoder complexity with the disadvantage of losing the flexibility to adjust α at each layer. When combined with the macroblock-based alpha adaptation, the collocated macroblock at different layers need to use the same alpha, which may be 0 or **frame_** α . Except the restriction of the alpha selection, single-loop enhancement layer is identical to the original SRFGS decoder, and the error in each layer is still localized within its own layer although all the layers are merged.

4.5 Experiment Results and Analyses

The coding efficiency of the SRFGS is compared with RFGS, H.264/AVC and the H.264/AVC SVC [2][3]. The test conditions adopt the test 1c of [20] specified by the MPEG Scalable Video Coding Ad Hoc Group. The R-D curves of sequences including Tempete, Bus and Container in CIF resolution and YCbCr 4:2:0 format are compared at four bitrates/frame-rates. The frame rate is measured in frames per second. The four bitrates cover 128kbps/15fps, 256kbps/15fps, 512kbps/30fps, and 1024kbps/30fps. The coding performance of AVC are presented in [26] using the JM42 test model [27], where RD-optimized and CABAC modules are enabled. Quarter-pixel motion vector accuracy is employed with search range of 32 pixels. Four reference frames are used. Only one I-frame is used at the beginning. The P-period is 3 in both 15fps and 30 fps. For the H.264/AVC SVC (denoted as “SVC” in the following), the reference software version JSVM_4_6 is used in the simulation. The GOP size is 4 for the Bus sequence, and is 8 for the Tempete and Container sequences. Hierarchical-B GOP structure [4], RD-optimized mode decision, and arithmetic coding are used in the simulation. The bitstream extraction has utilized the quality layer proposed in [28]. The reference frame number is one for the P-frame and is two for B-frame.

Table 4.1 **The value of (α, β) used in the simulation.**

The value of beta is the number of referenced bits.

(α, β)	Tempete	Bus	Container
Stack 0	(0.7500, 24320)	(0.9375, 17067)	(0.7500, 24320)
Stack 1	(0.7500, 78000)	(0.9375, 51200)	(0.7500, 58860)
Stack 2	N/A	N/A	(0.7500, 92160)

For RFGS and SRFGS, the base layer is JM42. The test conditions are identical to that used in AVC except that we have disabled RD-optimized and adopted only one reference frame. At 30 fps, the P-period is 6 for Tempete and Container sequences. The P-period is 4 for Bus sequence. At 15 fps, the P-period is half. The bit plane and entropy coding are as the same as that for the MPEG-4 FGS [6]. In SRFGS, 2 enhancement layer loops are used for Tempete and Bus sequences and 3 enhancement layer loops are used for Container sequence. The detailed α and β used in the simulation is shown in **Table 4.1**. Note that regarding to the value of β , we use the number of referenced bits instead of the number of referenced bit planes. A simple frame-level bit allocation with a truncation module is used in the streaming server. For various target bitrate, different bit allocation between P and B frames are test and the one lead to best RD-performance is used to get the final results. This bit allocation analysis is reasonable because it can be done once accompany with the bitstream encoding, and provide best bit allocation at various operating bitrate during the streaming services.

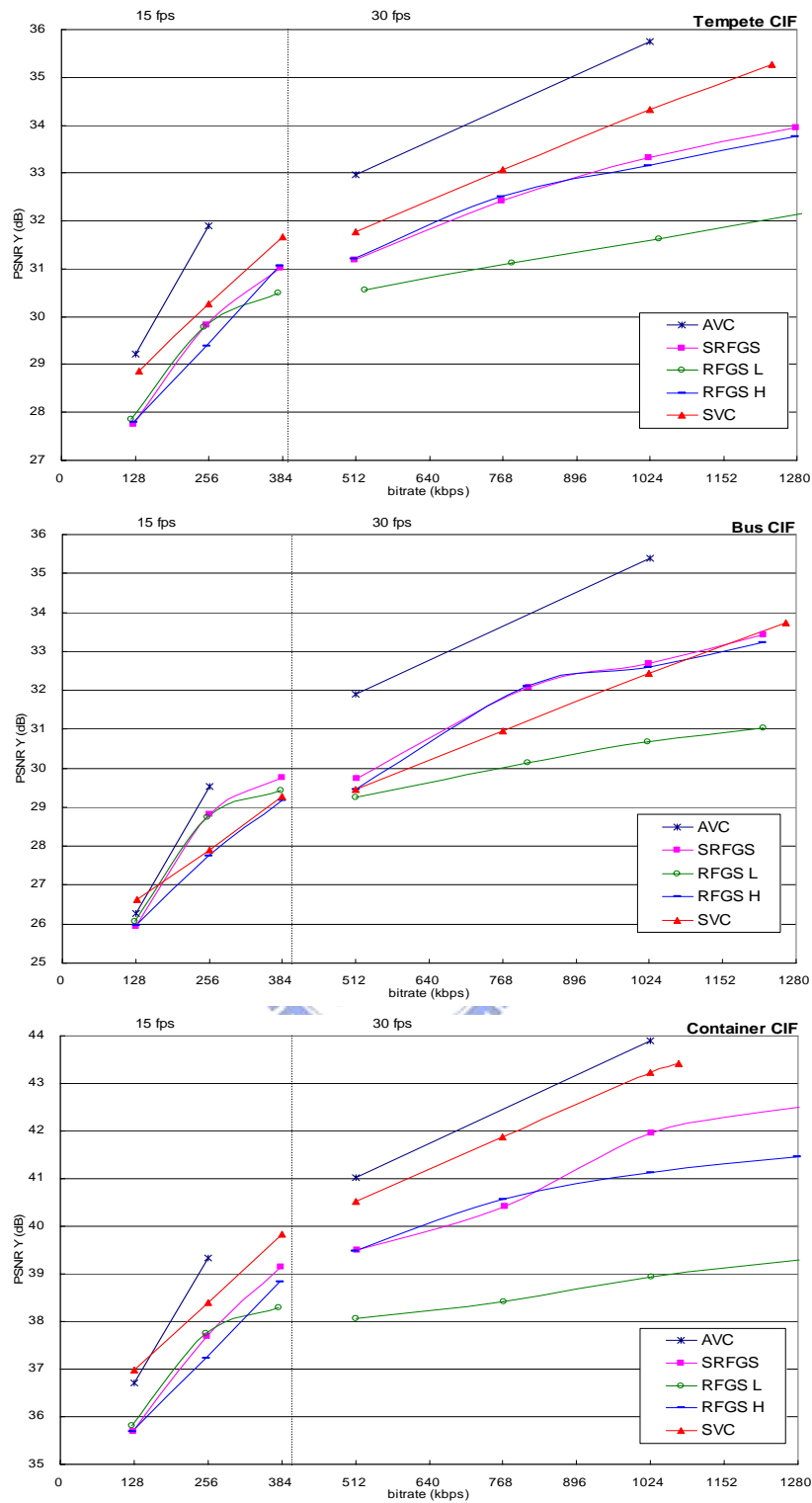
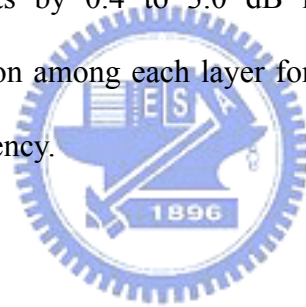


Figure 4.8 PSNR versus bitrate comparison between SRFGS, RFGS and AVC coding schemes for the Y component.

The simulation results are shown in **Figure 4.8**. Two RFGS results are shown, one has lower reference bitrate (labeled as RFGS_L) and the others have higher reference bitrate (labeled as RFGS_H). The SRFGS has similar performance with RFGS_L at low bitrate, and has improvements by 1.7 to 3.0 dB in PSNR at high bitrates. Since the SRFGS can remove more temporal redundancy at high bitrate than RFGS_L. As compared with RFGS_H, the quality of SRFGS is increased by 0.4 to 1.0 dB in PSNR at low bitrate because there is more drift error of RFGS_H at low bitrate. At high bitrate, the SRFGS increases 0.8 dB in PSNR at low motion sequence such as Container and has similar performance at high motion sequence, such as Tempete and Bus. For the high motion sequence there is less temporal correlation so the performance of the improved prediction technique in SRFGS decreases. At medium bitrate, SRFGS has at most 0.15 dB PSNR losses than RFGS_H. This comes from the fact that the increased dynamic range and sign bits of each layer in SRFGS slightly lower the coding efficiency. The simulation results show that RFGS can only be optimized at one operating point and SRFGS can be optimized at several operating points, which can serve wider bandwidth with superior performance. Compared to AVC, SRFGS has 0.4 to 1.5 dB PSNR loss at base layer. This is mainly because the MV in SRFGS is derived from both the base and enhancement layer information as described in Section 4.4.1. There is 0.7 to 2.0 dB PSNR loss at low bitrates and 2.0 to 2.7 dB loss at high bitrates. Compared with SVC, SRFGS has up to 1.5 dB PSNR loss at Tempete and Container sequences, but has 0.9 dB PSNR improvement at Bus sequence. Note that SVC has incorporated the hierarchical-B structure, the RD-optimized mode decision, and the arithmetic coding. These tools can also be integrated in the SRFGS structure to improve the performance.

4.6 Summary

In this chapter, we proposed a novel FGS coding technique named SRFGS. Based on RFGS, the SRFGS generalizes its prediction concept and structure to a multi-layer stack architecture. In each layer, the information to be coded is temporally predicted by the information of the previous time instance at the same layer. The stack concept allows the SRFGS to optimize at several operating points for various applications. With the bit plane coding and leaky prediction used in RFGS, SRFGS maintains the feature of fine granularity and error robustness. An optimized MB-based alpha adaptation is proposed to improve the coding efficiency. We also propose single-loop enhancement layer decoding scheme to reduce the decoder complexity. The simulation results show that SRFGS has improvements by 0.4 to 3.0 dB in PSNR over RFGS. Further investigation of the bit allocation among each layer for various types of video content can provide better coding efficiency.



CHAPTER 5

Relevance to the H.264/AVC SVC

5.1 Introduction

In the working of this thesis, the SVC standard is migrating from MPEG-4 FGS to the developing H.264/AVC SVC [2][3]. Many technologies developed based on the MPEG-4 FGS, including RFGS and SRFGS, have been adopted in the H.264/AVC SVC. In this chapter, we describe the application scenarios of the RFGS and SRFGS techniques in the H.264/AVC SVC.

5.2 RFGS in H.264/AVC SVC

In the H.264/AVC SVC, the non-anchor picture is inter-predicted with the hierarchical-B structure. Both the base and enhancement layer information is used in the prediction to improve the coding efficiency. The drifting error is limited because in the hierarchical structure, the length of the prediction path is reduced to the number of the layers. However, instead of using the hierarchical-B structure, the anchor picture is predicted by the previous anchor picture. Such that, the same problem in MPEG-4 FGS occur in the anchor pictures of the H.264/AVC SVC: prediction with the enhancement layer information improves the prediction efficiency, but also causes drifting error when the enhancement layer is truncated.

To solve this problem, the Adaptive Reference FGS (ARFGS) [29][30] is adopted

in the H.264/AVC SVC. Basically, ARFGS separate the prediction method into two categories. When the base layer coefficient is equal to zero, the prediction structure is identical with the RFGS prediction structure. When the base layer coefficient is non-zero, only the base layer information is used in the prediction, just like the MPEG-4 FGS. In the following, we describe the ARFGS prediction structure in detail and shown that ARFGS is basically the same with RFGS.

When there is no transform coefficient coded in the base layer, the ARFGS reference signal R_i is formed as:

$$R_i = (1 - \alpha)B_i + (\alpha E_{i-1})_{mc} \quad (5.1)$$

, where B_i is current base layer signal, α is the leaky factor, and E_{i-1} is the enhanced reference signal in the previous time instance. E_{i-1} is the sum of the base layer and enhancement layer information in the previous time instance:

$$E_{i-1} = B_{i-1} + D_{i-1} \quad (5.2)$$

, where D_{i-1} is enhancement layer information in the previous time instance. Because there is no coefficient coded in the base layer, B_i equals to the base layer signal in the previous time instance B_{i-1} .

$$B_i = (B_{i-1})_{mc} + \hat{B}_i = (B_{i-1})_{mc} \quad (5.3)$$

, where $(B_{i-1})_{mc}$ denotes the motion compensated version of B_{i-1} . \hat{B}_i is the coded coefficient in base layer, which is equal to zero. Equation (5.1) becomes

$$\begin{aligned} R_i &= B_i - (\alpha B_{i-1})_{mc} + \alpha (B_{i-1} + D_{i-1})_{mc} \\ &= B_i + (\alpha D_{i-1})_{mc} \end{aligned} \quad (5.4)$$

Such that, the residue signal coded in the enhancement layer \hat{D}_i can be generated with:

$$\begin{aligned}\hat{D}_i &= F_i - R_i \\ &= F_i - (B_i + \alpha D_{i-1mc})\end{aligned}\quad (5.5)$$

, where F_i is the original signal. It is obviously that the equation (5.5) is identical with equation (3.9) and (3.10).

When there are transform coefficients coded in the base layer, ARFGS gets the reference signal at the transform domain. For each of the transform block, if the collocated coded coefficient in the base layer \hat{B}_i is equal to zero, equation (5.1) is applied to generate the enhancement layer reference coefficient (in transform domain).

If the collocated coded coefficient in the base layer \hat{B}_i is non-zero, the enhancement layer reference coefficient is set to be the same with the base layer coefficient. This is the same with the MPEG-4 FGS approach and the RFGS structure with leaky factor equal to zero. After set all the enhancement layer reference coefficients in the handling block, the reference block is transform back to the spatial domain to derive the enhancement layer prediction residue \hat{D}_i .

With the above analysis, we can conclude that ARFGS can be viewed as an extension of RFGS, which adaptive selects the leaky factor at coefficient level according to the base layer transform coefficients. As shown in [30], with the test conditions specified in the core experiment [32], ARFGS has more than 4 dB PSNR improvement comparing with the structure that not using the enhancement layer information as reference.

5.3 SRFGS in H.264/AVC SVC

As an extension of the RFGS structure, the ARFGS proposed in [29][30] has the

same problem in RFGS: more enhancement layer information improves the prediction efficiency, but also causes more drifting error when it is truncated. To solve this problem, we proposed the multi-loop stack structures in Chapter 4. Similarly, ARFGS also incorporates the multi-loop stack structure, as proposed in [31].

As discussed in Chapter 4, to generate the SRFGS enhancement layer coded residue at layer Y and time instance i , $\hat{D}_{Y,i}$, the following equation is used:

$$\hat{D}_{Y,i} = F_i - B_i - D_{0,i} - D_{1,i} - \dots - D_{Y-1,i} - (\alpha_{Y,i-1} D_{Y,i-1})_{mc(mv_{Y,i-1})} \quad (5.6)$$

where the signal F_i is the original signal. B_i is the base layer reconstructed signal. $D_{X,i}$ is the enhancement layer reconstructed signal at layer X . $D_{Y,i-1}$ is the enhancement layer reconstructed signal of layer Y at previous time instance. $(D_{Y,i-1})_{mc(mv_{Y,i-1})}$ is the motion compensated version of the signal $D_{Y,i-1}$ using the motion vector $(mv_{Y,i-1})$. $(D_{Y,i-1})_{mc(mv_{Y,i-1})}$ is multiplied with the leaky factor $\alpha_{Y,i-1}$ to reduce the drifting error. Equation (5.6) shows that, to generate the coded residue $\hat{D}_{Y,i}$, all the base and enhancement layer information are removed to maximize the prediction efficiency.

From equation (4.13), the signal $D_{X,i}$ is the sum of the enhancement layer information at previous time instance and the residue coded in current time instance:

$$D_{X,i} = (\alpha_{X,i-1} D_{X,i-1})_{mc(mv_{X,i-1})} + \hat{D}_{X,i} \quad (5.7)$$

The leaky factor $\alpha_{X,i-1}$ controls the usage of the enhancement layer information at previous time instance. The larger the α the more the enhancement layer information is used in the prediction. When α equals to zero, there is no previous enhancement layer information used in the prediction. Zero α decreases the prediction efficiency, but also reduces the complexity because the motion compensation of this layer need not to be

invoked.

This idea is used in the ARFGS stack structure. In ARFGS, the stack structure is basically the same with the SRFGS. To reduce the complexity of multiple loops motion compensation at the decoder side, ARFGS slightly modified the way to generate the enhancement layer coded residue. Specifically, in ARFGS, the equation (5.6) is modified as following:

$$\hat{D}_{Y,i} = F_i - B_i - \hat{D}_{0,i} - \hat{D}_{1,i} - \dots - \hat{D}_{Y-1,i} - (\alpha_{Y,i-1} D_{Y,i-1})_{mc(mv_{Y,i-1})} \quad (5.8)$$

That is, instead of removing $D_{X,i}$ from the original signal, only the signal $\hat{D}_{X,i}$ is removed and the term $(\alpha_{X,i-1} D_{X,i-1})_{mc(mv_{X,i-1})}$ is not considered. With the lack of removing $(\alpha_{X,i-1} D_{X,i-1})_{mc(mv_{X,i-1})}$, ARFGS decoder generate the reconstructed signal with following:

$$ReconstructedSignal = B_i + \hat{D}_{0,i} + \hat{D}_{1,i} + \dots + \hat{D}_{Y-1,i} + \hat{D}_{Y,i} + (\alpha_{Y,i-1} D_{Y,i-1})_{mc(mv_{Y,i-1})} \quad (5.9)$$

Which means at decoder side, ARFGS only needs to perform motion compensation at the base layer (to generate the term B_i) and the highest enhancement layer (to generate the term $(\alpha_{Y,i-1} D_{Y,i-1})_{mc(mv_{Y,i-1})}$). The drawback is the decreasing of the coding efficiency. Note that at encoder side, multi-loops motion compensation is still required to generate the signal $\hat{D}_{X,i}$ of each layer.

As shown in [31], the SRFGS stack structure (equation (5.6)) has more than 2dB PSNR improvement compare with the non-stack structures that proposed in [29][30]. And the ARFGS stack structure (equation (5.8)) has more than 0.5dB PSNR loss comparing with the SRFGS stack structure (equation (5.6)).

5.4 Summary

In this chapter, we describe the applications of RFGS and SRFGS in the developing H.264/AVC SVC. In the H.264/AVC SVC, the RFGS prediction structure is adopted and extended to adapt the leaky factor at coefficient level. The SRFGS prediction structure is also adopted with some modification to reduce the decoder complexity. The simulation results proposed in [29]-[31] show that the RFGS and SRFGS prediction structure have up to 4dB and 2dB PSNR improvement in the H.264/AVC SVC, respectively.



CHAPTER 6

Robust Scalable Video Coding

6.1 Introduction

In the previous chapters, we propose RFGS and SRFGS to improve the prediction efficiency and the robustness simultaneously for MPEG-4 FGS. RFGS uses the enhancement layer information to improve the prediction efficiency, and uses leaky prediction to reduce the drifting error when enhancement layer is truncated. SRFGS extends the RFGS prediction scheme into a multi-loop stack structure. The stack structure improves the RFGS prediction efficiency and robustness. The stack structure allows close loop at several operation point and extends the application bitrate range.

Both of the RFGS and SRFGS techniques focus on the SNR scalability. In this chapter, based on the proposed leaky prediction and stack structure, we extend our work to support spatio-temporal and SNR scalability simultaneously, which we named as Robust Scalable Video Coding (RSVC). Except the extension on spatial and temporal scalability, in RSVC, we also improve the VLC-based FGS entropy coding that used in RFGS and SRFGS into arithmetic coding based FGS entropy coding.

The rest of this chapter will be organized as follows. In Section 6.2, we give an overview of the RSVC system structure. From Section 6.3 to 6.5, we describe spatial and SNR scalability, FGS, and temporal scalability, respectively. In Section 6.6, we describe the bitstream extraction and related error concealment method. The simulation

results are shown in Section 6.7. The comparison with H.264/AVC and the H.264/AVC SVC are given. Finally, the summary is given in Section 6.8.

6.2 The RSVC System Architecture

To support spatio-temporal scalability, in RSVC, we simply extend the stack structure proposed in SRFGS to support spatial scalability. The hierarchical-B structure provided in H.264/AVC is used to support temporal scalability.

The stack structure proposed in SRFGS is used to support multiple SNR layers. Although we only focused on FGS previously, the identical stack structure can be used to achieve CGS with coding the DCT coefficients in a non-embedded way. To support spatial scalability, an interpolator/decimator is inserted between layers (stacks) when the two layers (stacks) are coded in different spatial resolutions. To further improve the prediction efficiency, we extend the inter-layer prediction structure of SRFGS into an adaptive mean. More details of inter-layer prediction will be described in Section 6.3.

Figure 6.1 shows the encoder structure of RSVC. Three spatial or SNR layers are shown in the figures. Based on the stack concept in SRFGS, each spatial or SNR layer has almost identical structure. The enhancement layer information is used at higher layer to increase the spatial resolution and/or improve the prediction efficiency. To remove the inter-layer redundancy, the information coded at the lower layer, including texture, prediction information, and residue, can be used to predict the information at higher layer. In the temporal prediction, the hierarchical prediction structure, which will be detailed in section 6.5, is used to support temporal scalability. Leaky prediction is adopted in the enhancement layer to reduce the drifting error.

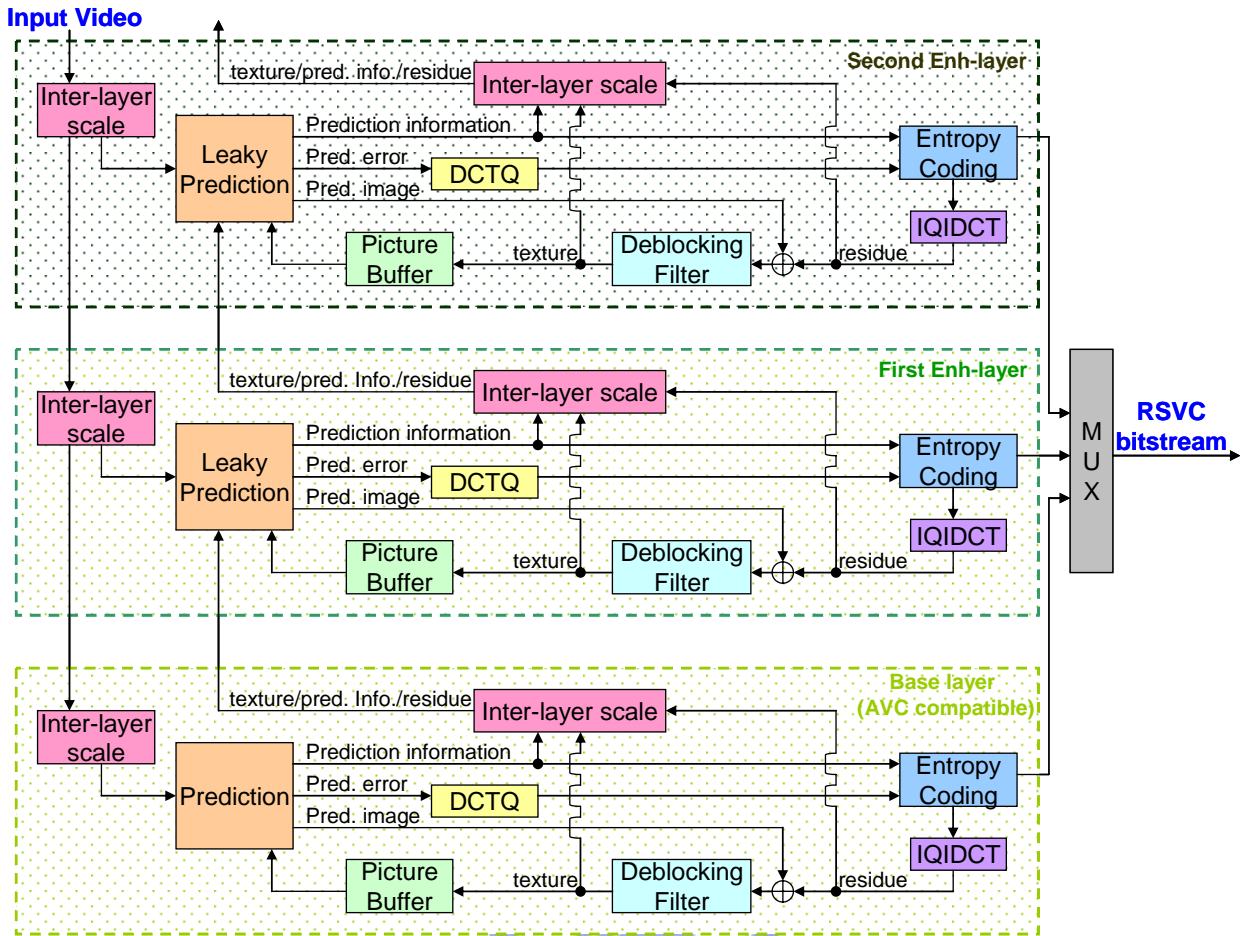


Figure 6.1. RSVC encoder structure with three spatial/SNR layers

We should mention that, the H.264/AVC SVC, which is developed after the proposing of the stack structure in SRFGS, also adopts very similar stack structure to support spatial and SNR scalability, as shown in Figure 2.2. However, in RSVC, we provide better inter-layer prediction with less constraint on the inter-layer prediction and with less coding overhead due to adaptive inter-layer prediction. We also implement the temporal scalability is a more efficient way to reduce the memory requirement at the decoder side. Further, when supporting FGS, RSVC can use stack structure to improve the prediction efficiency, while H.264/AVC SVC has not support stack structure for FGS yet.

6.3 Spatial Scalability and SNR scalability

As shown in Figure 6.1, RSVC supports spatial and SNR scalability with the stack structure. To improve the prediction efficiency, the information coded in the lower layer can be used in the higher layer to remove the inter-layer redundancy. The process that is used to remove the inter-layer redundancy is usually called inter-layer prediction.

Inter-layer prediction is extensively used in RFGS and SRFGS, but in a non-adaptive way. In SRFGS, only the quantization error of previous layer is coded in the current layer. Such that the reconstructed texture of previous layer is always removed. Further, in RFGS, the residue at base layer is always removed from the enhancement layer residue. Moreover, in RFGS and SRFGS, the prediction information of the base layer is always reused in the enhancement layers.

In RSVC, we make such inter-layer prediction all adaptive. The encoder can adaptively remove the texture or residue in the reference layer. In each layer, the prediction information can be coded separately, or be predicted from the reference layer. Due to the adaptive inter-layer prediction, many overheads need to be coded in the bitstream to signal whether the prediction is used or not. We carefully design the mode adaptation structure to reduce the overheads, which is especially important at low bitrate.

In the following sections, we describe three inter-layer prediction techniques in RSVC, including texture prediction, prediction-information prediction, and residue prediction. We also compare our methods with the related techniques in H.264/AVC SVC to show the advantage of RSVC.

6.3.1 Texture Prediction

Texture prediction gets the prediction from the reconstructed image of the reference layer. Instead of coding one more mode for adaptive texture prediction, we

simply change the meaning of the “DC prediction” in the intra 16x16 MB mode to reduce the overhead. The predictor of DC prediction is formed by averaging the value of the pixels at the boundary of the upper and left MBs, this single averaged value is then used to predict the entire MB. On the contrary, the predictor of the texture prediction is the reconstructed image at the reference layer, which usually provides better prediction because difference pixels could have different value. Such that, in RSVC, instead of coding one more mode as in H.264/AVC SVC, we simply use the DC prediction of intra 16x16 MB mode to indicate that the texture prediction is used or not.

When texture prediction is used, the decoder needs to decode the reference layer to get the reconstructed image. When the reference pixel is in inter mode, motion compensation need to be invoked in the reference layer. For a complexity/power constraint decoder, it is better to find a way to reduce the complexity raised by motion compensation. In H.264/AVC SVC, texture prediction is only allowed to the MB that has intra mode in the reference layer. And in the reference layer, the intra MB needs to get its predictor only from other intra MBs. In this way, to get the predictor of an MB, only the motion compensation at current layer need to be invoked. This feature is also referred to as “single loop decoding”. In RSVC, the same idea is adopted. However, instead of always enabling the single loop decoding, we make it configurable. For the applications that most of the clients do not have complexity/power constraint, RSVC can also support texture prediction without any restriction.

6.3.2 Prediction-Information Prediction

In RSVC, each spatial or SNR layer can have its own prediction information, including the MB mode, reference index and motion vector in inter MB, or the intra prediction mode in intra MB. RSVC can also reuse such prediction information in the reference layer, such that no prediction information needs to be coded in the current

layer. The prediction residue, which could be further predicted by the residue prediction in Section 6.3.3, is coded in the bitstream.

In H.264/AVC SVC, only the motion information of the inter MB is used for inter-layer prediction. The redundancy of intra prediction mode is not explored. Further, for spatial scalability, H.264/AVC SVC also supports a mode named “quarter-pixel refinement”. This mode gets the motion information from the reference layer, and can further refine the MV of the reference layer by a value ranges between -1 to 1. In RSVC, this mode is simply not adopted. The reason is the refinement MV needs to be transmitted not only once for the entire MB, but multiple times when there are many blocks in a MB. Therefore this mode is not that efficient. To achieve similar functionality, the separate coded MVs in the current layer should be enough. With eliminating this mode, we save the overhead to coding it and also reduce the complexity at mode decision.

6.3.3 Residue Prediction

Residue prediction subtracts the residue at the reference layer from the residue at the current layer. In RSVC, residue prediction can be adaptively enabled in any condition. In H.264/AVC SVC, residue prediction is only allowed when $cbp!=0$, which means there must have coefficients coded in the handling MB. Without such constraint, RSVC allows more optimization in the mode decision.

6.3.4 Skip Mode

In H.264/AVC, skip mode means using direct mode to derive motion vector, and there is no residue to be coded in the handling MB. In H.264/AVC SVC, the same meaning of skip mode is still hold. Such that it can not have skip mode when using inter-layer prediction. In RSVC, we allow more combination between the inter-layer prediction and the skip mode. For example, we can have skip mode with texture

prediction, or have skip mode with one of, or both of, the prediction-information prediction and the residue prediction. With more prediction modes are allowed to achieve skip mode, the prediction efficiency is improved, especially at low bitrate.

6.4 Fine Granularity Scalability (FGS)

In RSVC, there are two types of SNR scalability, coarse granularity scalability (CGS) and fine granularity scalability (FGS). CGS and FGS use identical prediction structure with spatial scalability, as described in Section 6.3. The difference is at the entropy coding. CGS directly uses the H.264/AVC CABAC entropy coding to provide non-embedded bitstream. However, FGS codes the DCT coefficients in an embedded way, such that the bitstream can be truncated at any position. To provide an embedded bitstream, the RSVC FGS extend the H.264/AVC CABAC to support bitplane coding. Further, leaky prediction will be enabled in FGS because bitstream could be truncated. In this section, we described the entropy coding and leaky prediction in the proposed FGS.

6.4.1 Entropy Coding

To support FGS, the DCT coefficients are coded in a bitplane fashion. In RSVC, we simply extend the H.264/AVC CABAC to support bitplane coding. The coding process is starting from the most significant bitplane, and is continued until reaching the least significant bitplane. In each bitplane, raster scan is used among the MBs. In each MB, zigzag scan is applied among coefficients with different frequencies. In each DCT block, coded_block_flag indicate there is coefficient become significant in this bitplane. The context of coded_block_flag comes from the same flag at neighboring blocks of the current bitplane.

The bitplanes of each coefficient are separated into significant bit and refinement bit. Different probability states are used for these two types. The significant bit is sent

when coded_block_flag equal to 1. The significant bits at different frequencies use different probability states. The probability state of the refinement bit comes from the Laplacian model. This is because the probability distribution of the residue value, which has larger probability at smaller value, can be approximate by the Laplacian model, as shown in Figure 6.2. Similar idea is also considered in the MPEG-4 FGS [6][10] and CABIC [36]. In MPEG-4 FGS, the Laplacian model is used to reconstruct the coefficients that are truncated during transmission. In CABIC, the Laplacian model is use to determine the probability state of the refinement bit, just like RSVC. To code the significant and refinement bit, the probability state of the less significant bitplane is coherent from the probability state at more significant bitplane. The reason is that the probability statistic among bitplanes should change gradually from more significant bitplane to less significant bitplane. Using same probability state among bitplanes can reserve the probability statistic of the current coding material and also reduce the memory requirement to store the probability states. It should be mentioned that although the significant and refinement bits are coded with different probability states, they are coded in the same zigzag scanning pass, which reduces the complexity comparing with coding the significant and refinement bits in two passes.

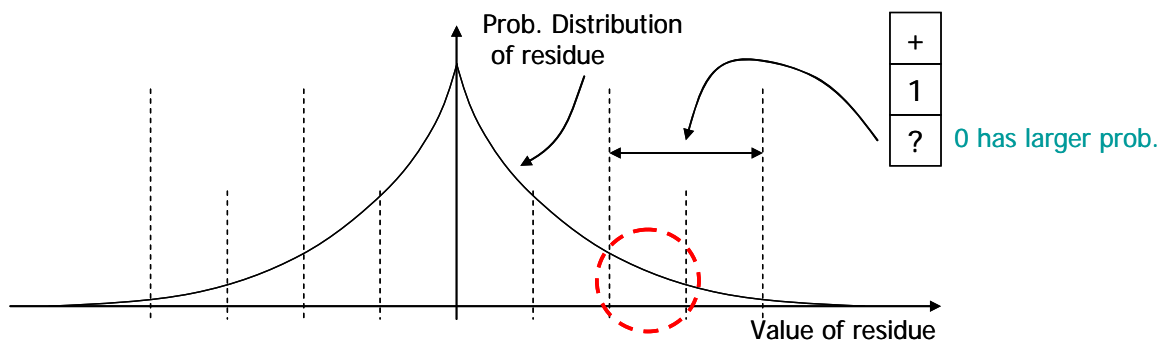


Figure 6.2. Probability distribution of the residue value can be approximate by a Laplacian model, where smaller value has larger probability.

In bitplane coding, the coding is start from the global maximal bitplane of the entire picture. The blocks that have small coefficient value might be visited several times before it becomes significant. Each time the blocks are visited, the coded_block_flag need to be sent with value equal to zero. To reduce such overhead, we utilize three methods. Firstly, the coded_block_patten in H.264/AVC CABAC is used to indicate are there non-zero coefficient in the handling blocks. Further, we categorize the coefficients into 6 types, which come from 3 color components (YUV) and 2 frequency bands (DC/AC). Each category has its own maximal bitplane. During the bitplane coding process, a block only need to be visited when the maximal bitplane of its own category is reached. Thirdly, when the coded_block_flags of the luma ac component in a MB are all zero, we group them into one flag “luma_ac_msb_not_reach_flag”. The coded_block_flag will be sent only when luma_ac_msb_not_reach_flag equals to 0. The luma_ac_msb_not_reach_flag is context by the same flag at neighboring MBs of the current bitplane.

Bitplane coding is done in the picture basis. However, during the RD-optimized mode decision of a MB, the (estimated) rate of that MB is required. Because the maximal bitplane of the entire picture is not obtained yet, we add several dummy

bitplanes on top of the MSB of the handling MB to simulate the missing bitplanes that appearing in the final bitplane coding. The number of dummy bitplanes depends on the image type, qp, and color components, and is ranging between 0 to 3, inclusively. After reaching the real MSB of the handling MB, the aforementioned bitplane coding is applied on the DCT coefficients to estimate the rate.

Except the coding of the residue, the coding of the prediction information, including the MB type, the motion information, and the intra prediction mode, is also a challenge problem in FGS. In RSVC, we support two methods to coding the prediction information. The first method sends the entire prediction information of a picture before sending any coefficient of that picture. This method is straightforward and has less complexity. Further, because the prediction information is usually more important than the prediction residue, sending the prediction information before any of the coefficients does order the data according to their importance. However, during the truncation, the prediction information is truncated by the raster scan order among MBs, such that it is possible that only the top half of the picture gets the correct prediction information. The bottom half prediction information need to be concealed from the reference layer, as described in Section 6.6.2.

The second method sends the prediction information of each MB just after the first significant coefficient of that MB is sent. In this way the prediction information of the MB that has smaller residue will be truncated first. This method is also used in the H.264/AVC SVC. However, this method is more complex and less efficient. Because the prediction information is sent in an arbitrary order, it is difficult to predict the current MB by its neighboring contexts. In our simulation, comparing with the first method, the second method leads to around 0.2 dB PSNR loss in coding efficiency.

6.4.2 Leaky Prediction

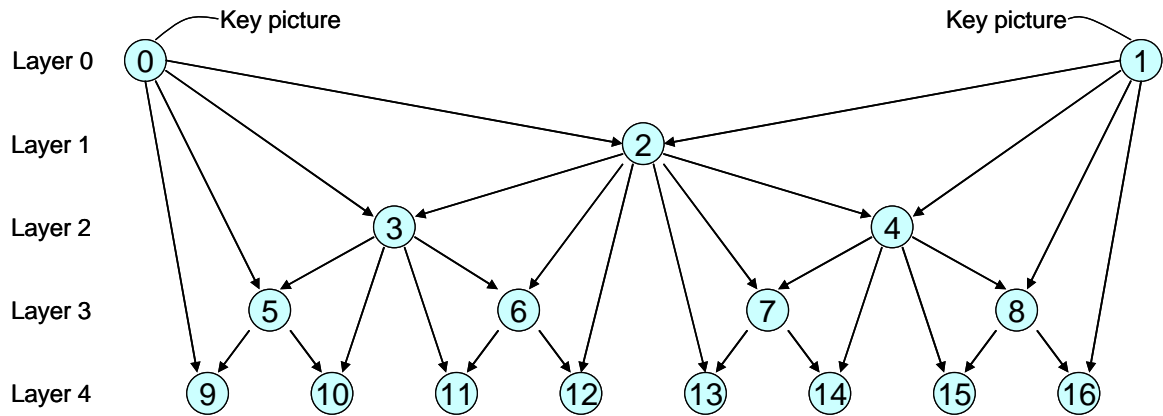
Similar with RFGS and SRFGS, we apply leaky prediction in RSVC to reduce the drifting error. The enhancement layer pixel will subtract from the reference layer pixel before multiply with the leaky factor. The reference layer pixel is then added back with such decayed enhancement layer information. In RSVC, intra prediction is also utilized and might cause drifting error. The leaky prediction is applied on both inter and intra MBs. Only the MBs use texture prediction needs not perform leaky prediction, because we assume the data at reference layer is already received when decoding the current layer.

6.5 Temporal Scalability

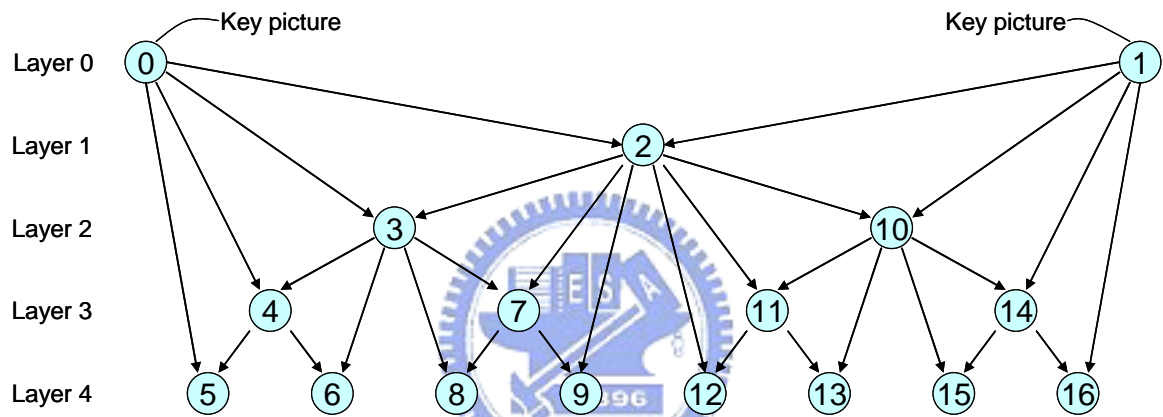
In the H.264/AVC based codec, temporal scalability can be easily supported by the hierarchical prediction structure, as described in Section 2.2.3 and in Figure 2.3 (b). However, the implementation of the hierarchical prediction structure significantly affects the Decoded Picture Buffer (DPB) requirement.

As shown in Figure 6.3, two implementations of the hierarchical prediction structure are discussed. Each circle represents a picture. The number in the circle denotes the coding order. The display order is from left to right. In the structure shown in (a), the pictures are coded in a layer-by-layer means. That is, the pictures at the lowest layer, which is layer 0, are coded first, followed by the entire pictures at layer 1, and then the entire pictures at layer 2, and so on. At the decoder side, assuming we want to decode the second picture in display order, which is the 9th picture in coding order; we need to decode all the previous 9 pictures before it and store them in the DPB. The DPB requirement for such scheme is equal to $N/2+1$, where N is the GOP size.

In the structure shown in (b), the pictures are coded with the coding order as close to the display order as possible. That is, we basically coded the picture according to



(a) Structure with picture coded with coding order layer-by-layer



(b) Structure with picture coded with coding order close to display order

Figure 6.3. Hierarchical prediction structure implementation

their display order, except the reference pictures need to be coded first. At the decoder side, assuming we want to decode the second picture in display order, which is the 5th picture in coding order; we only need to decode the previous 5 pictures before it and store them in the DPB. The DPB requirement for such scheme is reduced to $\log_2(N)+1$. When larger GOP size, the DPB size reduction by structure (b) is more significant. For example, with $N=64$, structure (a) needs DPB size equal to 65 pictures, while structure (b) only needs DPB size equal to 7 pictures. In the current implementation, RSVC utilizes structure (b), while H.264/AVC SVC utilizes structure (a).

It should be mentioned that although structure (b) significantly reduces the DPB requirement, it still provides same temporal scalability with structure (a). For example, to provide half of the frame rate, the decoder or streaming server can simply drop the pictures at layer 4 in both structures. To obtain the layer of a picture, the decoder or streaming server need to parse the picture header for both the two structures. It should be mentioned that for the structure (a), although the pictures are ordered by layers, the parsing process is still required. This is because without the parsing process, the decoder or streaming sever doesn't know where is the boundary between layers. The parsing process can be eliminated with adding an SEI (supplemental enhancement information) message before the picture to denote the temporal layer of that picture.

6.6 Bitstream Extraction and Error Concealment

In this section, we describe the RSVC bitstream extraction method and the error concealment at the decoder when the bitstream is truncated.

6.6.1 Bitstream Extraction

To achieve the target spatio-temporal resolution and request bitrate, the RSVC bitstream is extracted by the streaming server. The spatial layers and the pictures that exceed the request spatio-temporal resolution are firstly dropped. In the remaining bitstream, the SNR layers that totally exceed the requested bitrate will be dropped, too. If CGS entropy coding is used, the SNR layer that covers the requested bitrate is also removed. If FGS entropy coding is used, the bitstream can be extracted to provide exactly the requested bitrate. The bit will be allocated to the pictures at the lowest X temporal layers in an equal-percentage way. That is, assuming there are totally T_Y bits in the temporal layer Y, and the streaming server allocates E_Y bits to temporal layer Y. Equal-percentage means, $E_0/T_0 = E_1/T_1 = \dots = E_X/T_X$. This is because usually there are

more bits in the pictures at lower temporal layers. With allocating bits in an equally percentage way, the pictures at lower layers get more bits and thus provide better reference image quality. The value of X , which is the number of the lower layers that will be allocated bits, is obtained by multiple run. The X that provides best PSNR results is used. Basically, X increases when the requested bitrate increase.

6.6.2 Error Concealment

When the bitstream is truncated, we conceal the lost data at the decoder. The error concealment can be separated into two parts, the prediction information concealment and the coefficient concealment.

When the prediction information is lost, the prediction information in the reference layer is used for concealment. When the reference layer uses the texture prediction, most probably it can not get good prediction from the normal inter and intra prediction. In this case, we use texture prediction in the current layer to conceal the error. If texture prediction is not used in the reference layer, most probably the normal inter or intra prediction provides better prediction. We use both prediction-information prediction and residue prediction in the current layer for this case. The motion information or intra prediction mode of the reference layer is applied to the current layer to generate the predictor, and the residue at the reference layer is used to refine the prediction error.

When the coefficient is only partially refined, the Laplacian model that described in Section 6.4.1 is used to estimate the value of the remaining bitplanes.

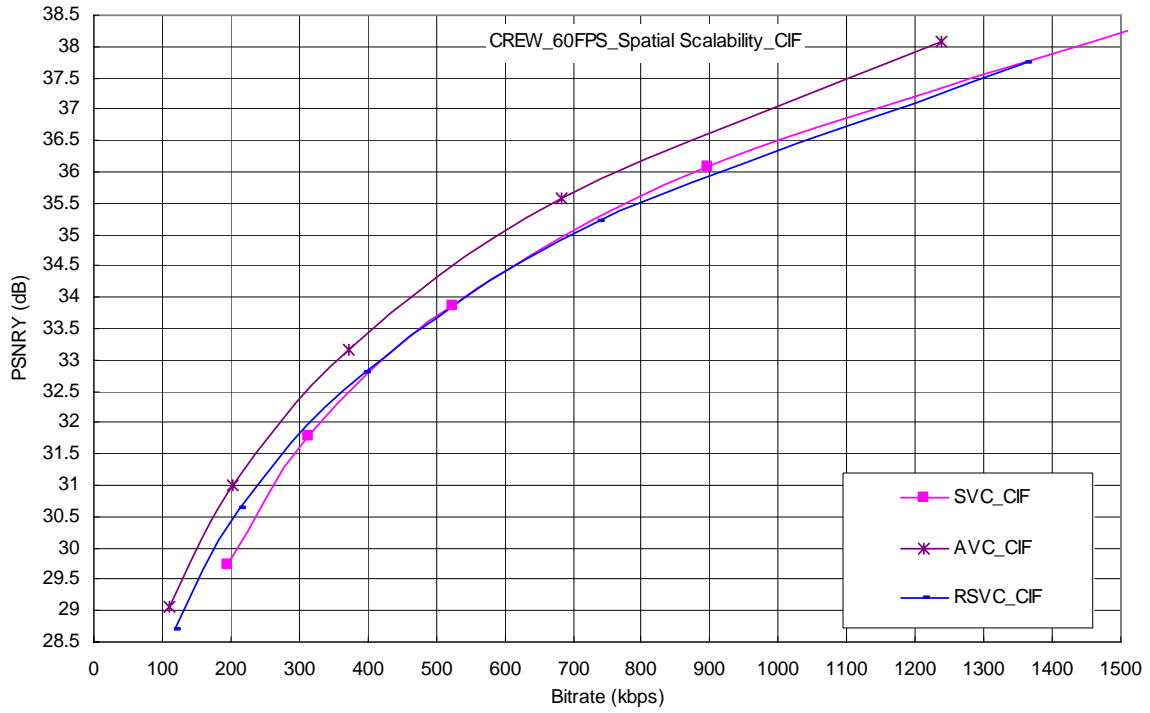
6.7 Simulation Results

In this section, we compare the RSVC simulation results with the H.264/AVC and the H.264/AVC SVC. We firstly show the simulation results for spatial scalability, followed by the results of SNR scalability. Finally the simulation results of combined

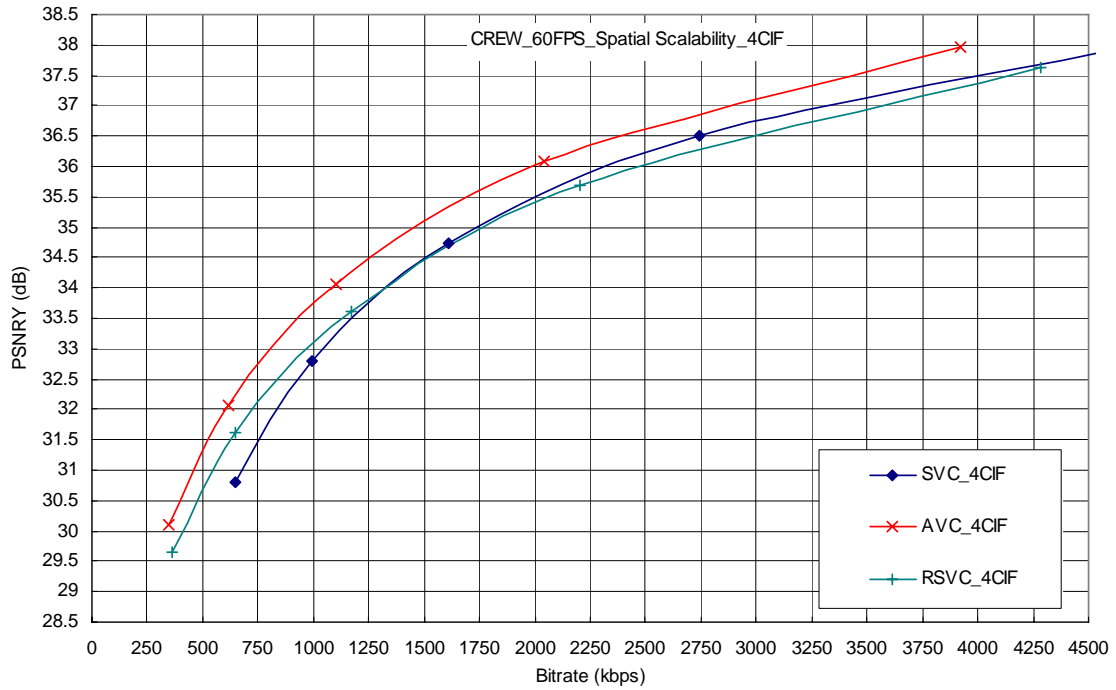
scalability are shown. Temporal scalability is not compared separately because all of the three codeds use hierarchical prediction structure and will lead to same performance.

In the simulation, the sequence “Crew” is used. According to different frame rate, the GOP sizes are 64@60fps, 32@30fps, and 16@15fps. The RD-optimized mode decision with all prediction modes are turned on. The CABAC is used as the entropy coding. The FREXT mode is not used, so only 4x4 transform is considered. For fair comparison, the single loop decoding feature is turned on in RSVC. The RSVC codec is implemented based on the H.264/AVC JM 10.1 reference software. The H.264/AVC SVC simulation results come from the JSVM 4.6 reference software.





(a) cif resolution



(b) 4cif resolution

Figure 6.4. Simulation results for spatial scalability.

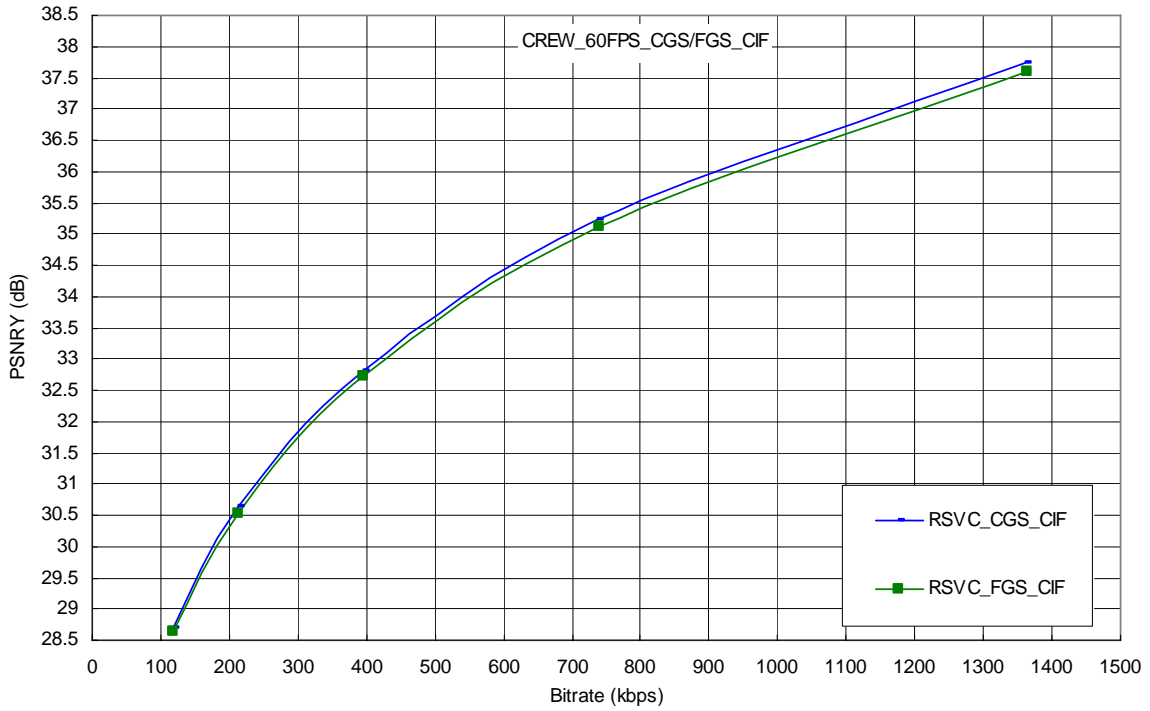
6.7.1 Spatial Scalability

Figure 6.4 shows the simulation results for spatial scalability. (a)/(b) are the results in Cif/4Cif resolution, respectively. The result at Qcif is not shown because the performance at the base resolution is similar. All the resolution is coded at 60fps. Comparing with H.264/AVC SVC, we can found that at low bitrate, RSVC provides 0.5 and 0.8 dB PSNR improvement at Cif and 4Cif resolution, respectively. At high bitrate, RSVC has around 0.2dB PSNR loss.

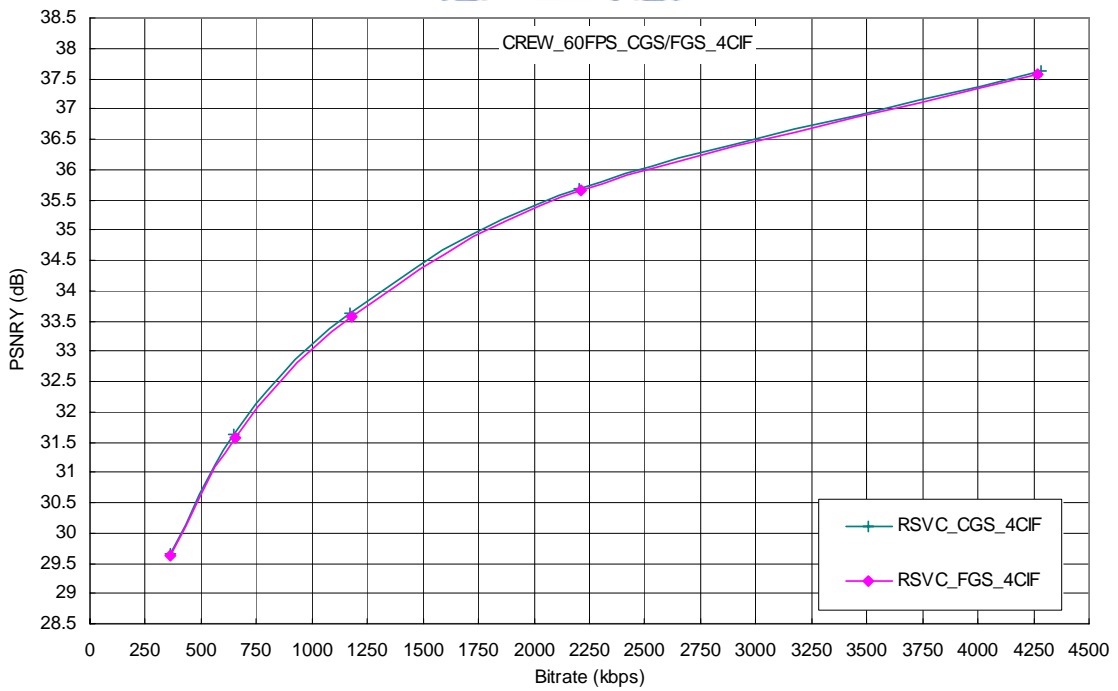
6.7.2 SNR Scalability

There are two results shown in this section. In Figure 6.5, we run the same simulation as shown in Figure 6.4 again, but change the entropy coding into the FGS mode. The prediction data is sent at the beginning of each picture. The leaky factor in FGS is set to 1. The result is compared with the entropy coding that using the CGS mode, which is already shown in Figure 6.4. The results show that comparing with the CGS coding that utilizes H.264/AVC CABAC, there is only 0.1dB PSNR loss in the proposed FGS coding.

In Figure 6.6, we compare the SNR scalability between RSVC and H.264/AVC SVC. The spatio-temporal resolution is coded at 4Cif@60fps. The RSVC is configured to have two enhancement stacks, while the H.264/AVC SVC does not support stack mode in FGS. Further, RSVC sends the prediction data at the beginning and has leaky factor equal to 0.875. The motion refinement feature in H.264/AVC SVC is turned on, but the quality layer tool is not used. This is because JSVM 4.6 does not support this combination. With the stack structure, RSVC provides up to 0.7dB gain over the H.264/AVC SVC. RSVC has around 0.2dB loss at the medium bitrate of each stack.



(b) cif resolution



(b) 4cif resolution

Figure 6.5. Simulation results for CGS and FGS entropy in RSVC.

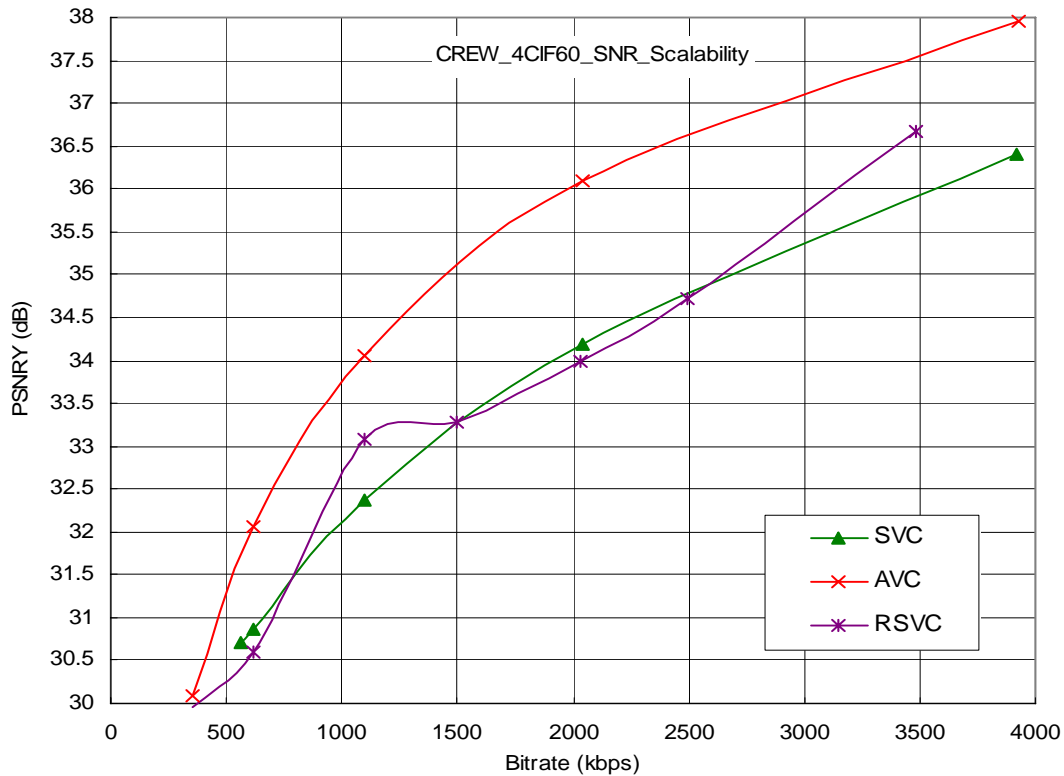
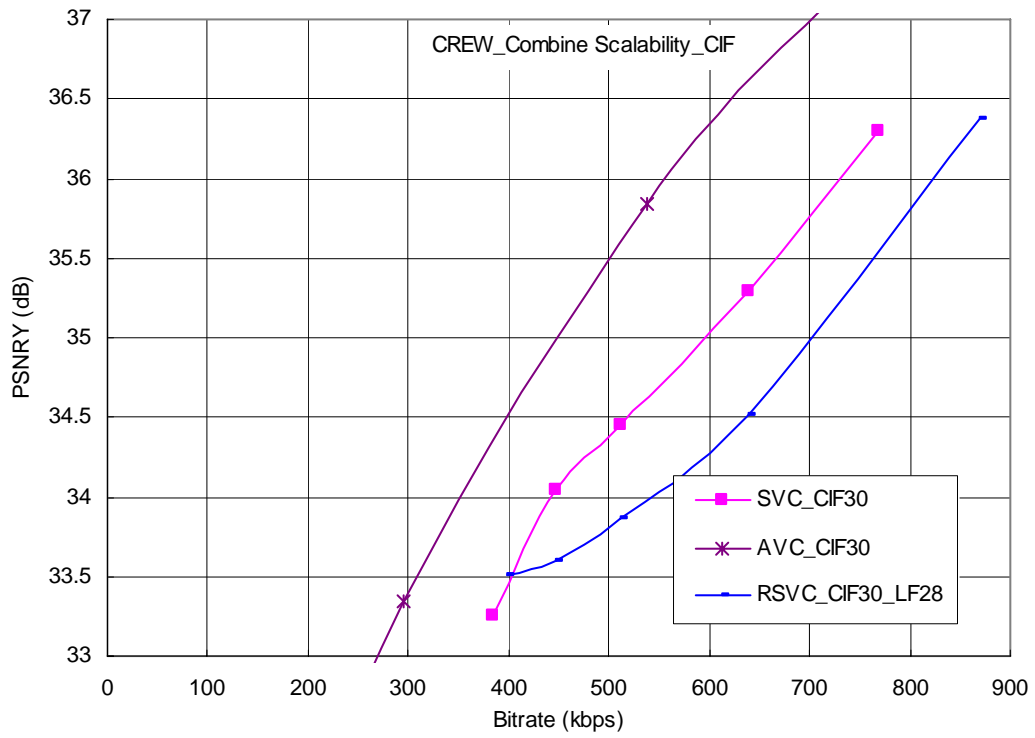


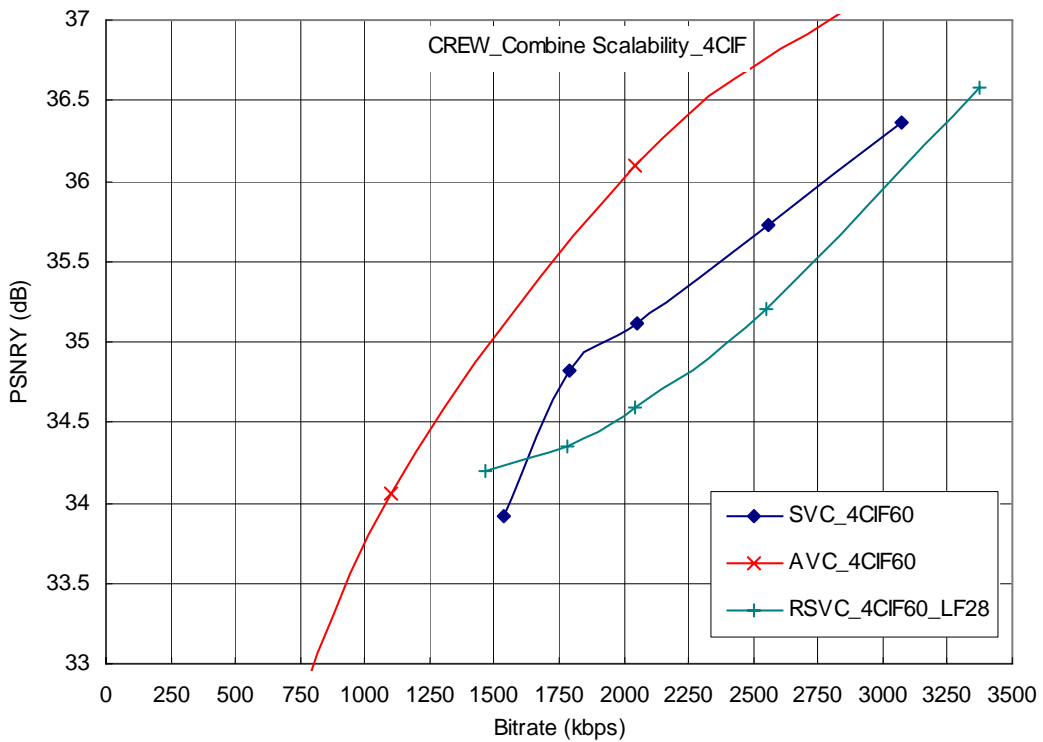
Figure 6.6. Simulation results for SNR scalability with FGS.

6.7.3 Combined Scalability

In this section, we show the combined scalability results. Three spatio-temporal resolutions are considered, including Qcif@15fps, Cif@30fps, and 4Cif@60fps. Because the bitrate range in each resolution is small, only one stack is used in RSVC. Other test conditions are the same with Section 6.7.2. Comparing with H.264/AVC SVC, at 4CIF, RSVC has 0.3dB gain at the lowest bitrate, and has at most 0.5dB loss at the medium and high bitrate. At CIF, RSVC has around to 0.7dB loss at the medium and high bitrate.



(a) Cif resolution



(b) 4Cif resolution

Figure 6.7. Simulation results for combined scalability.

6.8 Summary

In this chapter, the RFGS and SRFGS structure is extended to support spatio-temporal and SNR scalability simultaneously. Comparing with the H.264/AVC SVC, we relax the constraint and reduce the overhead of inter-layer prediction in spatial and SNR scalability. We extend the H.264/AVC CABAC to support bitplane coding and FGS. We also reduce the DPB requirement for supporting the temporal scalability. In the simulation results, RSVC has 0.8dB gain and 0.2dB loss at the low and high bitrate for spatial scalability, respectively. RSVC also provides 0.7dB gain in FGS. In combined scalability, RSVC provides -0.7dB to +0.3dB gain comparing with H.264/AVC SVC.



CHAPTER 7

Conclusions

The main contributions of this dissertation are summarized as following.

■ Robust Fine Granularity Scalability (RFGS)

We firstly proposed a novel FGS coding technique RFGS. The RFGS is a flexible framework that incorporates the ideas of leaky and partial predictions. Both techniques are used to provide fast error recovery when part of the bitstream is not available. The RFGS provides tools to achieve a balance between coding efficiency, error robustness and bandwidth adaptation. The RFGS covers several well-know techniques such as MPEG-4 FGS, PFGS and MC-FGS as special cases. Because the RFGS uses a high quality reference, it can achieve improved coding efficiency. The adaptive selection of bitplane number can be used to allow the tradeoff between coding efficiency and error robustness. The coding efficiency is maximized for a range of the target channel bandwidth. The enhancement layer information is scaled by a leak factor α , where $0 \leq \alpha \leq 1$ before adding to the base layer image to form the high quality reference frame. Such a leak factor is also used to alleviate the error drift.

Our experimental results show that the RFGS framework can improve the coding efficiency up to 4 dB over the MPEG-4 FGS scheme in terms of average PSNR. The error recovery capability of RFGS is verified by dropping the first few frames of a GOV at the enhancement layer. It is also demonstrated that tradeoff between coding

efficiency and error attenuation can be controlled by the leak factor α . We also provide an approach to select the parameters and its performance approaches that of a near-optimal exhaustive search of parameters. Such a technique provides a good balance between coding efficiency and error resilience.

■ **Stack Robust Fine Granularity Scalability (SRFGS)**

We further proposed the SRFGS to improve the performance of RFGS. Based on RFGS, the SRFGS generalizes its prediction concept and structure to a multi-layer stack architecture. In each layer, the information to be coded is temporally predicted by the information of the previous time instance at the same layer. The stack concept allows the SRFGS to optimize at several operating points for various applications. With the bit plane coding and leaky prediction used in RFGS, SRFGS maintains the feature of fine granularity and error robustness. An optimized MB-based alpha adaptation is proposed to improve the coding efficiency. We also propose single-loop enhancement layer decoding scheme to reduce the decoder complexity. The simulation results show that SRFGS has improvements by 0.4 to 3.0 dB in PSNR over RFGS. Further investigation of the bit allocation among each layer for various types of video content can provide better coding efficiency.

■ **Applications in the H.264/AVC SVC**

The SRFGS has been reviewed by the MPEG committee and has been ranked as one of the best algorithms according to the subjective testing in the Report on Call for Evidence on Scalable Video Coding. Furthermore, the proposed ideas in both of RFGS and SRFGS are also adopted in the developing H.264/AVC SVC. They show more than 2dB PSNR improvement.

■ **Robust Scalable Video Coding (RSVC)**

In RSVC, the RFGS and SRFGS structures are extended to support

spatio-temporal and SNR scalability simultaneously. To support spatial and SNR scalability, we provide a flexible inter-layer prediction method with modest overhead. We further extend the H.264/AVC CABAC to support bitplane coding and FGS. We implement the hierarchical prediction structure efficiently to support temporal scalability with limited decoded picture buffer. In the simulation results, RSVC has -0.7dB to +0.8dB PSNR improvement comparing with H.264/AVC SVC.



APPENDIX A

Streaming Video Application Based on H.264/AVC SVC for Mobile WiMAX

This Appendix shows a streaming video application of H.264/AVC SVC. The mobile WiMAX is adopted in the communication part for the streaming services.

A.1 Introduction

To address the increasing demand for broadband wireless access (BWA), the IEEE 802.16 family of standards [33][34] and their associated industry consortium, WiMAX forum, are developed and formed. The 802.16 family of standards aim to provide high data rate access over large areas to a large number of users. The 802.16-2004 standard [33], also referred to as WiMAX, provides such services for fixed subscribers in the wireless metropolitan area network (WirelessMAN). Recently, the 802.16e-2005 [34], also referred to as “mobile WiMAX”, extends the 802.16-2004 standard to further support the mobile subscribers that moving at vehicular speed. In this work, the mobile WiMAX system is adopted to develop the streaming video applications.

To serve video streaming for wireless mobile communication, SVC has several advantages over the non-scalable video coding. For a single user, the transmission bandwidth is time-varying due to the mobility and the fluctuated available resources. Besides, different users are located at different positions; the different signal quality leads to different transmission bandwidth. It is difficult to support all users with a single non-scalable bitstream. Moreover, there is no priority in the non-scalable bitstream.

This leads to inefficient error protection because both the more important data and the less important data have the same performance in an error-prone channel.

SVC provides simple solutions for these problems. According to the network conditions and receiver capabilities, the pre-encoded SVC bitstream can be easily adapted by the streaming server to provide various spatial, temporal and quality (SNR) resolutions. Further, the SVC layered structure put the data with different importance into different layers. The unequal erasure protection (UEP) can be easily incorporate with SVC to provide more protection for the more important data. With such features, the SVC bitstream is more suitable than the non-scalable bitstream to be transmitted over an error-prone channel with fluctuated bandwidth.

A.2 System Architecture

A.1.1 Overview of the System Architecture

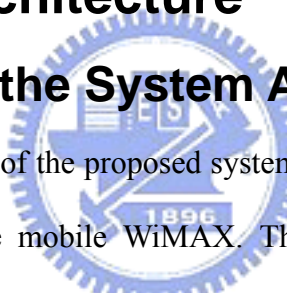


Figure A.1 shows an overview of the proposed system architecture for the H.264/AVC SVC video streaming over the mobile WiMAX. The proposed system architecture basically includes the streaming server, the base station (BS), and the mobile station (MS). Some pre-encoded video in H.264/AVC SVC format is stored in the streaming server. The MS send a request for video streaming to the BS. The channel quality is also feedback to the BS. The BS computes the available bandwidth between the MS and BS according to the channel quality, and sends the information to the streaming server. The streaming server adapts the requested video bitstream according to the available bandwidth, and sends the extracted video packet to the BS. The BS then transmits this data to MS.

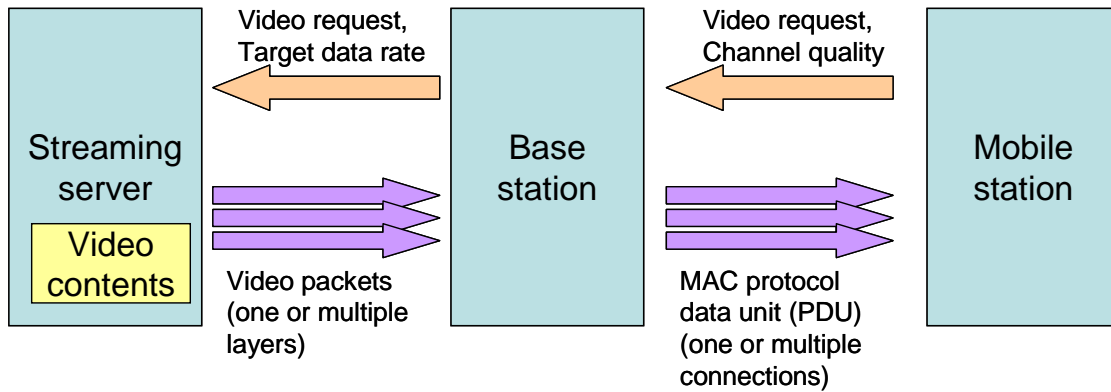


Figure A.1 SVC video streaming architecture

Because the bandwidth between BS and MS is time-varying, the above process is actually done repeatedly for each of a specified time period (referred to as “report period” in the following description). In each of the report period, the MS feedback the current channel quality to the BS, the BS computes the current available bandwidth and reports it to the streaming server. The server then extracts the video packets of this period and sends it back to the BS.

A.1.2 The H.264/AVC SVC Streaming Server

In each of the report period, the BS requests a target bitrate from the streaming server. The streaming server analyses the bitstream at the GOPs that covered by the current report period, and extract video packets according to the target bitrate, as shown in **Figure A.2**. In the SVC bitstream, the data at lower spatial-SNR resolution is more important. And in each spatial-SNR layer, the lower temporal layer is more important. Therefore, according to the requested bit-rate from the BS, the lower spatial-SNR layers are firstly extracted. At the spatial-SNR layer where the target bitrate cannot cover all the data, only the data at the lower temporal layers is extracted. With FGS coding, the data at the same temporal layers can be further truncated at any position to provide the exactly request bit-rate. The extracted data are then sent to the BS.

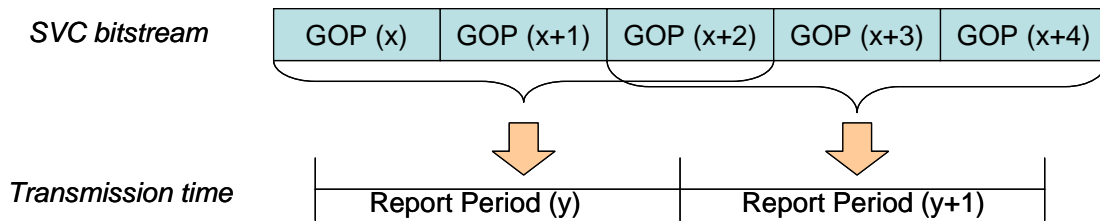


Figure A.2 The transmitted GOPs in each report period.

It should be mentioned that some GOPs may belong to two report periods (such as the GOP(x+2) in **Figure A.2**). For such GOPs, the data that already sent in the first report period will not be sent again. However, if the second report period allows higher bandwidth, the remaining data in such GOPs will be transmitted. This makes the video quality smoother when the bandwidth changes frequently. Depend on the pre-load time of the related streaming service, the number of the overlapped GOPs between the report periods can be further extended to provide more smooth video quality. Further, this structure also allows retransmission at the streaming server with suitable pre-load time.

Mobile WiMAX supports multiple connections between MS and BS. To support this feature, the streaming server can allocate the data to be sent to the BS into several connections according to the importance. The more important data is allocated to the connection that has more protection (i.e., higher transmission priority and/or more MAC retransmission times). To address the bandwidth fluctuation effect, in the proposed two-connection implementation, the server allocates the most important 80% data into the first connection, and put the remaining data into the second connection. This allows the BS need only re-transmit the more important data when the real bandwidth is smaller than the expected bandwidth. In the future work, the percentage of the data that put in the first connection should be adaptively according to the estimate channel model, which may further improve the overall video quality.

A.1.3 The Mobile WiMAX Simulation Platform²

The streaming server and the mobile WiMAX system (including the BS and MS) are inter-connected with an IP-based backhaul network. In the mobile WiMAX system, one or multiple connections are established between the BS and MS for the streaming video services. Each connection has its own QoS (quality of service) parameters, which allows the data with different importance to be handled differently. In the data transmission, the MS measure the channel quality of the downlink (DL) channel and feedback this information to the BS in the uplink (UL) channel. The BS collects this DL channel quality for computing the currently available bandwidth, and reports this data to the streaming server in each of the report period.

The streaming server then sends several video packets back to the BS. The video packets are stored as MAC service data units (SDUs) in the queue of the BS. In different channel conditions, the best size of the transmitted data unit, which is the MAC protocol data unit (PDU), is also different. Depend on the channel conditions, the BS will compute the best PDU size and fragment each SDU into multiple PDUs. To reduce the packet lost rate, the simulation platform adopts the MAC retransmission mechanism. When the PDU is lost and the MS has not return the acknowledge signal, BS will retransmit the lost PDU again. The retransmission time is configurable. Due to the mobility of the MS, handover from BS to BS might happen. To simplify the design of the simulation platform, a perfect seamless handover is assumed and the impact of the handover gap is not considered in the current work.

² The mobile WiMAX simulation platform used in the streaming video system are developed and provided by Hung-Hui Juan and Ching-Yao Huang, both are with the Department and Institute of Electronics Engineering, NCTU, Taiwan. In this section, we only provide a brief description of the mobile WiMAX simulation platform.

Table A.1 The average bitrate of the SVC bitstream at various spatial-SNR and temporal resolutions

Spatial-SNR layers	Bitrate@3.75fps	Bitrate@7.5fps	Bitrate@15fps	Bitrate@30fps
QCIF-SNR0	10.9801	14.8971	19.3172	23.9405
CIF-SNR0	56.6238	67.7029	81.5013	103.4000
CIF-SNR1	152.9394	172.2780	205.6717	247.6827
CIF-SNR2	335.3745	387.9433	456.7107	539.1969
CIF-SNR3	616.8722	747.8123	911.1053	1095.1940

A.3 Simulation Results

A.1.4 Test Conditions

The SVC bitstream used in the simulation is encoded by the reference software JSVM 4.8. We encode the bitstream to provide two spatial layers including QCIF (Quarter Common Intermediate Format, 176x144) and CIF (Common Intermediate Format, 352x288). As shown in Table A.1, the QCIF resolution has one SNR layer and the CIF resolution including four SNR layers. Including a QCIF layer allows the decoder has the flexibility to compensate lost (CIF layer) pictures at temporal and/or spatial domain.

In each of the spatial-SNR layers, the GOP size is limited to 8 pictures to reduce the buffer requirement at the mobile station but still provide four layers temporal scalability through the hierarchical prediction structure. The intra-pictures is inserted every 64 pictures (8 GOPs) to provide the error recovery point. With the five spatial-SNR layers and the four temporal layers in each spatial-SNR layer, up to 20 layers bitstream adaptation is allowed. Further, the FGS is used from the CIF-SNR1 to the CIF-SNR3 layers, such that the bitstream can be truncated at any point to achieve the requested bitrate. The test sequence is making up from 13 commonly used MPEG

test sequences to form a 3600 pictures video, which including bus, football, foreman, mobile, city, crew, harbour, soccer, coastguard, container, mother daughter, stefan, and table tennis. The average bitrate of the SVC bitstream at various spatial-SNR and temporal resolutions are shown in Table A.1.

To show the advantage of SVC, we also do the video streaming with the AVC bitstream for comparison. The AVC bitstream has no spatial and SNR scalability, but can achieve temporal scalability with hierarchical-B structure. Due to the limitation of scalability, we encode three AVC bitstream at different bitrate; therefore we can switch the AVC bitstream in different cell loading. The three bitstreams includes high, medium, and low average bitrate at 691kbps, 397kbps, and 228kbps, respectively. The GOP structure of the AVC bitstream is the same with the SVC bitstream. To serve different request bitrate, the streaming server truncates the AVC bitstream into different temporal resolution. The reference software JM10.1 is used in the simulation. The RD-optimization is enabled. The different bitrate is achieved with different constant QP, where the setting of the QPs at different temporal levels is according to [35].

In the mobile WiMAX simulation platform, the WirelessMAN OFDMA TDD (time division duplex) mode and PUSC (partial usage of sub-channel) are adopted in the PHY (physical layer) configurations. The cell loading is defined as the percentage of the occupied OFDMA slots in downlink. The FFT size is 2048 and the bandwidth is 6MHz. The OFDMA channelization parameters defined in [33] are used. The connection between the MS and BS could be 1 or 2. The max MAC retransmission time is set to be 6 (for the 1-connection service and both the 2 connections in the 2-connection service).

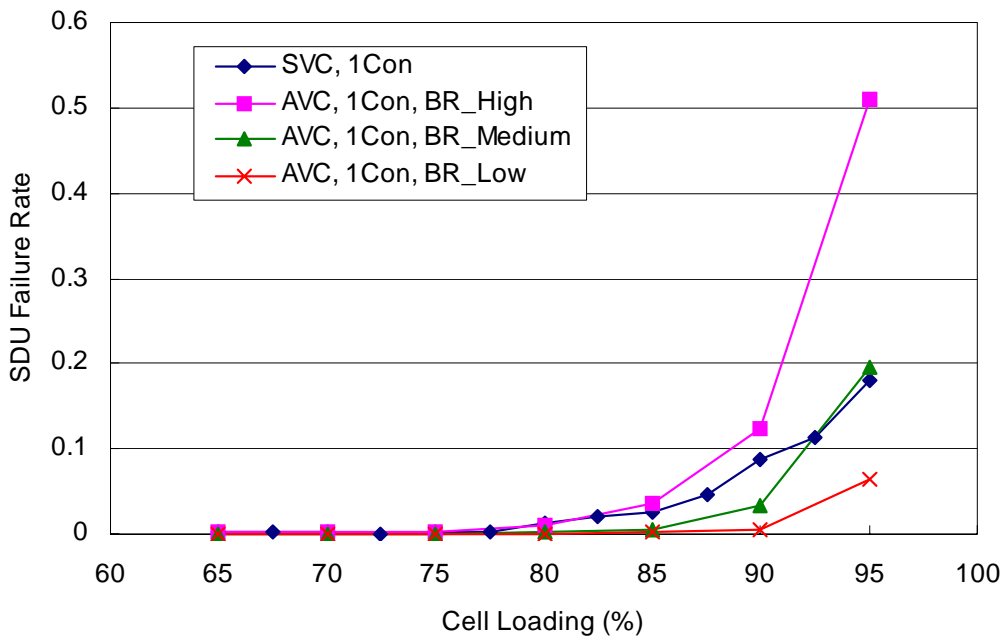


Figure A.3 The SDU failure rate in 1-connection service.

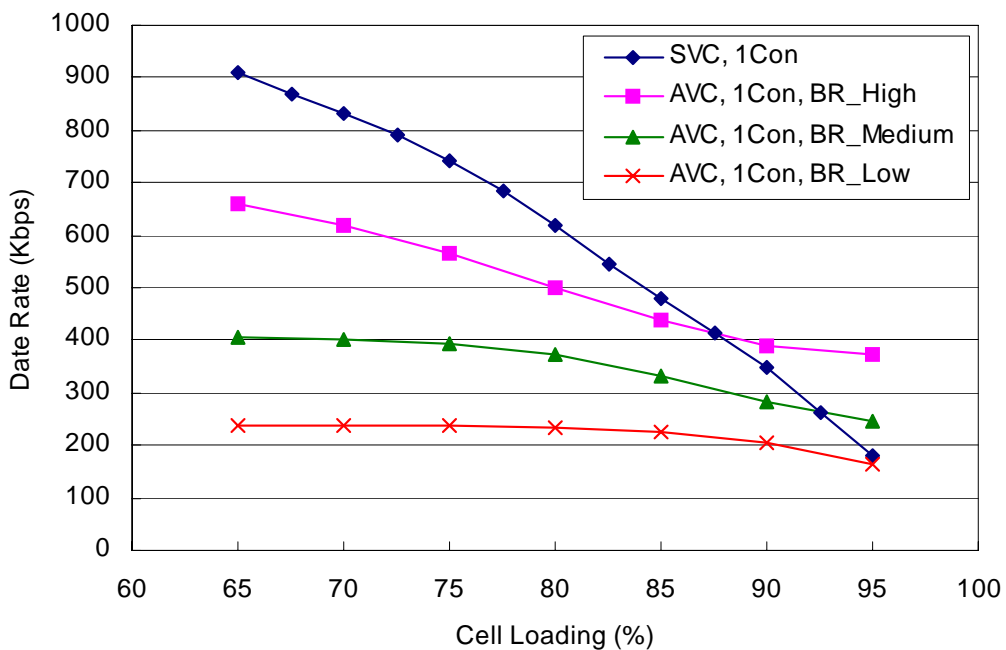


Figure A.4 The data rate in 1-connection service.

A.1.5 Simulation results

In the simulation, three combinations of the video encoding method and the

WiMAX transmission method are considered: SVC bitstream with 1-connection service, SVC bitstream with 2-connection service, and AVC bitstream with 1-connection service. The simulation results are the average performance of 5 tests in order to simulate the randomness of the channel errors.

Figure A.3 and **Figure A.4** shows the SDU failure rate of the 1-connection and 2-connection services, respectively. For the two connection service, the connection A transmits more important data and connection B transmits less important data. We can found that for the SVC bitstream and at the same cell loading, the connection A in 2-connection service has much lower SDU failure rate comparing with the 1-connection service. This is because when the bandwidth decreases, connection A has higher transmission priority comparing with connection B such that the data in connection B is dropped first. For the AVC bitstream, due to the limitation of scalability, larger original average bitrate cause larger SDU failure rate. At cell loading equal to 95%, the high-bitrate AVC bitstream has 50% SDU failure rate. This is because the available transmission bitrate is smaller than the minimum bitrate of the AVC bitstream (at the smallest temporal resolution), such that the channel is overloaded.

Figure A.5 and **Figure A.6** shows the data rate of the 1-connection and 2-connection cases, respectively. Data rate is the bitrate of the received video packet at MS; it might smaller than the total bandwidth because some bandwidth is consumed by the retransmission. Regarding the SVC bitstream, the data rate decrease according to the cell loading. In the 2-connection scenario, around 80% data is allocated in connection A, and remaining is allocated in connection B. Regarding the AVC bitstream, temporal scalability provides a limited bitrate adaptation. When the cell is heavily loaded, the data rate may not decrease sufficiently and thus overload the channel and increase the SDU failure rate.

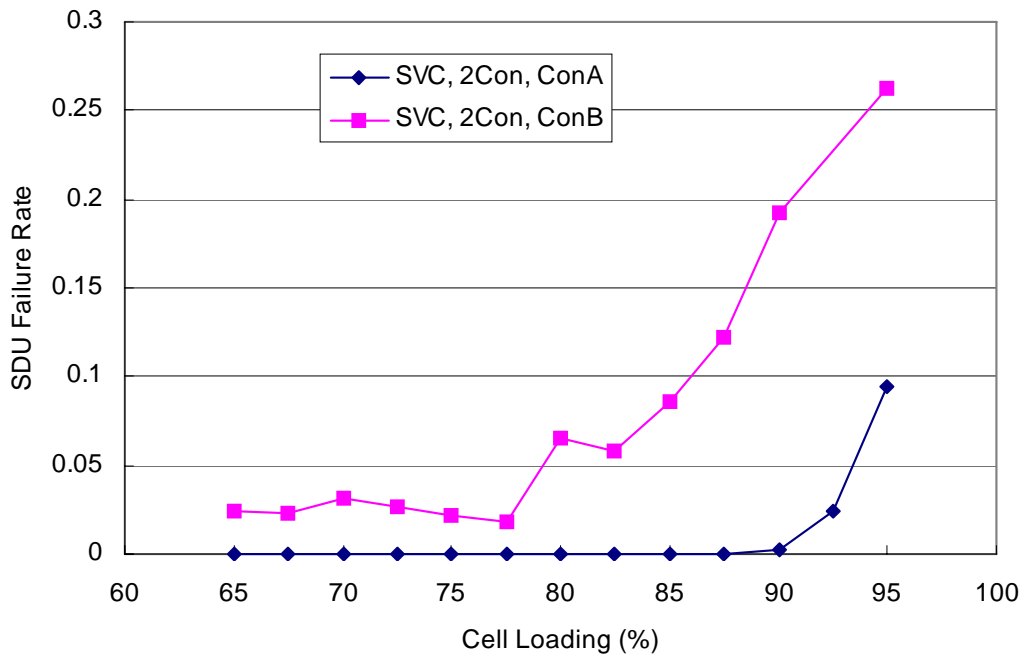


Figure A.5 The SDU failure rate in 2-connection service.

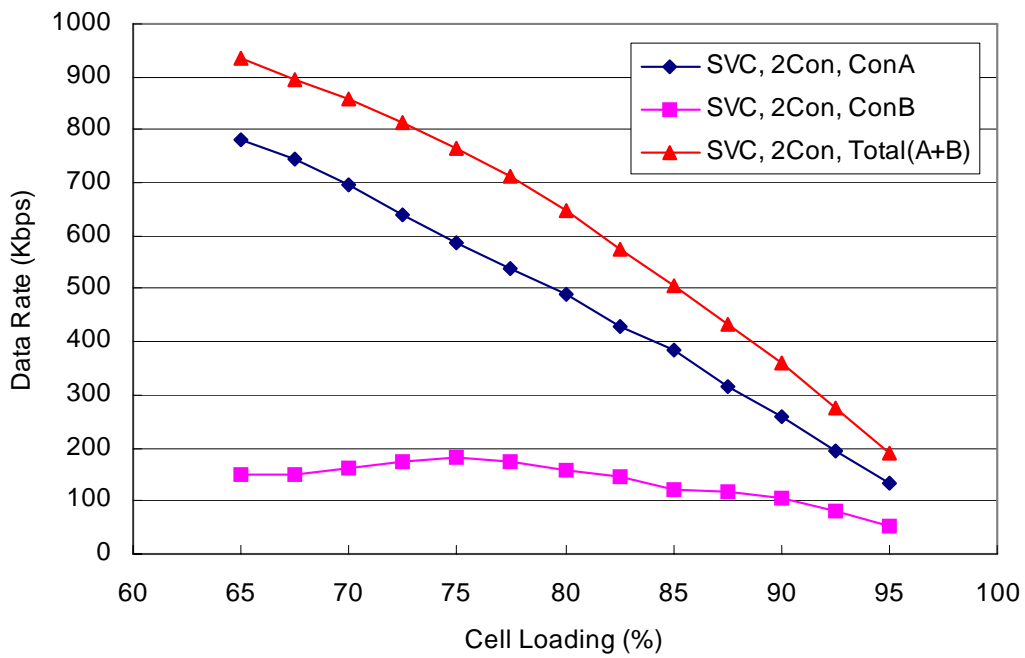


Figure A.6 The data rate in 2-connection service.

Figure A.7 shows the PSNR of the decoded video at the MS. When a picture is lost, the previous picture is repeated for computing PSNR. For the SVC bitstream, the

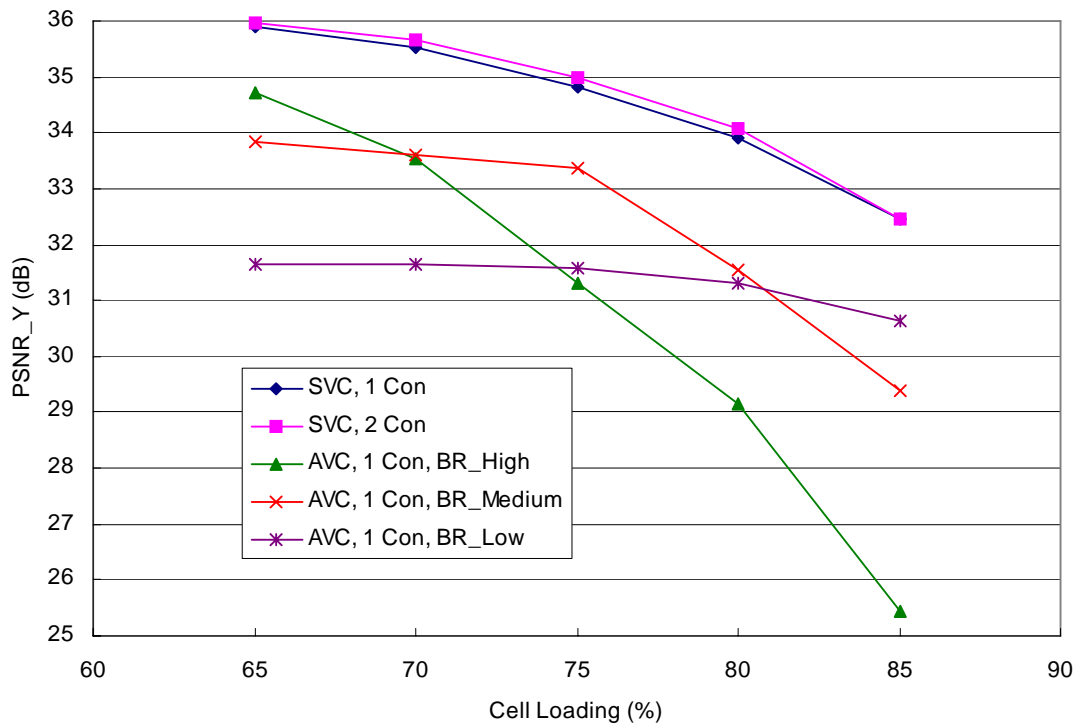


Figure A.7 The PSNR results of the streaming services.

2-connection service has up to 0.15 dB PSNR improvement over the 1-connection service. For the AVC bitstream, it is clear that temporal scalability is not sufficient to support the large varying bitrate and bitstream switching among different cell loading is required: with the increasing of the cell loading, we should switch to the AVC bitstream that has lower average bitrate. Comparing with the best performance of AVC bitstreams, SVC bitstream achieve 1.2 to 1.8 dB improvement.

A.4 Summary

In this Appendix, we build application architecture of the H.264/AVC SVC streaming services with the mobile WiMAX system. Both one and two connections services are considered. A stream server is developed that can adapt the bitstream and separate the video packets into different connections according to its importance. The

streaming services with AVC bitstream are also performs to show the advantage of the SVC. The simulation results show the SVC has more than 1dB PSNR improvement over the AVC bitstream. The two connection service also slightly improves the transmission robustness and video quality over the one connection service.



Bibliography

- [1] "Advance Video Coding for Generic Audiovisual Services," ITU-T and ISO/IEC JTC1, ITU-T Recommendation H.264 – ISO/IEC 14496-10 AVC, 2003.
- [2] "Joint Draft 5: Scalable Video Coding," ITU-T and ISO/IEC JTC1, JVT-R201, Jan. 2006.
- [3] "Joint Scalable Video Model JSVM-5," ITU-T and ISO/IEC JTC1, JVT-R202, Jan. 2006.
- [4] H. Schwarz, D. Marpe, and T. Wiegand, "Comparison of MCTF and closed-loop hierarchical B pictures," ITU-T and ISO/IEC JTC1, JVT-P059, Jul. 2005.
- [5] J. R. Ohm, "Advances in Scalable Video Coding," *Proceedings of the IEEE*, vol. 93, no. 1, pp. 42–56, Jan. 2005.
- [6] "Streaming video profile-- Final Draft Amendment (FDAM 4)," ISO/IEC JTC1/SC29/WG11/N3904, Jan. 2001.
- [7] F. Wu, S. Li, Y. Q. Zhang, "A framework for efficient progressive fine granularity scalable video coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 11, no. 3, pp. 332–344, March. 2001.
- [8] H. C. Huang, C. N. Wang, T. Chiang, "A Robust Fine Granularity Scalability Using Trellis Based Predictive Leak," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 12, pp. 372–385, June 2002.
- [9] H. C. Huang and T. Chiang, "Stack Robust Fine Granularity Scalability", *IEEE Int. Symp. Circuits Syst*, 2004.
- [10] W. Li, "Overview of fine granularity scalability in MPEG-4 video standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 11, pp. 301–317, Mar. 2001.
- [11] MPEG Video Group, "Information technology — Coding of audio-visual objects—Part 2: Visual ISO/IEC 14496-2: 2001," *International Standard*, ISO/IEC JTC1/SC 29/WG 11 N4350, July 2001.
- [12] M. Schar, H. Radha, "Motion-compensation based fine-granular scalability (MC-FGS)," ISO/IEC JTC1/SC29/WG11, MPEG00/M6475, Oct. 2000.
- [13] K. Y. Chang, R. W. Donaldson, "Analysis, optimization, and sensitivity study of differential PCM systems operating on a noisy communication channels," *IEEE Trans. Commun.*, vol. COM-20, pp. 338-350, June 1972.
- [14] M. Ghanbari and V. Seferidis, "Efficient H.261-based two-layer video codecs for ATM networks," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 5, no. 2, pp. 171–175, Apr.

1995.

- [15] A. Fuldseth and T. A. Ramstad, "Robust subband video coding with leaky prediction, " in *Proc. DSP Workshop*, Loen, Norway, Sept. 1996, pp. 57-60.
- [16] S. Li, F. Wu, and Y.-Q. Zhang, " Experiment Results with Progressive Fine Granularity Scalable (PFGS) Coding, " ISO/IEC JTC1/SC29/WG11, MPEG99/M5742, Oct. 1999.
- [17] "FGS experiments, " ISO/IEC JTC1/SC29/WG11, MPEG00/N3316, March. 2000.
- [18] F. Wu, S. Li, X. Y. Sun, and Y.-Q. Zhang, " MacroblocK-based progressive fine granularity scalable coding, " ISO/IEC JTC1/SC29/WG11, MPEG01/M6779, Jan. 2001.
- [19] "Report on MPEG-4 Visual Fine Granularity Scalability Tools Verification Test, " ISO/IEC JTC1/SC29/WG11, MPEG02/M8002, Jan. 2002.
- [20] "Call for evidence on scalable video coding advances," ISO/IEC JTC1/SC29/WG11/N5559, March 2003.
- [21] H. C. Huang, W. H. Peng, C. N. Wang, T. Chiang, and H. M. Hang, "Stack Robust Fine Granularity Scalability: Response to Call for Evidence on Scalable Video Coding " ISO/IEC JTC1/SC29/WG11/M9767, July 2003.
- [22] J. W. Wood and P. Chen, "Improved MC-EZBC with Quarter-pixel Motion Vectors" ISO/IEC JTC1/SC29/WG11/M8366, May 2002.
- [23] "Report on Call for Evidence on Scalable Video Coding (SVC) technology," ISO/IEC JTC1/SC29/WG11/N5701, July 2003.
- [24] H. C. Huang, C. N. Wang, T. Chiang, and H. M. Hang, "H.26L-based Robust Fine Granularity Scalability (RFGS), " ISO/IEC JTC1/SC29/WG11/M8604, July 2002.
- [25] Y. He, R. Yan, F. Wu, and S. Li, "H.26L-based fine granularity scalable video coding, " ISO/IEC JTC1/SC29/WG11/M7788, Dec. 2001.
- [26] T. Ruser and M. Wien, "AVC Anchor Sequences for Call for Evidence on Scalable Video Coding Advances " ISO/IEC JTC1/SC29/WG11/M9725, July 2003.
- [27] JVT reference software version 4.2, <http://iphome.hhi.de/suehring/tml/download/>
- [28] I. Amonou, N. Cammas, S. Kervadec, and S. Pateux, "Layered quality optimization of JSVM-3 when considering closed-loop encoding," ITU-T and ISO/IEC JTC1, JVT-Q081, Oct. 2005.
- [29] Y. Bao, M. Karczewicz, J. Ridge, and X. Wang, "Improvements to Fine Granularity Scalability for Low-Delay Applications," ITU-T and ISO/IEC JTC1, JVT-O054, Apr. 2005.
- [30] Y. Bao and M. Karczewicz, "CE7 Report, FGS coding for low-delay applications," ITU-T and ISO/IEC JTC1, JVT-Q039, Oct. 2005.
- [31] X. Wang, M. Karczewicz, J. Ridge, and N. Ammar, "CE7 Report, Multiple FGS layer coding for low-delay applications," ITU-T and ISO/IEC JTC1, JVT-R077, Jan. 2006.
- [32] "Core Experiment on FGS coding for Low Delay Applications (CE-7)," ITU-T and ISO/IEC JTC1, JVT-P307r1, July. 2005.

- [33] IEEE 802.16-2004, "IEEE Standard for Local and metropolitan area networks Part 16: Air Interface for Fixed Broadband Wireless Access Systems," June, 2004.
- [34] IEEE 802.16e-2005, "IEEE Standard for Local and metropolitan area networks Part 16: Air Interface for Fixed and Mobile Broadband Wireless Access Systems, Amendment 2: Physical and Medium Access Control Layers for Combined Fixed and Mobile Operation in Licensed Bands and Corrigendum 1," February, 2005.
- [35] H. Schwarz, D. Marpe, and T. Wiegand, "Hierarchical B pictures," ITU-T and ISO/IEC JTC1, JVT-P014, Jul. 2005.
- [36] W.-H. Peng; T. Chiang; H.-M. Hang; and C.-Y. Lee, "A context adaptive bit-plane coder with maximum-likelihood-based stochastic bit-resuffling technique for scalable video coding," IEEE Trans. Multimedia, vol. 8, pp. 654–667, Aug. 2006.



Curriculum Vitae

Personal Information

- Hsiang-Chun Huang
- Ph.D Candidate, Institute of Electronics, National Chiao-Tung University
- 1001 Ta-Hsueh Road, HsinChu 30010, Taiwan.
- Office: ED422
- E-mail: sleeping.ee89g@nctu.edu.tw

Education

- Ph.D Candidate, Institute of Electronics, National Chiao-Tung University, HsinChu, Taiwan, 2001—present.
- M.S. Student, Institute of Electronics, National Chiao-Tung University, HsinChu, Taiwan, 2000—2001.
- B.S. Degree, Dept. of Electronics Engineering, National Chiao-Tung University, HsinChu, Taiwan, 1996—2000.

Research Interest

- Video Compression
 - Scalable video coding
 - Codec optimization

Work Experience

- 2004-present, Member of Technique Staff
 - Ambarella Co., Sunnyvale, Calif., USA.
 - Develop the video coding algorithm for H.264/AVC SoC
 - Develop the video coding reference model for H.264/AVC SoC
 - Develop the auto focus algorithm for digital camcorder
- 2003-2004, Research Assistant
 - Research project of Ambarella Co., Sunnyvale, Calif., USA.
 - Develop the video coding algorithm for H.264/AVC SoC
- 2001-present, Research Assistant
 - Institute of Electronics Engineering, NCTU
 - Invent the Robust Fine Granularity Scalability
 - Invent the Stack Robust Fine Granularity Scalability
 - Develop a scalable video-on-demand server with variable block size motion descriptor (with member in laboratory)
 - Develop a H.264/AVC compliant encoder with the function of fast/inverse

- video playback (with member in laboratory)
- 2001-2003, Teaching Assistant
 - 視訊多媒體培訓班多媒體製作實務, NCTU/經濟部工業局
 - Edit the handout, including MPEG-2/MPEG-4/AVC encoder/decoder introduction, codec speed optimization, and error resilience decoder development.
 - Give instructions in NCTU, Ulead Co., and Acer Aspire Park.
- 2000-2001, Research Assistant
 - Research project of Luxxon Co., Mountain View, Calif., USA.
 - To prove the concept of MPEG-4 FGS in multimedia streaming system.
 - Develop the MPEG-4 FGS error resilience decoder.

Publication

- Journal Paper
 1. **H. C. Huang**, W. H. Peng, T. Chiang, and H. M. Hang, “Advances in the Scalable Extension of H.264/AVC,” *IEEE Communication Magazine*, 2006. (Accepted with minor revision)
 2. H. H. Juan, **H. C. Huang**, C. Y. Huang, and T. Chiang, “The Cross-layer Mobile WiMAX MAC Designs for H.264/AVC based Scalable Video Coding Services,” *IEEE Journal on Selected Areas in Communications*, 2006. (Submitted)
 3. **H. C. Huang** and T. Chiang, “Stack Robust Fine Granularity Scalable Video Coding,” *Journal of the Chinese Institute of Engineers*, 2006. (Accepted and to be published)
 4. **H. C. Huang**, C. N. Wang, and T. Chiang, “A Robust Fine Granularity Scalability Using Trellis Based Predictive Leak,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 12, pp. 372–385, June 2002.
- Conference Paper
 1. S. H. Yang, **H. C. Huang**, T. Chiang, “Efficient VCR Functionality Implementation in AVC with Symmetric Tree Prediction Structure,” *Workshop on Consumer Electronics*, 2005.
 2. **H. C. Huang** and T. Chiang, “Stack Robust Fine Granularity Scalability”, *IEEE Int. Symp. Circuits Syst*, 2004.
 3. B. Y. Wang, **H. C. Huang**, T. Chiang, “Complexity Reduction Using a Variable Block Size Motion Descriptor for a Scalable Video-On-Demand Server,” *Workshop on Consumer Electronics*, 2003.
 4. **H. C. Huang**, C. N. Wang, T. Chiang, “A Robust Fine Granularity Scalability Using Trellis Based Predictive Leak ,” *IEEE Int. Symp. Circuits Syst*, 2002
 5. **H. C. Huang**, C. N. Wang, T. Chiang, “MPEG-4 streaming video for wireless applications,” *International Symposium on Communications*, 2001.
 6. **H. C. Huang**, C. N. Wang, T. Chiang, “MPEG-4 streaming video profile,” *Workshop on Lee and MTI center*, 2001.

7. W. S. Peng, **H. C. Huang**, T. Chiang, “Optimization of selective enhancement for MPEG-4 fine granularity scalability,” *Workshop on Consumer Electronics*, 2000.

■ MPEG Contribution

1. **H. C. Huang**, W. H. Peng, Y. C. Lin, C. N. Wang, T. Chiang, H. M. Hang, “Update of the Response to Cfp on Scalable Video Coding Technology: Proposal S07 -- A Robust Scalable Video Coding Technique,” ISO/IEC JTC1/SC29/WG11, MPEG04/M10724, March 2004.
2. **H. C. Huang**, W. H. Peng, Y. C. Lin, C. N. Wang, T. Chiang, H. M. Hang, “Response to Cfp on Scalable Video Coding Technology: Proposal S07 -- A Robust Scalable Video Coding Technique,” ISO/IEC JTC1/SC29/WG11, MPEG04/M10569/S07, March 2004.
3. **H. C. Huang**, W. H. Peng, C. N. Wang, T. Chiang, H. M. Hang, “Stack Robust Fine Granularity Scalability: Response to Call for Evidence on Scalable Video Coding,” ISO/IEC JTC1/SC29/WG11, MPEG03/M9767, July 2003.
4. **H. C. Huang**, C. N. Wang, T. Chiang, “H.26L-based Robust Fine Granularity Scalability (RFGS) ,” ISO/IEC JTC1/SC29/WG11, MPEG02/M8604, July 2002.
5. **H. C. Huang**, C. N. Wang, T. Chiang, “A Robust Fine Granularity Scalability (RFGS) Using Predictive Leak ,” ISO/IEC JTC1/SC29/WG11, MPEG02/M8409, May 2002.
6. **H. C. Huang**, C. N. Wang, T. Chiang, “Evidence for improving the existing fine granularity scalability tool,” ISO/IEC JTC1/SC29/WG11, MPEG01/M7393, July 2001.
7. **H. C. Huang**, T. Chiang, “Verifications results of CE Q7: Sprite Generation,” ISO/IEC JTC1/SC29/WG11, MPEG01/M7122, March 2001.
8. **H. C. Huang**, C. N. Wang, T. Chiang, “Verification Result of PFGS,” ISO/IEC JTC1/SC29/WG11, MPEG01/M6890, Jan. 2001.

■ Patent

1. **H. C. Huang**, C. N. Wang, T. Chiang, H. M. Hang, “Architecture and method for fine granularity scalable video coding,” United States patent, patent number 7,072,394. 2006.
2. **H. C. Huang**, C. N. Wang, T. Chiang, H. M. Hang, “Architecture for stack robust fine granularity scalability,” filed United States patent, application number 10/793830. 2005
3. **H. C. Huang**, C. N. Wang, T. Chiang, H. M. Hang, “Architecture and method for fine granularity scalable video coding,” filed United States patent, application number 11/136780. 2005.
4. **黃項群**, 王俊能, 蔣迪豪, 杭學鳴, “可調整位元流大小的影像編解碼裝置” 中華民國專利, 公告號I233306, 2005。
5. **黃項群**, 王俊能, 蔣迪豪, 杭學鳴, “堆疊式影像編碼與解碼裝置” 中華民國專利, 公告號I242979, 2005。