FAST MODE DECISION METHODS IN H.264/AVC STANDARD

A

Thesis submitted

in Partial Fulfilment of the Requirements

for the degree of

DOCTOR OF PHILOSOPHY

By

AMRITA GANGULY

to the

Department of Electronics and Electrical Engineering

Indian Institute of Technology Guwahati

Guwahati - 781039, Assam, India

September 2011

FAST MODE DECISION METHODS IN H.264/AVC STANDARD

Amrita Ganguly

This thesis is dedicated to my father and the greatest influence on my life

Late Mr.Amiya Kumar Ganguly

Certificate

This is to certify that the thesis entitled "**Fast Mode Decision Methods in the H.264/AVC Standard**", submitted by **Amrita Ganguly** (06610215), a research scholar in the *Department of Electronics and Electrical Engineering, Indian Institute of Technology Guwahati*, for the award of the degree of **Doctor of Philosophy**, is a record of an original research work carried out by her under my supervision and guidance. The thesis has fulfilled all requirements as per the regulations of the institute and in my opinion has reached the standard needed for submission. The results embodied in this thesis have not been submitted to any other University or Institute for the award of any degree or diploma.

Dated:

Guwahati.

Anil Mahanta
Professor
Dept. of Electronics and Electrical Engg.
Indian Institute of Technology Guwahati
Guwahati - 781039, Assam, India.

Acknowledgements

I would like to begin by thanking my parents. I understand any amount of gratitude shown to them is inadequate. My late father's unconditional support in every aspect of my life is the inspiration that this PhD is completed.

I feel it as a great privilege in expressing my sincere gratitude to my supervisor Prof. Anil Mahanta for his excellent guidance. He has supported and motivated me all through my PhD study.

I am also thankful to my doctoral committee members Prof. P. K. Bora, Dr. Rohit Sinha and Dr. S. R. M. Prasanna for sparing their precious time to evaluate the progress of my work. I am grateful to Dr. A. Mishra who was in my doctoral committee for a brief period. Their suggestions have been invaluable. I would also like to thank other faculty members for their kind help extended during my academic studies.

I am grateful to all the members of the research and technical staff of the department without whose help I could not have completed this thesis. My special thanks to Josephine, Sanjib, Mr. L. N. Sharma, Sidananda and Jharna for maintaining an excellent computing facility and providing various resources useful for the research work.

My friends at IITG made my life joyful and were constant source of encouragement. Among my friends, a special thanks to Nirmala with whom I would always share the ups and downs of my research work. I would like to thank Padam Priyal, Shyam, Rupaban, Mrinal, Sumitra, Narasimhamurthy, Kuntal, Mridupawan, Dhrubajyoti, Dayananda and Vijay. Their love and support made this a special place and made me enjoy my work.

My deepest gratitude goes to my family for their continuous love and support throughout my studies specially to my most precious jewels, Radha and Roop. They never understood my absence from home. My husband, Hemanta, has constantly inspired me. A special thanks to my mother and mother-in-law for their unconditional support.

Amrita Ganguly

Abstract

The H.264/AVC video coding standard offers significantly improved compression efficiency and flexibility compared to previous standards. H.264/AVC provides better video at a lower bit rate and this translates into lesser storage requirement. H.264/AVC compression makes it possible to transmit HD television over a limited-capacity broadcast channel. H.264/AVC is a sophisticated compression method. However, for all its advantages, it must be acknowledged that it requires more processing power to encode video with H.264/AVC as compared to the previous standards. The standard is complex and its implementation is more challenging. The computationally expensive H.264/AVC coder can lead to higher encoding and decoding times.

This thesis presents some new methods for reduction of the encoding time and the complexity of the H.264/ AVC encoder. A detailed study of the intra and inter prediction processes was carried out. Based on the observations gleaned from this study, a number of fast mode decision algorithms are proposed in the spatial domain and in the transform domain.

Natural videos have objects with strong edges. Homogeneous region in a frame can be detected from the edges in it. This characteristic of the video is utilized for a new algorithm for intra and inter prediction process based on the edge histogram characteristic of the image. The reduction in complexity is achieved by selecting only a few coding modes from the total available modes depending upon the edge histogram characteristics and the homogeneity of the image.

There are many regions in a video that are stationary. These regions generally get encoded in the SKIP mode. SKIP mode is simple to compute and if this mode can be detected prior to performing the inter prediction process, large computational saving is possible in the encoder. Thus, a low complexity algorithm is developed by predicting the "skipped" macroblocks prior to motion estimation through early detection of the zero quantized coefficients. A weighted prediction algorithm is next proposed that identifies a macroblock based on certain parameters. Weights are assigned for these parameters

and the encoding modes are selected accordingly. These algorithms are developed in the spatial domain.

Transform domain based fast encoding algorithms are proposed where the statistics of the quantized transform coefficients are utilized for the mode decision process. The energy of the transform coefficients are usually concentrated in the low fequency components. An algorithm is proposed where the decision is taken from the study of the low frequency coefficients of a block. Finally, a Subband/DCT based fast encoding algorithm is developed for H.264/AVC where the wavelet decomposition of the input image is utilized for the fast mode decision process. This concept is also used in a method for fast intra prediction in Scalable Video Coding.

Simulation results show that the algorithms achieve significant savings in complexity with a negligible loss in rate-distortion performance. Subjective evaluations show that these techniques result in similar perceptual quality when compared to a reference encoder JM12.4 of the H.264/AVC. These algorithms are likely to be useful in implementing real-time H.264/AVC standard encoders in low bitrate applications and for devices with restricted computational capability.

Contents

List of Figures xvii

List of Tables xxi

List of Acronyms xxiii

List of Symbols xxvii

1 Introduction 1

1.1 The Role of Standards . . . 3

1.2 Motivation for Fast Mode Decision . . . 5

1.3 Contributions and Overview of the Thesis . . . 9

2 H.264/AVC Video Coding Standard 12

2.1 Subjective Quality Measurement . . . 13

2.2 Objective Quality Measurement . . . 14

2.3 H.264/AVC Video Coding Standard . . . 15

2.4 H.264/AVC Encoder Processes . . . 16

2.4.1 Prediction Process . . . 16

2.5 Intra Prediction Process . . . 17

2.5.1 4×4 Intra Prediction . . . 17

2.5.2 16×16 Intra Prediction Process (Luma Component) . . . 19

2.5.3 Chroma Prediction Modes . . . 20

2.6 Inter Prediction Process . . . 20

2.6.1 SKIP mode . . . 22

2.6.2 Motion Vector Prediction . . . 22

2.7 Transform and Quantization . . . 23
2.7.1 Forward Transform . . . 24
2.7.2 Quantization . . . 25
2.7.3 Entropy Coding . . . 27
2.7.4 Rate Distortion Optimization . . . 28
2.8 Low Complexity Coding: Fast Mode Decision Algorithms . . . 30
2.8.1 Fast Intra Prediction Algorithms: A Review . . . 31
2.8.2 Fast Inter Prediction Algorithms: A Review . . . 33
2.9 Conclusions . . . 36
3 Experimental Method 39
3.1 Introduction . . . 40
3.2 Implementations . . . 40
3.2.1 Reference Software . . . 40
3.2.2 Test Platform . . . 40
3.2.3 Implementation and Test of the Algorithms . . . 41
3.3 Test Video Sequences Used for Experiments . . . 41
3.3.1 Classification of Sequences . . . 42
3.4 Performance Evaluation . . . 43
3.4.1 Rate Distortion Performance . . . 44
3.4.2 Computational Complexity . . . 45
3.4.3 Bit Rate . . . 47
3.4.4 Video Quality . . . 47
3.5 Conclusion . . . 48
4 Edge Histogram Based Fast Mode Decision for H.264/AVC 49
4.1 Observations from Exhaustive Intra Prediction Process . . . 50
4.1.1 Mode dependency of sequences at different QP . . . 51
4.1.2 Directional Correlation . . . 52
4.1.3 Residual Complexity . . . 54

4.2 Fast Intra Mode Decision in EHFMD Algorithm . . . 55
4.2.1 Determination of Edge Histogram . . . 56
4.2.2 Determination of Residual Complexity . . . 56
4.2.3 Overall Intra Prediction in EHFMD Algorithm . . . 57
4.3 Experimental Observations . . . 58
4.3.1 Discussion . . . 61
4.4 Observations from Full Search ME in H.264/AVC . . . 61
4.4.1 Mode Distribution for Different Test Sequences . . . 64
4.4.2 Correlation Between Prediction Block Size and Motion Direction . . . 64
4.5 Fast Inter Mode Decision in EHFMD Algorithm . . . 67
4.5.1 Determination of Stationarity . . . 67
4.5.2 Determination of Homogeneous Region . . . 71
4.5.3 Directional Motion Estimation . . . 73
4.5.4 Overall Inter Prediction in EHFMD Algorithm . . . 74
4.6 Experimental Results . . . 76
4.6.1 *Distortion and Compression Ratio Comparisons* . . . 76
4.6.2 *Computational Speedup* . . . 77
4.7 Conclusions . . . 78

5 Fast Mode Decision based on Early Zero Block Detection and Weighted Prediction 81

5.1 Early Zero Block Detection based Fast Mode Decision (EZBD-FMD) . . . 83
5.1.1 Further Observations from Full Search ME . . . 83
5.1.2 SKIP mode detection from frame difference residues . . . 84
5.1.3 Early Detection of AZCB for H.264/AVC . . . 86
5.2 EZBD-FMD Algorithm . . . 89
5.2.1 *Level A: Early SKIP mode Decision* . . . 90
5.2.2 *Level B: Early Detection of ZBs* . . . 92
5.3 Experimental Observations . . . 95

5.3.1 *Distortion and Compression Ratio Comparisons* 95
5.3.2 *PSNR-Rate Curves* 97
5.3.3 *Subjective Comparison* 99
5.4 Mode Decision Using Weighted Prediction 103
5.4.1 Determination of Motion Content in a MB 103
5.4.2 Determination of Homogeneity in a MB 104
5.4.3 Determination of Predicted MV 105
5.4.4 Determination of Predicted mode from Neighboring Modes 106
5.4.5 Overall WBFMD Algorithm 107
5.5 Experimental Observations 108
5.5.1 *Distortion and Compression Ratio Comparisons* 108
5.5.2 *Computational Speedup* 109
5.5.3 *Comparison with JM12.4 Modes* 109
5.6 Conclusions 109
6 Fast Mode Decision in the Transform Domain 111
6.1 Transform Domain based Fast Mode Decision (TDFMD) 113
6.1.1 Observations from Transform Coefficients in Intra Prediction Process 113
6.1.2 Observations from Transform Coefficients in Inter Prediction Process 115
6.2 Fast Intra Prediction of TDFMD 116
6.2.1 Experimental Observations with All Intra Frames 117
6.3 Fast Inter Prediction of TDFMD 118
6.4 Experimental Observations 120
6.4.1 *Distortion and Compression Ratio Comparisons* 120
6.5 Selective Coefficient Fast Mode Decision (SCFMD) 123
6.5.1 Observation On Selective Transform Coefficients 123
6.6 Proposed SCFMD Algorithm 125
6.7 Experimental Observations 126
6.7.1 *Distortion and Compression Ratio Comparisons* 127

6.7.2 *PSNR-Rate Curves* . . . 127
6.8 Conclusions . . . 128

7 Subband/DCT Based Fast Mode Decision Algorithms 131

7.1 Subband/DCT based Fast Mode Decision for H.264/AVC (SBFMD) . . . 132
7.1.1 Proposed SBFMD Algorithm . . . 133
7.2 Experimental Observations . . . 136
7.2.1 Complexity Analysis . . . 136
7.2.2 *Distortion and Compression Ratio Comparisons* . . . 136
7.3 Subband/DCT based Fast Encoding Algorithm For Scalable Video Coding 138
7.3.1 Analysis of Intra prediction modes . . . 139
7.3.2 Proposed Intra Fast Mode Decision Algorithm for SVC . . . 140
7.4 Experimental Observations . . . 143
7.5 Conclusions . . . 144

8 Fast Mode Decision for Intra only Spatial SVC 147

8.1 Analysis of Intra prediction modes . . . 148
8.1.1 Correlation of Modes Across Layers . . . 149
8.2 Proposed Fast Mode Decision Algorithm . . . 149
8.2.1 Steps in Fast Mode Decision in Spatial SVC . . . 150
8.3 Experimental Observations . . . 153
8.4 Conclusions . . . 156

9 Summary and Conclusions 157

9.1 Summary of the Work . . . 158
9.2 Scope for the Future Work . . . 163

Bibliography 165

List of Figures

1.1 Runtime percentage of functional blocks in H.264/AVC [1] 7

2.1 Block diagram of the encoding process for the VCL of H.264/AVC [2] . . . 16
2.2 Labeling of Prediction samples and the Directions of prediction 18
2.3 4×4 Intra Prediction Modes [3] 18
2.4 16×16 Intra Prediction Modes [3] 19
2.5 Partitioning of a macroblock and a sub-macroblock for motion-compensated prediction [2] 21
2.6 Current and neighbouring partitions 23

3.1 Test Scenario For the Implementations 41
3.2 Sample frames from test video sequences in CIF resolution 44
3.3 Rate Distortion Plot 45

4.1 Distribution of MBs in I4MB and I16MB modes 52
4.2 Directional Intra Prediction modes for I4MB and I16MB 52
4.3 A typical Edge Histogram for a MB 53
4.4 Performance of Mobile sequence at different values of T_{RC} 57
4.5 RD Performance for various sequence in the CIF and the QCIF resolution. 62
4.6 Inter prediction block sizes for a MB 63
4.7 Partition choices for different sequences overlaid on a frame. 65
4.8 Difference image of frame 10 and 11 of Mother and Daughter sequence . . 68
4.9 Relation between MBs in SKIP Mode and the S_{DIFF} 69

4.10 Distribution of number of MBs in different encoding modes with $S_{Amp(N)}$ for Mobile Sequence 72
4.11 Partitioning into different regions 74
4.12 Typical edge histograms for the frame difference image 75
4.13 RD Performance for various sequence in the CIF and the QCIF resolution 79

5.1 Distribution of modes for different QP 84
5.2 Relation between number of ZBs and selected mode 87
5.3 Flowchart for Level A for SKIP mode detection 91
5.4 RD Performance for various sequence in the CIF and the QCIF resolution 98
5.5 Snapshots of News:(a) Encoded by JM12.4 (b) Encoded by EZBD-FMD Algorithm 99
5.6 Snapshots of Coastguard:(a) Encoded by JM12.4 (b) Encoded by EZBD-FMD Algorithm 100
5.7 Snapshots of Stefan:(a) Encoded by JM12.4 (b) Encoded by EZBD-FMD Algorithm 101
5.8 Snapshots of Mobile:(a) Encoded by JM12.4 (b) Encoded by EZBD-FMD Algorithm 102
5.9 Neighboring MBs: C is the current MB and Values Assigned for different Modes 106

6.1 RDO Cost Computation Process 112
6.2 Transform Coefficients for I16MB and I4MB prediction for a MB in Foreman sequence (SAD difference between I16MB and I4MB mode is 20) . . . 114
6.3 Flowchart for the Fast Intra Prediction of TDFMD 117
6.4 Flowchart for the DMD of TDFMD 120
6.5 Snapshots of frames 1-4 for Mobile(QCIF): (a)-(d) Encoded with JM12.4. (e)-(h) Encoded with TDFMD 122
6.6 Snapshots of frames 1-4 for News (CIF): (a)-(d) Encoded with JM12.4. (e)-(h) Encoded with TDFMD 122

6.7 First six low frequency transform coefficients in zig-zag order 123
6.8 Observations on the first six transform coefficients 125
6.9 Final Encoding Mode Decision 127
6.10 RD curves of sequences from different classes 129

7.1 Example to demonstrate the Wavelet Decomposition of an image 133
7.2 RD Performance for various sequences in the CIF and the QCIF resolution 137
7.3 Prediction Mode Distribution (Base Layer QCIF and Enhancement Layer CIF) 140
7.4 Block Diagram for encoding and decoding of the video sequence 142
7.5 Enhancement Layer Reconstructed Frame at different QP(base) 144

8.1 Flow Chart for proposed Base Layer Prediction 150
8.2 Flow Chart for proposed Enhancement Layer Prediction 151
8.3 RD curve for (a) Crew and (b) Bus for Enhancement Layer 156
8.4 Frame 1 of CREW at different QP for BL and EL 156

List of Tables

2.1 Value of PF at different positions 26

2.2 Value of MF at different positions 27

3.1 Different Classes of Sequences 42

3.2 Value of N and W for different block sizes 46

4.1 Mode Distribution (%) at different QP 51

4.2 Results For All Intra Frame Sequences 59

4.3 Results For IPPP ... Sequences 60

4.4 Number of candidate Modes 60

4.5 Distribution of MBs (%) in different Modes at QP=32 66

4.6 Mode Selection for Different Regions 74

4.7 Performance Comparison For Different Sequences 77

5.1 Relation between Residues per MB and SKIP mode 86

5.2 Performance Comparison For Different Sequences 96

5.3 Difference Frame Weights DF_{wt} 104

5.4 Homogeneous MB Weights Hom_{wt} 105

5.5 Predicted MV Weights pmv_{wt} 106

5.6 Neighboring Mode Weights NM_{wt} 107

5.7 Final Encoding Modes for MBs 107

5.8 Performance Comparison For Different Sequences 108

5.9 % of MBs encoded in the same mode w.r.t JM12.4 109

6.1 Results for All Intra Frame Sequences 118
6.2 Performance Comparison For Different Sequences 121
6.3 Performance Comparison For Different Sequences 128

7.1 Look-up Table for High Frequency Subbands 135
7.2 Reduction In Intra and Inter RDO Calculations 136
7.3 Results For Different Sequences 137
7.4 Simulation Results for Base Layer and Enhancement Layer at different QP 143

8.1 Look-up Table for I4MB and I8MB Modes for Enhancement Layer. 152
8.2 Simulation Results at different QP 154
8.3 Simulation Results of Enhancement Layer 155

List of Acronyms

AVC	Advanced Video Coding
AZCB	All Zero Coefficient Block
BL	Base Layer
BP	Baseline Profile
CIF	Common Intermediate Format
CAVLC	Context Adaptive Variable Length Coding
CABAC	Context Adaptive Binary Arithmetic Coding
dB	Decibel
DCT	Discrete Cosine Transform
EHFMD	Edge Histogram based Fast Mode Decision
EL	Enhancement Layer
EZBD-FMD	Early Zero Block Detection based Fast Mode Decision
FMD	Fast Mode Decision
HDTV	High Definition Television
ITU-T	International Telecommunication Union-Telecommunication sector
ISO/IEC	International Organization for Standardization and International Electrotechnical Commission
I4MB	4×4 Intra Prediction Mode
I16MB	16×16 Intra Prediction Mode
JM	Joint Model
JPEG	Joint Photographic Experts Group
JSVM	Joint Scalable Video Model

JVT	Joint Video Team
MB	Macroblock
ME	Motion Estimation
MP	Main Profile
MPEG	Moving Picture Experts Group
MSE	Mean Squared Error
MV	Motion Vector
MVD	Motion Vector Difference
MV_p	Predicted Motion Vector
NAL	Network Abstraction Layer
PSNR	Peak Signal to Noise Ratio
QCIF	Quarter Common Intermediate Format
QP	Quantization Parameter
QP_{BL}	QP in Base Layer
QP_{EL}	QP in Enhancement Layer
RD	Rate Distortion
RDO	Rate Distortion Optimization
SAD	Sum of Absolute Difference
SBFMD	Subband Based Fast Mode Decision for H.264/AVC
SCFMD	Selective Coefficient Fast Mode Decision
SSD	Sum of Squared Difference
TDFMD	Transform Domain Fast Mode Decision
VBS	Variable Block Size
VBSME	Variable Block Size Motion Estimation
VCEG	Video Coding Experts Group
VCL	Video Coding Layer
VLC	Variable-length coding
WBFMD	Weight Based Fast Mode Decision

XP	Extended Profile
ZB	Zero Block

List of Symbols

$Amp(D_{i,j})$	Amplitude of Edge Vector at (i,j)
$Amg(D_{i,j})$	Direction of Edge Vector at (i,j)
$\bigotimes$	Element by element multiplication
D_{REC}	Distortion
Δ PSNR	Increase/Decrease in PSNR
Δ Rate	Increase/Decrease in Bitrate
Δ T	Increase/Decrease in Encoding time
$D_{i,j}$	Edge Vector at (i,j)
$e(\cdot)$	Residual Subblock
$E_I(\cdot)$	Integer Transform of Residual Block
$E_q(\cdot)$	Quantized Coefficient of Residual Block
$\hat{E}_{MC}$	Motion Compensated Residual Block
f_t	Frame at time 't'
$\hat{f}_t$	Reconstructed frame at time 't-1'
a≫b	Arithmetic right shift of a two's complement integer representation of 'a' by 'b' binary digits
a%b	Modulus
I	Intra Macroblock
I_{BL}	Intra BL mode in SVC
$J_{MODE}(\cdot)$	Lagrange Cost Function
λ_{MODE}	Lagrange Parameter
MB	A block of 16×16 pixels

$M(\cdot)$	Multiplying Factor
N	Neighborhood Window Size
P	Inter Predicted Macroblock
pmv	Predicted Motion Vector
pmv_x	Predicted Motion Vector in Horizontal Direction
pmv_y	Predicted Motion Vector in Vertical Direction
Qstep	Quantizer Step
R_DF	Frame Difference Residue Block
$R(\cdot)$	Rate after entropy coding
RC	Residual Complexity
S_DIFF	Sum of Residuals of R_DF
$S_{Amp(N)}$	Sum of Edge Histogram Amplitudes in a $N \times N$ block
T_1	Threshold
T_2	Threshold
T_{RC}	Threshold For Residual Complexity
$T_{th(N)}$	Homogeneity Threshold
T_S	Threshold SKIP
W_{ij}	Quantized and Scaled Coefficient
Z_{ij}	Quantized Coefficient.

1

Introduction

Contents

1.1 The Role of Standards 3
1.2 Motivation for Fast Mode Decision 5
1.3 Contributions and Overview of the Thesis 9

1. Introduction

The world has seen a dramatic development in video coding since the early 1990s. Video applications play a major role in our everyday life. The uses vary from low resolution internet images to high quality digital images, from very low bit rate videophony using mobile devices to high definition television (HDTV). The advantages of digital representation of audio, image and video are manyfold. Digital systems are less expensive, more reliable, easy to program and flexible. They are usually compatible with other digital systems. Digitized information can be transported through a noisy channel without noticeable degradation.

A digital picture/image is represented as a rectangular array of pixels with dimensions ranging from hundreds by hundreds to thousands by thousands. A sequence of digital images comprise a digital video. Video as a sequence of video frames, however, involves a huge amount of data. An uncompressed CIF (352×288) 4:2:0 video at 30 frames/sec video requires 36.5Mbits/sec and 23.3 Gbytes to store one 90-minute video. Storing this type of raw data is not practical. Visual information typically contains high statistical redundancy among the source pixels. Moreover, a human eye may tolerate or may not notice some distortion in visual signals. Video compression or video encoding is the process of reducing the amount of data required to represent a digital video signal prior to transmission or storage. Through compression, the 'raw' digital video is reduced to a manageable size for transmission or storage and then reconstructed for display. Compression involves a complementary pair of systems, a compressor (encoder) and a decompressor (decoder). The encoder converts the source data into a compressed form requiring a reduced number of bits prior to transmission or storage. The decoder converts the compressed form back into a representation of the original video data. The encoder/decoder pair is often described as a CODEC (enCOder/DECoder). Video compression algorithms operate by removing redundancy in the temporal, spatial and/or frequency domains. By removing different types of redundancy, it is possible to compress the data significantly at the expense of some information loss.

Compression of image and video has been a very active field of research for the last

two decades. Many compression algorithms have been proposed. This has led to the development of a number of international standards including JPEG, MPEG and the H.26x series of standards.

1.1 The Role of Standards

Since the early 1990s, international study groups namely the VCEG (Video Coding Experts Group) of ITU-T (International Telecommunication Union-Telecommunication sector) [4] and MPEG (Moving Picture Experts Group) [5] of ISO/IEC (International Organization for Standardization and International Electrotechnical Commission) have researched the video coding techniques for various applications of moving pictures. The ITU-T developed H.261 [6] as the first video coding standard for video conferencing application. The MPEG-1 video standard was completed in 1991 with the development of the ISO/IEC specification 11172, which is the standard for coding of moving picture and associated audio for digital storage media at up to about 1.5 Mbits/s [7]. MPEG-1 video coding standard was developed for storage in compact disk. MPEG-2 is formally referred to as ISO/IEC specification 13818, which is the second phase of MPEG video coding solution for applications not originally covered by the MPEG-1 standard. The MPEG-2 standard was completed in 1994 [8]. Its target bit rates for NTSC/PAL are about 2-15 Mbits/s and it is optimized at about 4 Mbits/s. In general, MPEG-2 can be seen as a superset of the MPEG-1 coding standard and is backward compatible to MPEG-1 standard.

For covering the very wide range of applications such as shaped regions of video objects as well as rectangular pictures, the MPEG-4 standard referred to as ISO/IEC specification 14496 was developed in 1999 [9]. This also includes natural and synthetic video/audio combinations with interactivity built in. To provide better compression of video compared to previous standards, H.264/ MPEG-4 part 10 or AVC (Advanced Video Coding) [10] video coding standard was developed by the JVT (Joint Video Team) consisting of experts from VCEG and MPEG. The final drafting work on the first version of the standard was

completed in May 2003. H.264/AVC fulfills significant coding efficiency, simple syntax specifications and seamless integration of video coding into all current protocols and multiplex architectures. Thus, H.264/AVC can support various applications like video broadcasting, video streaming and video conferencing over fixed and wireless networks and over different transport protocols. H.264/AVC provides better video at a lower bitrate and this translates into less storage requirement. H.264/AVC compression makes it possible to transmit HD television over a limited-capacity broadcast channel.

There are many technical advances in the H.264/AVC standard to reduce the bandwidth and provide good quality video. H.264/AVC incorporates more advanced intra-prediction scheme for encoding I-frames. This scheme greatly reduces the bit size of an I-frame while maintaining a high-quality image by predicting smaller blocks of pixels within each macro-block in a frame. H.264/AVC also improves block-based motion compensation used in encoding P- and B-frames. Use of variable block-size motion estimation (VBSME) with block sizes as large as 16×16 and as small as 4×4, enables precise segmentation of moving regions. It utilizes quarter-pixel precision for motion compensation, enabling precise description of the displacements of moving areas. H.264/AVC also reduces the typical blocky artifacts seen in MPEG-4 by using an in-loop de-blocking filter. Entropy coding of coefficients in H.264/AVC uses three different methods: Exp-Golomb codes, context adaptive variable length coding (CAVLC) and context adaptive binary arithmetic coding (CABAC). These methods are more efficient than the methods typically used to code coefficients in earlier designs.

H.264/AVC contains a rich set of video coding tools. All coding tools are not required for all applications. For certain applications, only a subset of the coding tools is required to implement the application. Therefore, subsets of coding tools are defined. These subsets are called Profiles. A decoder may choose to implement only one subset (Profile) of tools, or choose to implement some or all profiles. The original standard defined the Baseline Profile (BP), the Extended Profile (XP) and the Main Profile (MP). Some extensions to the original standard were later included. These extensions are known as the Fidelity

Range Extensions (FRExt). In the FRExt amendment, four profiles were defined namely High Profile (HP), High 10 Profile (Hi10P), High 4:2:2 Profile (Hi422P) and High 4:4:4 Profile (Hi444P).

1.2 Motivation for Fast Mode Decision

The H.264/AVC standard improves the coding performance significantly by adopting many advanced techniques. These improvements, however, come at the cost of increased processing power required to encode video in H.264/AVC. The standard is complex and its implementations are very challenging. The computationally expensive H.264/AVC coder can lead to slow coding and decoding times.

H.264/AVC has been developed to deliver much higher compression ratios compared to earlier video coding standards like MPEG-2. However, this higher degree of compression (up to 2-3 times more efficient than MPEG-2) comes at the expense of much higher computational requirements. This additional computational complexity is mainly due to three key techniques: Entropy encoding, smaller block size and in-loop deblocking [11].

- Entropy Encoding

  Entropy encoding is a technique used to store large amount of data by examining the frequency of patterns within it and encoding this in another smaller form. Entropy coding refers to the encoding of parameters, coefficients, or other numeric values using binary codes. H.264/AVC allows for a variety of entropy encoding schemes. The entropy coding in H.264/AVC uses universal variable-length coding (VLC) for all syntax elements except the quantized coefficients. The coefficients can be coded using either CAVLC or CABAC. The new CABAC scheme adds 5-20% to compression efficiency but is much more computationally demanding.

- Smaller Block Size Prediction

  All video codecs treat portions of the video image in blocks, often processed in isolation from one another. Independent of the number of video pixels in the image,

the number of blocks has an effect on the computational requirements. While previous standards has a fixed block size of 16 pixels on a side (referred as 16×16), H.264/AVC permits the simultaneous mixing of different block sizes (from 16×16 down to 4×4 pixels). This permits the codec to accurately define fine detail with smaller blocks while not having to waste small blocks on coarse detail. In the process considerable complexity is added in the encoding process in H.264/AVC.

- In-loop Deblocking

  De-blocking is a post-processing step that adaptively smoothes the edges between adjacent blocks. In H.264/AVC in-loop deblocking is introduced and the deblocking filter is used on the block edges. The deblocking filter is a part of both the encoding and the decoding path hence refered to as in-loop filter. Since the de-blocking is not optional for H.264/AVC, it adds significantly to the computational overhead of H.264/AVC decoder.

In addition to the increase in the complexity due to the incorporation of these techniques in the encoder, the overall coding complexity is also high. In the H.264/AVC standard, a picture is partitioned into fixed-size macroblocks (MB) each covering a rectangular picture area of 16×16 samples of the luma component and 8×8 samples of each of the two chroma components [3, 12]. Depending upon the type of frame, intra or inter, prediction is done wherein spatial redundancies are exploited in the first case and the temporal redundancies in the latter. The rate-distortion optimization (RDO) procedures are utilized in the intra and inter prediction of H.264/AVC for the purpose of selecting the optimum mode. To choose an optimal mode for a MB, the encoder calculates the RD cost for all possible combinations and chooses the mode with the minimum RD cost [13]. Transformation and quantization has to be performed for each RDO process. Thus the computational burden for such exhaustive search is very high. The runtime percentage of each function is shown in Figure. 1.1.

The runtime percentage in the motion estimation (ME) process is the highest [1]. ME

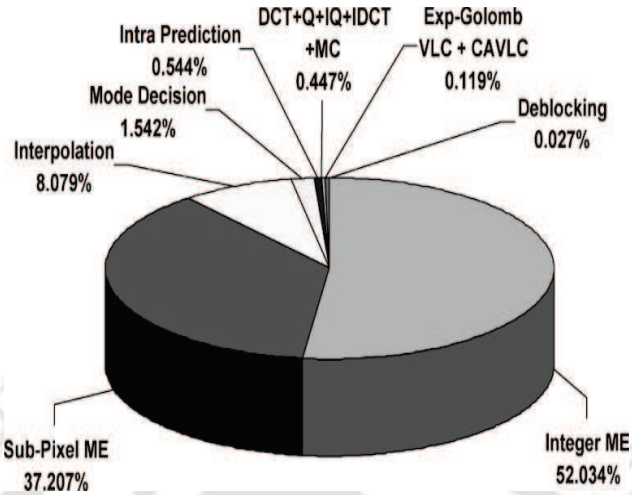


Figure 1.1: Runtime percentage of functional blocks in H.264/AVC [1]

is one of the most exhaustive video encoder process both in terms of memory requirement and computational load. The ME calculations are further complicated in H.264/AVC because of the variable block size (VBS) option. The VBS Full Motion Estimation (VBS-FME) search option allows the use of combination of block sizes ranging from 16×16 to 4×4 within a MB for luminance component and correspondingly blocks of quarter sizes for the chrominance components. The reference software of H.264/AVC, Joint Model (JM) employs the RDO technique for motion estimation. The RDO technique yield the best results in terms of visual quality and coding bit rate, however at the cost of increased complexity.

Consider a QCIF sized (176×144) video sequence to be encoded using the H.264/AVC. Each frame in this sequence will be represented by (11×9)=99 macroblocks. Each macroblock may be encoded either as a I (Intra) macroblock or as a P (Inter) macroblock .

(i) If the block is coded as a I macroblock, then it will be coded either in the 16×16 mode (I16MB) or in the 4×4 mode (I4MB). The total numbers of modes to be searched for the 16×16 mode are 4. For the 4×4 mode, the macroblock will be divided into sixteen 4×4 blocks and for each 4×4 blocks, there will be 9 modes to

be tested. Thus for the 4×4 modes the total number of modes to be searched will be $9 \times 16 = 144$ modes. Hence for the I MB, the total encoding modes to be tested are (144+4)=148.

(ii) If the block is coded as a P macroblock, ME is performed on the MB. There are seven partition sizes for the ME. There is one partition size for 16×16 block, two partitions each for 16×8 and 8×16 blocks, four partitions for 8×8 subblock, eight partitions each for 8×4 and 4×8 subblocks and sixteen partitions for 4×4 subblocks totalling to 41 possible partition sizes for a MB. Thus the ME complexity in searching all these partitions is very high. Considering the seven block sizes at the macroblock and submacroblock levels, there will be 41 predictions modes to be searched for the motion compensation. The macroblock is also tested for the intra modes. Thus for the MB total modes to be searched will be (41+148)= 189 prediction modes.

Thus for a single frame in the QCIF resolution, if encoded as an I frame the total number of prediction modes searched will be 148×99=14652. If the frame is encoded as a P frame, the total modes to be searched will be 189×99=18771. RDO will have to be performed for each mode to choose the best mode that gives the minimum rate distortion cost. Thus the encoder has a huge space of coding options. There are hundreds of possible mode combinations and thus it is necessary to find the cost for each of the mode prediction calculations before arriving at the best mode. Exhaustive search in this space is a computationally intensive process.

There are practical constraints in the implementation of the full rate distortion optimized mode selection process. Usually H.264/AVC codecs do not have the computational resources to carry out such extensive search process. This has led to the development and proposals of many low complexity coding approaches. These algorithms tend to have a poor to average rate-distortion performance. There is a tradeoff between the rate-distortion performance and the computational complexity. The goal of these algorithms is

to develop low complexity methods with minimum reduction in the rate-distortion performance. In general, the low complexity algorithms have lower rate-distortion performance than the benchmark 'full complexity' codec.

H.264/AVC achieves significant improvement in the video quality and reduced bandwidth but at the cost of greatly increased computational complexity at both the encoder and the decoder. This thesis investigates the complexity that arises from the encoding and decoding process of the H.264/AVC from full mode decision process. Detailed study of the modes is carried out and the factors upon which the final mode is decided is investigated. From the study it is found that complexity of the mode decision process is very high. An attempt has been made here to reduce the complexity and thus speeding up the encoding process. The reduction in the complexity has been investigated both in the spatial and in the transform domain.

1.3 Contributions and Overview of the Thesis

The focus of this thesis is the reduction of the encoding time and the complexity of the H.264/ AVC encoder. Throughout the work, we have made an attempt to reduce the complexity while at the same time maintain the quality and the bitrate of the video. The thesis is organized as follows:

The purpose of Chapter 2 is to give the reader a comprehensive understanding of the H.264/AVC video coding standard with a particular emphasis on the prediction process for both the intra and the inter mode decision. The latter part of the chapter presents a brief review of previous work on fast mode decision algorithms.

In Chapter 3, we describe the experimental setup for the simulations for the proposed algorithms.

In Chapter 4, we explore the possibility of using the edge characteristics of the video and its residue image for fast mode decision in both the intra and the inter mode decision process. Detailed experimental results are included to show the effectiveness of this approach for fast mode decision.

Chapter 5 focuses on the inter mode decision process. The first part of the chapter describes a fast encoding process that uses the early detection of the all zero blocks in the residue images for the fast inter mode decision process. The second part of the chapter illustrate a method of weighted prediction fast decision process based on a detailed study of the MBs and incorporates some salient features of the MBs.

The focus of Chapter 6 is to examine the mode decision process in the transform domain. This chapter describes a decision process that is based on the statistics of the transform coefficients after performing the transformation and quantization of the residues of the motion estimation process. Transform based prediction of the intra and inter process is described in this chapter.

Chapter 7 attempts to formulate a mode decision process by utilizing the subband/DCT approach. The input frame is first decomposed into four subbands. Mode decision for the whole frame is done based on the prediction of the LL subband. This approach is also applied for fast mode decision process for the SVC video coding standard. Detailed results are included to show the effectiveness of this method for both the H.264/AVC and the SVC video coding standard.

A new method is proposed in Chapter 8 for fast intra prediction for spatial scalable video coding. The mode decision in the base layer is QP dependent. Selection of the different intra modes are based on rigorous mode decision process. For the enhancement layer, the encoding information of the base layer is used to narrow down the search space of the intra modes. The results included in the chapter demonstrates the effectiveness of the proposed algorithm.

Concluding remarks, contributions of the thesis and the directions for future work are discussed in Chapter 9.

.....

2

H.264/AVC Video Coding Standard

Contents

2.1 Subjective Quality Measurement 13
2.2 Objective Quality Measurement 14
2.3 H.264/AVC Video Coding Standard 15
2.4 H.264/AVC Encoder Processes 16
2.5 Intra Prediction Process 17
2.6 Inter Prediction Process 20
2.7 Transform and Quantization 23
2.8 Low Complexity Coding: Fast Mode Decision Algorithms . . 30
2.9 Conclusions 36

Digital video can be visualized as a three-dimensional discrete valued signal. It can be represented as an ordered sequence of still images called frames. The frames are distributed along the temporal coordinate and each frame is a two-dimensional signal defined on two spatial coordinates. A video signal offers a representation for a real two-dimensional moving scene. A real world or natural video scene is typically composed of multiple objects each with its own characteristic shape, size, depth, texture and illumination. Within the video there are spatial characteristics such as texture variation within the scene, number and shape of objects, colour, etc. and temporal characteristics such as object motion, changes in illumination and movement of the camera or the viewpoint.

Video coding is a process of compressing and decompressing the digital video signal. The main aim of the video coding is to represent the scene in such a way that it should be visually similar to the original scene. The accuracy of the representation of the visual scene is measured to determine the performance of the visual communication system. Video quality analysis and evaluation of compression systems in video coding is not an easy task. There usually exists a difficulty while making a test involving human perception which varies from person to person. Two types of evaluations are done to assess its quality, a subjective quality test and an objective quality test.

2.1 Subjective Quality Measurement

This assessment of video quality is made based on human being's perceptual ability to make a non-metric decision. It is not possible for the Human Visual System (HVS) to be aware of the detailed pixel level information in a digital video. The three phenomenon described below generally tend to mask the precise information.

(i) Higher Luminance Sensitivity : Of the three color coordinates of each pixel , the luminance and the two chrominance components, the human eye is more sensitive to changes in the luminance component (which represents the brightness) compared to the two chrominance components representing the colors.

(ii) Low Pass Spatial Filtering: The human eye is not very sensitive to high spatial frequencies. Regions with rapidly changing luminance components among adjacent pixels are poorly perceived by the human eye. Thus the HVS can be represented as a low pass filter that filters the high frequencies before sending the information to the optic nerve.

(iii) Perceptual irrelevancy: An observer usually perceives a scene by fixating on a particular sequence of points rather than concentrating on the whole scene simultaneously. It does not usually take into account the exact movement and the changes in the scene. The small movements in the background usually go unnoticed as they do not carry significant information.

2.2 Objective Quality Measurement

It is a consistent and inexpensive means by which the quality of the processed video quality is measured using a standardized metric. The most widely used objective quality measure used in current video and image compression quality assessment is the Peak Signal to Noise Ratio (PSNR), referring to the peak signal power in the image and the power of the corrupting noise. The decoded and reconstructed picture quality is usually measured by its PSNR value.

The PSNR is measured on a logarithmic scale and depends on the mean squared error (MSE) between the original and an impaired image or video (in this case the decoded or reconstructed image or video) relative to $(2^n - 1)^2$, the square of the highest-possible signal value of the image and n is the number of bits per image sample.

The MSE is usually evaluated using a simple objective distortion measures such as SSD (the sum of squared difference) defined as

$$MSE = \frac{1}{M \times N} \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} [O(i,j) - O_r(i,j)]^2 \tag{2.1}$$

where O is the original image and O_r is the reconstructed image.

The PSNR is defined as,

$$PSNR = 10.\log_{10}\frac{(2^n-1)^2}{MSE}. \tag{2.2}$$

2.3 H.264/AVC Video Coding Standard

In Section 1.1, a brief history about the emergence of the H.264/AVC standard is given [13]. The H.264/AVC design covers a Video Coding Layer (VCL), which efficiently represents the video content and a Network Abstraction Layer (NAL) which formats the VCL representation of the video and provides header information in a manner appropriate for conveyance by particular transport layers or storage media. The VCL is the heart of the compression. It is a block-based hybrid video-coding system. The VCL in H.264/AVC video encoder carries out prediction, transformation and encoding processes to produce a compressed H.264 bitstream. A simplified block diagram of typical encoder processing elements for the VCL is provided in Figure 2.1. Decoding processes are conceptually a subset of these encoding processes and are shown in the shaded region of the figure.

The typical encoding operation for a picture begins with splitting the picture into blocks of samples. Each block correspond to a block of 16×16 pixels and is called a macroblock (MB). The first picture of a sequence is typically coded in intra (intra-picture) mode without using any other pictures as prediction references. The encoding process chooses which neighboring samples are to be used for intra prediction and how these samples are to be combined to form a good prediction and sends an indication of its selection to the decoder.

For the remaining pictures of a sequence or between random access points typically Inter (inter-picture) coding is utilized. Inter coding employs inter picture temporal prediction (motion compensation) using other previously decoded pictures. The encoding process for temporal prediction includes choosing the motion data that identifies the reference pictures and spatial displacement vectors that are applied to predict the samples of each block. The residual of the prediction (either Intra or Inter), which is the difference

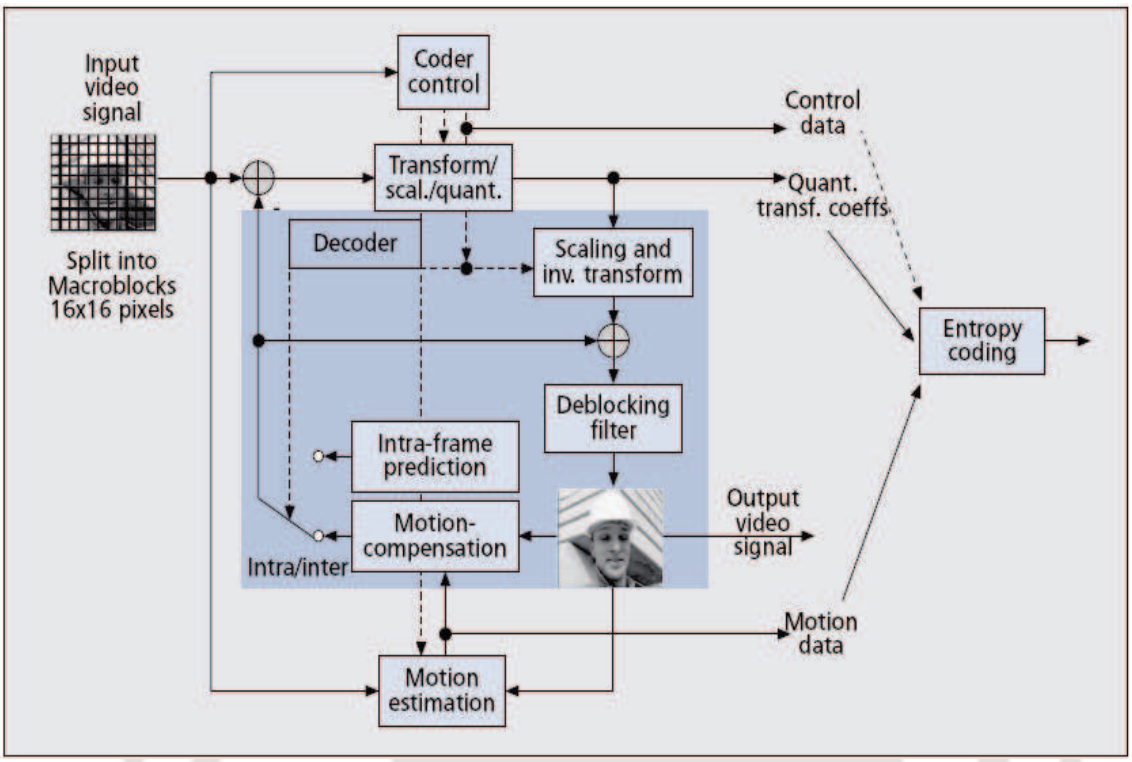


Figure 2.1: Block diagram of the encoding process for the VCL of H.264/AVC [2]

between the original input samples and the predicted samples for the block is transformed. The transform coefficients are then scaled and approximated using scalar quantization. The quantized transform coefficients are entropy coded and transmitted together with the entropy-coded prediction information for either Intra- or Inter-frame prediction.

2.4 H.264/AVC Encoder Processes

The building blocks of the H.264/AVC encoder is shown in Figure 2.1. It includes the Intra Prediction block, the motion estimation and compensation block, the de-blocking filter block, transform and quantization block and the entropy coding block [2, 11, 14–17].

2.4.1 Prediction Process

Much of the performance gain of the H.264/AVC over the previous standards is due to its efficient prediction methods [15]. Many improved prediction methods have been included in the standard which results in near accurate prediction. This in turn means

that the residual contains very little data which leads to a good compression performance. H.264/AVC supports intra prediction using data within the current frame and inter prediction using motion compensated prediction from previously coded frame and multiple reference frames. There is considerable flexibility in the prediction process for H.264/AVC. For the prediction purpose, the MBs are classified into three categories:

- I macroblock: Prediction is done using neighboring samples in the current frame.
- P macroblock: Predicted from samples in a previously-coded frame which may be before or after the current frame in the display order.
- B macroblock: predicted from samples in one or two previously-coded frames using both past and future frames.

2.5 Intra Prediction Process

Spatial redundancy within an image is the basis of intra prediction. The intra prediction uses samples from adjacent, previously coded blocks to predict the values in the current block. In an intra MB, there are three intra prediction block sizes namely 16×16, 8×8 and 4×4 for the luma component. Out of these, 8×8 block size is used for the 'High' Profiles only. A single prediction block is generated for each chroma component.

2.5.1 4×4 Intra Prediction

Each MB is partitioned into sixteen 4×4 subblocks and each 4×4 subblock is predicted from spatially neighboring samples [3]. The prediction block P for a subblock represented by pixels (a,b,. . . ,p) is calculated based on the neighboring samples labeled (A,B,. . . ,L,X) in Figure 2.2.

There are a total of 9 optional prediction modes for each 4×4 luma block. The modes are as follows:

(i) Mode 0 (Vertical): The upper samples A, B, C, D are extrapolated vertically.

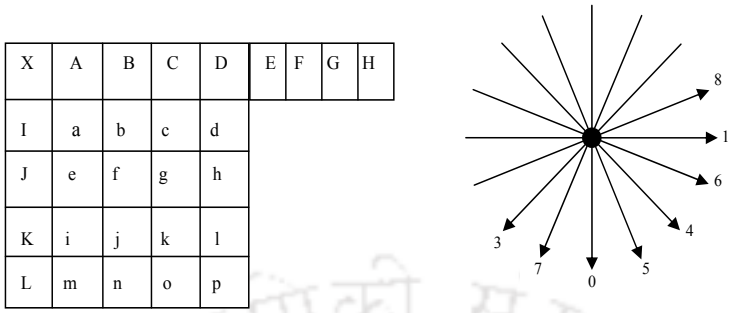


Figure 2.2: Labeling of Prediction samples and the Directions of prediction

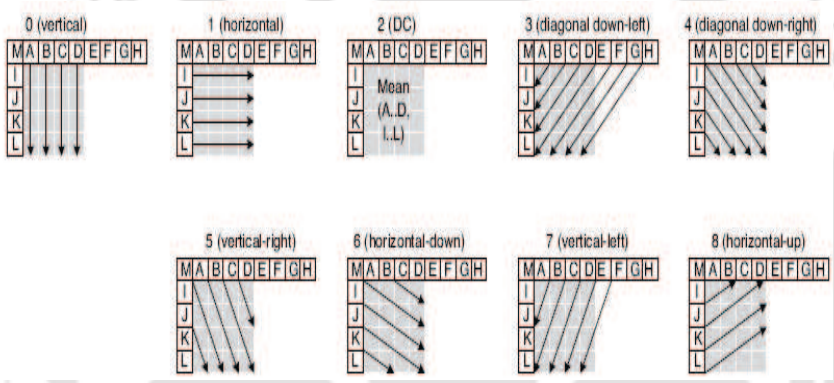


Figure 2.3: 4×4 Intra Prediction Modes [3]

(ii) Mode 1 (Horizontal): The left samples I, J, K, L are extrapolated horizontally.

(iii) Mode 2 (DC): All samples in P are predicted by the mean of samples (A ... D, I... L.)

(iv) Mode 3 (Diagonal Down Left): The samples are interpolated at a 45° angle between lower-left and upper-right.

(v) Mode 4 (Diagonal Down-Right): The samples are extrapolated at a 45° angle down and to the right.

(vi) Mode 5 (Vertical-Left): Extrapolation at an angle of approximately 26.6° to the left of vertical.

(vii) Mode 6 (Horizontal-Down): Extrapolation at an angle of approximately 26.6° below

horizontal.

(viii) Mode 7 (Vertical-Right): Extrapolation at an angle of approximately 26.6° to the right of vertical.

(ix) Mode 8 (Horizontal-Up): Interpolation at an angle of approximately 26.6° above horizontal.

Figure 2.3 shows the nine prediction modes for the 4×4 intra prediction. The arrows indicate the direction of prediction in each mode.

2.5.2 16×16 Intra Prediction Process (Luma Component)

The entire MB is predicted in one operation. Four modes are available

(i) Mode 0 (Vertical): Extrapolation from upper samples H.

(ii) Mode 1 (Horizontal): Extrapolation from left samples V.

(iii) Mode 2 (DC): Mean of upper and left samples.

(iv) Mode 3 (Plane): A linear plane function is fitted to the upper and left-hand samples H and V.

Figure 2.4 shows the four prediction modes for the 16×16 intra prediction.

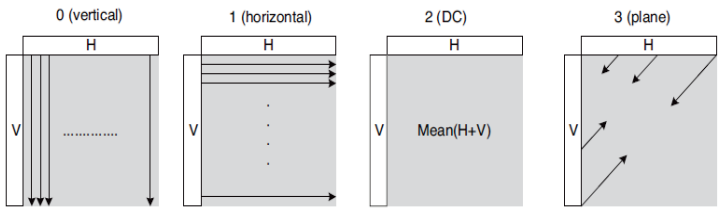


Figure 2.4: 16×16 Intra Prediction Modes [3]

2.5.3 Chroma Prediction Modes

For the chroma component of a MB, the prediction is done from the previously encoded chroma samples above and/or to the left. Both the chroma components always use the same prediction mode. The chroma prediction is defined for three possible block sizes, 8×8 chroma in 4:2:0 format, 8×16 chroma in 4:2:2 format and 16×16 chroma in 4:4:4 format. The prediction modes for the chroma components are very similar to the four modes of the 16×16 prediction. Only the naming is different. The modes are DC (mode 0), horizontal (mode 1), vertical (mode 2) and plane (mode 3).

2.6 Inter Prediction Process

Inter-prediction reduces the temporal correlation between frames with the help of motion estimation and compensation [15]. A block of luma and chroma component is predicted from a frame called the reference frame that has been previously coded and transmitted. The reference frame is chosen from a list of previously coded pictures. In H.264/AVC encoder the current picture samples are predicted and subtracted from the original samples to form a residual which is then coded and transmitted. The salient features of the H.264/AVC inter prediction are as follows:

(i) Variable block sizes for motion compensation,

(ii) Multiframe references for prediction,

(iii) Generalized B frame prediction,

(iv) Use of B frames as references,

(v) Weighted prediction,

(vi) Fractional pixel accuracy for motion vectors.

These extensions in H.264/AVC improve the coding performance while increasing the complexity substantially [17]. The current picture can be partitioned into a complete MB

or smaller blocks called sub-macroblocks. A MB of 16×16 luma samples can be partitioned into sub-macroblock of sizes up to 4 × 4. The macroblock partitioning correspond to the luma region block sizes of 16 × 16, 16 × 8, 8 × 16, and 8 × 8 whereas the sub-macroblock partitioning have block sizes of 8 × 8, 8 × 4, 4 × 8, or 4 × 4 samples. Sub-macroblock partition is possible only when the macroblock partition size is 8 × 8. Figure 2.5 shows the partitioning of a 16 × 16 MB.

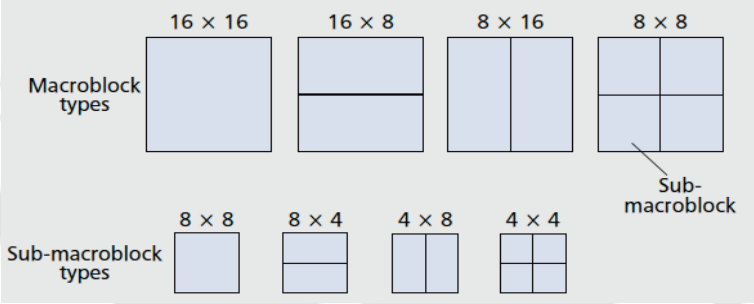


Figure 2.5: Partitioning of a macroblock and a sub-macroblock for motion-compensated prediction [2]

The inter prediction process involves choosing a prediction block in the reference picture and subtracting this from the original block. The offset between the position of the current partition and the prediction block in the reference picture is a motion vector (MV). The prediction block may be generated from a single prediction region in a reference picture for a P or B macroblock or from two prediction regions in reference pictures for a B macroblock.

H.264/AVC uses MV accuracy of one-quarter of the luma sample. Sub-pel motion compensation can provide significantly better compression performance than integer-pel compensation but at the expense of increased complexity. The sub-pel accuracy increases the coding efficiency at high bitrates and high video resolutions. Quarter-pel accuracy outperforms half-pel accuracy. Since the sub-pixel luma and chroma samples are not available in the reference pictures, these are generated by interpolating the neighboring integer pel samples. The half-pel samples are first created using a 6-tap FIR filter with weights (1, -5, 20, 20, -5, 1)/32. Once all the half-pel samples are available, each quarter-pel sample

is produced using bilinear interpolation between neighboring half- or integer-pel samples. For 4:2:0 video source sampling, 1/8 pel samples are required in the chroma components (corresponding to 1/4 pel samples in the luma). These samples are interpolated between integer-pel chroma samples using linear interpolation.

2.6.1 SKIP mode

In addition to the encoding modes described above, a P MB can also be coded in the so-called SKIP mode. If a MB has motion characteristics that allows its motion to be effectively predicted from the motion of neighboring MBs and further, if it contains all zero quantized transform coefficients, then it is flagged as skipped and is identified as the SKIP mode. For this mode, neither a quantized prediction error signal nor a motion vector or reference index parameter has to be transmitted. Only one bit per MB has to be transmitted to signal a skipped MB. In general, the motion vector used for reconstructing the SKIP MB is identical to the motion vector predictor for the 16×16 block. However, under special conditions a zero motion vector is used instead.

2.6.2 Motion Vector Prediction

The number of MVs required for a given partition size increases with the reduction in the partition size. For example, only two MVs are required for signaling a 16×16 partition whereas 32 MVs are required for a 4×4 partition size. Thus for smaller partition size a large number of bits are required to signal the choice of MVs to the decoder. When a video object moves, the neighboring objects also move along the same direction. MVs of neighboring partitions are often highly correlated. Each MV is thus predicted from the nearby previously coded partitions. A predicted motion vector MV_p is formed based on previously calculated MVs. The motion vector difference, MVD, between the current vector and the MV_p is encoded and transmitted. The method of forming the MV_p depends upon motion compensation and the availability of the nearby MVs.

Let E in Figure 2.6 be the current MB, MB partition or sub- macroblock partition.

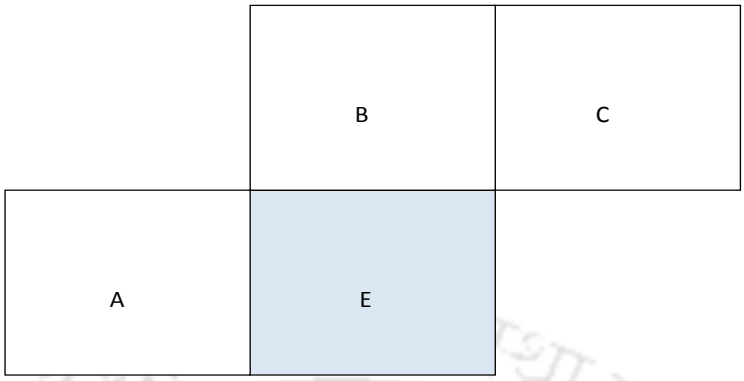


Figure 2.6: Current and neighbouring partitions

Let the MB or sub- macroblock partition immediately to the left of E be A, immediately above E be B and above and to the right of E be C. If there are more than one partition to the left of E, only the uppermost partition is chosen as A. In case there are more than one partition above E, only the leftmost is chosen as B. The predicted motion vector is obtained as follows [12]:

(i) for transmitted partitions excluding 16×8 and 8×16 partition sizes, MV_p is the median of motion vectors for partitions A, B and C,

(ii) for 16×8 partitions, MV_p for the upper 16×8 partition is predicted from B and for the lower 16×8 partition is predicted from A,

(iii) for 8×16 partitions, MV_p for the left 8×16 partition is predicted from A and for the right 8×16 partition is predicted from C,

(iv) for skipped macroblocks, a 16×16 MV_p is generated as if the block were encoded in the16×16 inter mode.

2.7 Transform and Quantization

H.264/AVC uses transform coding of the prediction residual. The transform it uses, however, is an integer transform instead of the 8×8 discrete cosine transform (DCT)

traditionally used in many video coding standards [3, 15]. Transformation using integer transform involves only additions and shift operations and no mismatch exists between the forward and inverse transforms.

H.264/AVC uses an adaptive transform block size of 4×4. Transform block size of 8×8 is used in High Profiles only. The smaller block size leads to a significant reduction in ringing artifacts. The "baseline" profile of H.264/AVC uses three transforms depending on the type of residual data that is to be coded. A default transform process is available wherein each 4×4 block within a luma MB is transformed, scaled and quantized to produce a block of 4×4 quantized coefficients. If the prediction type for the MB is 16×16 intra prediction, a second transform is applied to the DC coefficients of the first transform. The DC coefficients of each 4×4 blocks are taken to form a 4×4 DC coefficient block. The standard specifies the 4×4 Hadamard transform for this block of luma DC coefficients. The transformed DC coefficient block and the remaining AC coefficients are scaled, quantized and then transmitted. Each of the 4×4 block of Cb or Cr samples is transformed. The four DC coefficients are transformed using 2×2 Hadamard transform. The transformed DC coefficient block and the remaining AC coefficients are scaled, quantized and transmitted.

2.7.1 Forward Transform

Let X be a block of 4×4 residuals [3]. The 4×4 two dimensional DCT on the block is given by

$$Y = A.X.A^T \tag{2.3}$$

where

$$A = \begin{bmatrix} a & a & a & a \\ b & c & -c & -b \\ a & -a & -a & a \\ c & -b & b & -c \end{bmatrix} \tag{2.4}$$

Here a=1/2, $b = \sqrt{\frac{1}{2}}\cos\frac{\pi}{8}$ and $c = \sqrt{\frac{1}{2}}\cos\frac{3\pi}{8} = bd$ with $d = \sqrt{2} - 1 \approx 0.414$

To simplify the implementation of the transform, d is approximated by 0.5. For ensuring orthogonality, b also needs to be modified so that $b = \sqrt{\frac{2}{5}}$.

The matrix multiplication can be factorized into the following equivalent form

$$Y = (CXC^T) \otimes E$$

$$Y = \left(\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & d & -d & -1 \\ 1 & -1 & -1 & 1 \\ d & -1 & 1 & -d \end{bmatrix} \begin{bmatrix} & & & \\ & X & & \\ & & & \\ & & & \end{bmatrix} \begin{bmatrix} 1 & 1 & 1 & d \\ 1 & d & -1 & -1 \\ 1 & -d & -1 & 1 \\ 1 & -1 & 1 & -d \end{bmatrix} \right) \otimes \begin{bmatrix} a^2 & ab & a^2 & ab \\ ab & b^2 & ab & b^2 \\ a^2 & ab & a^2 & ab \\ ab & b^2 & ab & b^2 \end{bmatrix} \quad (2.5)$$

E is a matrix of scaling factors and the symbol $\bigotimes$ indicates that element by element multiplication of CXC^T and the scaling matrix E. The final forward transform becomes

$$Y = ((C_f X C_f^T) \otimes E_f$$

$$Y = \left(\begin{bmatrix} 1 & 1 & 1 & 1 \\ 2 & 1 & -1 & -2 \\ 1 & -1 & -1 & 1 \\ 1 & -2 & 2 & -1 \end{bmatrix} \begin{bmatrix} & & & \\ & X & & \\ & & & \\ & & & \end{bmatrix} \begin{bmatrix} 1 & 2 & 1 & 1 \\ 1 & 1 & -1 & -2 \\ 1 & -1 & -1 & 2 \\ 1 & -2 & 1 & -1 \end{bmatrix} \right) \otimes \begin{bmatrix} a^2 & \frac{ab}{2} & a^2 & \frac{ab}{2} \\ \frac{ab}{2} & \frac{b^2}{4} & \frac{ab}{2} & \frac{b^2}{4} \\ a^2 & \frac{ab}{2} & a^2 & \frac{ab}{2} \\ \frac{ab}{2} & \frac{b^2}{4} & \frac{ab}{2} & \frac{b^2}{4} \end{bmatrix} \quad (2.6)$$

2.7.2 Quantization

H.264/AVC uses a scalar quantizer. The parameter can take 52 values. These values are arranged so that an increase of 1 in quantization parameter means an increase

of quantization step size by approximately 12% (an increase of 6 means an increase of quantization step size by exactly a factor of 2).

The basic forward quantizer operation is as follows:

$$Z_{ij} = round(Y_{ij}/Qstep) \tag{2.7}$$

where Y_{ij} is the coefficient of the forward transform, Qstep is the quantizer step size and Z_{ij} is the quantized coefficient.

The post-scaling factor $a^2, ab/2$ and $b^2/4$ is incorporated into the forward quantizer. First, the input block X is transformed to give a block of unscaled coefficients $W = CXC^T$. Then, each coefficient W_{ij} is quantized and scaled in a single operation:

$$Z_{ij} = round\left(W_{ij}.\frac{PF}{Qstep}\right) \tag{2.8}$$

where PF is either $a^2, ab/2$ or $b^2/4$ depending on the position (i,j) as given in Table 2.1

Table 2.1: Value of PF at different positions

Position	PF
(0,0),(2,0),(0,2) or (2,2)	a^2
(1,1),(1,3),(3,1) or (3,3)	$b^2/4$
others	$ab/2$

The factor (PF/Qstep) is implemented in the H.264/AVC reference model software as a multiplication by MF (a multiplication factor) and a right-shift, thus avoiding any division operations.

$$Z_{ij} = round\left(W_{ij}.\frac{MF}{2^{qbits}}\right), \tag{2.9}$$

where

$$qbits = 15 + floor(QP/6).$$

In integer arithmetic, the above can be implemented as

$$|Z_{ij}| = sign(W_{ij})[(|W_{ij}|\,.MF + f) \gg qbits)], \tag{2.10}$$

where $\gg$ indicates binary right shift. In the reference software, f is $2^{qbits}/3$ for intra blocks and $2^{qbits}/6$ for inter blocks.

The first 6 values of MF depending on QP and the coefficient position (i,j) are given as:

Table 2.2: Value of MF at different positions

QP	Position (0,0),(2,0),(0,2),(2,2)	Position (1,1),(1,3),(3,1),(3,3)	Position other
0	13107	5243	8066
1	11916	4660	7490
2	10082	4194	6554
3	9362	3647	5825
4	9192	3355	5243
5	7282	2893	4559

2.7.3 Entropy Coding

H.264/AVC supports two different types of coding methods. For the syntax elements, a simpler entropy coding method is used. Here, a single infinite-extent codeword table is used for all syntax elements except the quantized transform coefficients. The mapping to the single codeword table is customized according to the data statistics. The single codeword table chosen is an exp-Golomb code with very simple and regular decoding properties.

For transmitting the quantized transform coefficients, the efficient method called CAVLC is employed. CAVLC uses variable length coding scheme and codes the transformed coefficients by exploiting inter symbol redundancy. In this scheme, VLC tables for various syntax elements are switched depending on the already transmitted syntax elements. After transform and quantization, the probability that the level of coefficients is zero or ± 1 is very high. CAVLC handles the zero and ± 1 coefficients in a manner different from the levels (sign and magnitude) of the non-zero coefficients.

The efficiency of entropy coding can be improved further if CABAC is used [18]. The arithmetic coding scheme achieves good compression performance through probability models selection for each syntax element according to the element's context, adapting

probability estimates based on local characteristics and using arithmetic coding [3]. The coding system involves the following stages:

(i) Binarization: It is a type of preprocessing unit applied before the actual encoding. Here the non binary valued symbols like the transform coefficients, motion vectors etc. are converted to binary code prior to arithmetic coding. This process is similar to the process of converting a data symbol into a variable length code but the binary code is further encoded by the arithmetic coder prior to transmission.

(ii) Context Model Selection: A probability model is chosen from a selection of available models depending upon the statistics of recently coded data symbols. The context model stores the probability of each bit being '1' or '0'.

(iii) Arithmetic Coding: This coder encodes each bin according to the selected probability model.

(iv) Probability Model: The selected context model is updated based on the actual coded value.

CABAC typically provides a reduction in bit rate between 5%-15% compared to CAVLC. The highest gains are obtained when coding interlaced TV signals.

2.7.4 Rate Distortion Optimization

Lossy compression results in higher compression ratios but at the cost of imperfect source representation. The trade-off between source fidelity and coding rate is the rate distortion optimization (RDO) problem. How much fidelity of the source should be given up in order to reduce the bits required for encoding is the main idea of the RDO process. The goal of the RDO is to optimize the overall fidelity so as to minimize distortion subjected to a constraint rate. Rate-distortion optimization requires an ability to measure distortion. RDO is primarily used by video encoders to improve quality in any encoding situation (image, video, audio or otherwise) where decisions have to be made that affect both file size and quality simultaneously.

The most popular approach for optimization in video is the Lagrangian optimization techniques. Since there are a number of modes for the INTRA and the INTER prediction process, the RDO procedure is required to select the best prediction mode that gives the minimum distortion at the lowest rate. Consequently, the coding mode for each block is determined using the Lagrangian cost function given by J [19, 20]. Let the Lagrange parameter (λ_{MODE}) and the quantizer value QP be given. The Lagrangian mode decision for a MB S_k is calculated using the Lagrange's function by minimizing the following cost function

$$\mathrm{J_{MODE}(S_k, MODE|QP, \lambda_{MODE}) = D_{REC}(S_k, MODE|QP) + \lambda_{MODE} R(S_k, MODE|QP)} \tag{2.11}$$

For the intra mode, the distortion $D_{REC}(S_k, MODE|QP)$ is measured as the sum of the squared differences (SSD) between the reconstructed and the original MB pixels.

$$SSD = \sum_{(x,y)\in A} |s[x,y,t] - s'[x,y,t]|^2, \tag{2.12}$$

where A is the subject MB. The rate $R(S_k, MODE|QP)$ is the rate that results after entropy coding.

The computation of the Lagrangian costs for the INTER modes is much more demanding. Given the Lagrange parameter (λ_{MOTION}) and the decoded reference picture s', rate-constrained motion estimation for a block S_i is performed by minimizing the Lagrangian cost function

$$m_i = \arg\min_{m\in M}\{D_{DFD}(S_i, m) + \lambda_{MOTION} R_{MOTION}(S_i, m)\} \tag{2.13}$$

where M is the set of possible coding modes and D_{DFD} is the distortion term given by

$$D_{DFD} = \sum_{(x,y)\in A_i} |s[x,y,t] - s'[x - m_x, y - m_y, t - m_t]|^p \tag{2.14}$$

with p=1 for the sum of absolute difference (SAD) and p=2 for the SSD. $R_{MOTION}(S_i, m)$ is the number of bits to transmit all components of the motion vector (m_x, m_y). Further,

λ_{MODE} is the Lagrangian constraining parameter which is a function of the QP and is given as,

$$\lambda_{MODE} = 0.85 \times 2^{(QP-12)/3}. \tag{2.15}$$

When SAD is used, $\lambda_{MOTION} = \sqrt{\lambda_{MODE}}$ and when SSD is used $\lambda_{MOTION} = \lambda_{MODE}$ The mode which gives the lowest value of the cost function J_{MODE} is the selected mode for encoding the video.

2.8 Low Complexity Coding: Fast Mode Decision Algorithms

The H.264/AVC standard supports a number of coding modes and options. There is a significant gain in the compression efficiency over previous standards due to different techniques used in the standard. The use of various advanced coding tools and mode selection process eventually increase the computational complexity of the encoder. Therefore, low-complexity algorithms have been proposed to reduce the computational complexity. The common approach in such fast algorithms is to reduce the number of modes to be searched.

Motion estimation and the mode decision in the H.264/AVC standard is the most time consuming and computationally intensive process. Hence extensive research efforts have gone into reducing the complexity of the intra and the inter mode decision process.

Video sequences consist of a collection of frames in the temporal direction. Each frame in the video have regions with similar pixel values, that is, they are homogeneous. There are regions with complex texture and varying illumination. They have different objects in them that have different orientations. Movement of objects between frames can be continuous motion in a direction, there may be new objects coming into the frame. The objects take different positions in different frame. When seen as a sequence, it gives the sense of the scene's dynamics. The frames have lot of similarities and these inherent properties of video sequences are exploited for the fast encoding approaches.

2.8.1 Fast Intra Prediction Algorithms: A Review

Intra mode decision process involves the estimation of the RD cost for different modes and then selecting one final mode. The calculation of all the modes are wasteful of the computational resource of the encoder. In this regard a number of low complexity algorithms have been proposed in literature [21–41]. A fast intra coding algorithm was proposed in [21, 22] where the modes to be used for encoding was decided based on the local edge information obtained from the edge direction histogram for a MB. Some edge information based fast algorithms are also discussed in [23–26]. In [25], the dominant edges in a MB are detected before performing the actual RDO technique. Five edges were defined namely the vertical, horizontal, 45° diagonal, 135° diagonal and the non-directional edges. These edges are found using the spatial domain filtering operations. The dominant edge is the one with the maximum value of the edge histogram amplitude. A few mode out of all the modes are chosen for the encoding depending upon the dominant edges while others are ignored.

Low complexity algorithms in [27, 28] use spatial correlation with the adjacent modes. It uses simple directional masks and neighboring modes for complexity reduction. For the 4×4 intra prediction , the authors observed that the directional correlation of a block is generally consistent with edge directions. Secondly, the prediction mode of current block is highly correlated with the modes of adjacent blocks. Some modes are chosen depending upon the direction of the edge in the block. In [29] a method was proposed for both chroma and luma intra prediction. Large size is chosen for homogeneous region. Normalized SAD statistics are used for the smaller size prediction.

The RD cost is calculated for each mode before deciding on the final mode. Therefore, many rate control fast encoding algorithms have been proposed for intra prediction [30–35]. In [30], a novel rate control algorithm for low complexity mobile applications was proposed. The residual complexity of the intra pictures are determined without performing the intra prediction. The local features and textures in the frame are exploited for finding the complexity that will remain when intra prediction is done. RDO cost based

intra prediction was proposed in [31] where only four modes of the 4×4 block are calculated. Based on the RDO cost of these modes and its correlation with neighboring modes and the fact that the dominating direction of a smaller block is similar to that of a larger block, the candidate modes of 8x8 blocks and 16x16 macroblocks are determined. The authors in [32] proposed a new rate enhanced rate-distortion cost function and combining it with the sum of absolute integer-transformed differences (SAITD) and a rate predictor the modes for the intra prediction are determined. This method gives a faster means for calculating the SAITD which scores over the total computations required for the sum of absolute Hadamard-transformed differences (SATD) that is used in the reference software.

In [36], a method was proposed for efficient intra prediction using template matching. A sample predictor is created by template matching. This is used as an additional mode for the encoder. The complexity at the decoder increases since the template matchig operation needs to be performed. A intra luma prediction algorithm using collocated weighted chroma pixel value is proposed in [37]. The luma blocks are predicted as a multiplication of a weight calculated from the collocated chroma and luma pixels and the upsampled luma pixels. This is used as an additional intra prediction mode. This works best with mid-range bitrate. Authors in [38] proposed a method wherein spatial domain and transform domain features are extracted from the target block. SAD and SATD thresholds are defined and the modes that give SAD or SATD values below the thresholds are selected for encoding. This method improves the encoding time by around 10%.

Many fast mode decision algorithms are also available in literature for the scalable extension of the H.264/AVC standard. Fast inter layer intra prediction techniques are proposed for the SVC standard in [25,35,42–46]. In [35] a fast mode decision algorithm is proposed. It is based on macroblock-adaptive rate-distortion (R-D) estimation for intra-only scalable video coding (SVC). The log-linear R-D relationship of inter-dependent layers are used to predict the better performer among the Intra4x4 and Intra8x8 prediction types at the enhancement layers. [42] proposes an efficient scalable intra coding

scheme with spatial scalability. For the intra prediction in enhancement layer, a prediction method is proposed that uses information from both the base layer and the enhancement layer. In [43] the mode distribution relationship between base layer and enhancement layers is employed to reduce the candidate mode set at enhancement layers. [46] proposes an intra frame dyadic spatial scalable coding framework based on a subband/wavelet coding approach for MPEG-4 AVC/H.264 scalable video coding (SVC). An attempt has been made here to join the subband filter banks with the traditional macroblock and DCT based video coding system. This method can efficiently work together with the traditional subband filter banks for providing improved efficiency for intra-frame dyadic spatial scalable coding.

2.8.2 Fast Inter Prediction Algorithms: A Review

Motion estimation and mode decision are the most computation intensive part of inter frame coding. It is evident that the percentage of runtime taken by the inter mode decision is the highest amongst all the encoding processes as shown is Figure. 1.1. Hence there have been extensive studies on ways and means to reduce the computational complexity involved in ME and mode decision. Researchers employ different methods for reducing the complexity of inter frame prediction. One approach for fast ME is to limit the number of modes to be searched. Based on certain criteria these algorithms exclude some modes to speed up the encoding process while trying to maintain similar coding efficiency obtained from full motion search of the H.264/AVC reference software.

Detecting SKIP mode early and save unnecessary mode calculations is one way to reduce the encoder complexity. Early prediction of SKIP mode employs early detection algorithms to estimate the number of ZBs in a MB. These algorithms try to predict the SKIP mode beforehand and save unnecessary mode calculations. This method is appealing as the computation of the SKIP mode is very simple as compared to the other modes. There are homogeneity based approach where choice of modes depends upon some homogeneity measures. Fast mode decision using homogeneity of the MBs examines not

only the SKIP mode but also considers other modes for ME. The mode decision process based on this approach give better coding and compression efficiency for a wider range of video sequences. This is due to the detailed study of the associated modes at different levels of computation.

Many fast inter prediction algorithms have been reported in literature [1, 47–96]. In [47], a hybrid Unsymmetrical-cross Multi-Hexagon-grid Search (UMHexagonS) algorithm was proposed for integer pel motion estimation in H.264/AVC using the hybrid and hierarchical motion search strategies. Based on a common agreed conclusion that the movement in the horizontal direction is much heavier than that in the vertical direction for natural picture sequences, the optimum motion vector can be nearly accurately predicted by an unsymmetrical-cross search. There was a large reduction in the number of points to be searched for the motion estimation which resulted in a large saving in the computation time keeping distortion minimum. In [52], a method of mode selection was described based on the RDO performance. For a P or a B slice, early SKIP detection is performed. If SKIP conditions are satisfied then the MB is coded in the SKIP mode. Otherwise the inter mode prediction process is performed. The choice of the intra mode decision for the P and B frames depends upon the spatial correlation between pixels at a boundary of the current and its adjacent upper and left encoded blocks under the best inter mode. Two parameters are defined: Average Rate (AR) for the best inter mode and the Average Boundary Error (ABE). If AR is more than ABE, then intra modes are to be tested for the best mode, otherwise the algorithms skips the intra modes in the prediction process.

[87] proposed a homogeneity based fast mode selection algorithm. The spatial homogeneity and the temporal stationarity is taken into consideration for the mode selection process. The spatial homogeneity is decided by the edge intensity of the MB. The homogeneity of a $N \times N$ block where N=16 or N=8, the homogeneity is determined by the sum of magnitude of the edges at all pixels in this block. If the sum is below a particular threshold defined, the block is considered homogeneous otherwise it is split for further

inter mode search. The temporal stationarity is based on the pixel differences of the MB under consideration and the collocated MB in the previous frame. Based on these conditions the modes to be selected are decided. In [96], a relative homogeneous concept is adopted to reduce the inter mode search. It is shown here that the SAD value of the image block has strong correlations with the sum of the edge amplitudes of all pixels in the block and the current quantization interval. Hence content based early termination thresholds are developed for the motion estimation process.

[76] proposes a mode selection method prior to actual encoding. In this work the authors develop simple optimizations for the mode selection which is done by minimizing the distortion with the motion vectors and the header information. A high proportion of the computation effort goes in the calculation of the number of bits required for the DCT coefficients. The authors propose a modified cost function which takes into account the distortion and the bits used for the header information of the motion vectors and the mode selection. The Lagrange multiplier is also modified and the burden of computation of the bits for the residual information is saved. The work in [80] describes a hierarchical structure for mode decision comprising of three levels. Early termination can be effected in any of the levels to avoid full Lagrangian estimation. The first level carries out early detection of SKIP modes by determining whether a neighboring MB is encoded in the SKIP mode. If true, then the temporal stationarity is tested based on a threshold derived from the SAD values of the block and the neighboring blocks. If below the threshold, then the MB is encoded in the SKIP mode and the prediction process is terminated early. If the above is not true, then the MB enters the second level where it is tested for spatial homogeneity. The complexity of the MB is determined by comparing the variance of the MB with the possible maximum variance. If found homogenous, then the larger block size inter modes are activated. If this condition is also not satisfied, then the smaller inter block sizes are activated in the third level. This method gives a large saving in the encoding time.

All-zero blocks(AZB) of the DCT is very common in very low bit rate encoding. If

somehow the AZB can be detected prior to performing the DCT and quatization, the number of computations needed to do the DCT and the quantization will greatly reduce. Thus, early detection of AZB in very important. In [97] an early detection method for AZB was proposed. The authors defined a sufficient condition for quantizing all DCT coefficients to zero without actually computing the coefficients. This algorithm was further improved in [82] where the author theoretically derived a precise condition for detecting the AZB. The authors in [83] derive a sufficient condition under which each quantized coefficients becomes zero. A precise sufficient condition is then proposed in which thresholds are defined which depend upon the various frequency components of the coefficients. This algorithms shows better performance than the method in [82].

The work in [66] proposed an efficient zero block decision algorithm using early zero block detection and applied this method for both intra and inter mode decision process. The authors noted that for very low bit rate encoding or high QP, most of the MBs tend to get encoded in the SKIP mode or with larger block sizes. The number of zero blocks in a MB is determined by using the early zero block detection algorithm. Depending upon the number of zero blocks in a MB, decision is taken on the encoding mode to be utilized for motion compensation. They also derive a simple near sufficient condition for early zero block detection for enhancing the performance of the zero detection algorithm. Splitting and merging algorithms are used by the authors in the semi-stationary and the non stationary MBs instead of the full search. This algorithm resulted in a large saving in the encoding time.

2.9 Conclusions

The H.264/AVC video coding standard can deliver significantly improved compression efficiency compared with previous standards. It gives better quality at same bitrate or similar quality at reduced bitrate. Due to its high compression efficiency, it has wide range of applications from mobile phone services to HDTV transmission. However, with performance improvements, the computational cost also becomes significantly higher. Thus,

there are practical constraints in the implementation of the encoder for real time applications. This calls for the development of low complexity algorithms for the encoding process.

Many low-complexity algorithms have been developed. The main hypothesis behind these algorithms is that the computational complexity of video compression is to be reduced without loss in the quality and bitrate consideration. The main focus of these algorithms is to reduce the computations for the intra and inter prediction process. After a study of these algorithms the following observations can be made:

(i) The fast encoding algorithms are typically based on certain attributes and activity of the video sequence.

(ii) These algorithms are mostly threshold based where the threshold selection is based on trial and error methods.

(iii) The algorithms gives varying results for sequences with different activity levels.

(iv) The results usually degrade with the increase in the activity level of the video sequence

The main focus of this work is to develop low complexity algorithms while maintaining original rate-distortion performance. The following chapters describe the main contributions of this research in relation to the primary objective. The next chapter (Chapter 3) presents the experimental method used in evaluation of various algorithms proposed in this work.

3

Experimental Method

Contents

3.1 Introduction 40
3.2 Implementations 40
3.3 Test Video Sequences Used for Experiments 41
3.4 Performance Evaluation 43
3.5 Conclusion 48

3.1 Introduction

The implementation and the procedure for performance evaluation and comparison of various algorithms considered in this work are described in this chapter. The test video sequences used and the methods for the performance evaluation is detailed in the following sections.

3.2 Implementations

The algorithms are implemented and tested by software simulation. The standard JM Reference Software (JM 12.4) [98] is used for the implementation.

3.2.1 Reference Software

For the proposed algorithms and those referred for comparison , the H.264/AVC Reference software codec JM12.4 is used for the experiments. The performance of the JM reference software is used as the benchmark for comparison of the algorithms. This helps in a fair comparison between different low complexity algorithms. The source code for the software can be downloaded from http://iphome.hhi.de/suehring/tml/download/ [98]. The reference software gives the essential statistics like the PSNR, the bitrate and the encoding time.

3.2.2 Test Platform

A personal computer has been used for the implementations of the algorithms. The specifications are given as follows:

Processor: Intel (R) Core(TM)2Duo @ 3.00 GHz

Memory (RAM): 2 GB

Operating System: Windows XP

3.2.3 Implementation and Test of the Algorithms

Figure 3.1 shows the procedure adopted to test the algorithms developed. The reference software is modified and the proposed method is incorporated into it. The test sequences are encoded using both the original reference software and the modified software. The complexity, quality and the bitrate performance obtained from both are used for the comparison purposes.

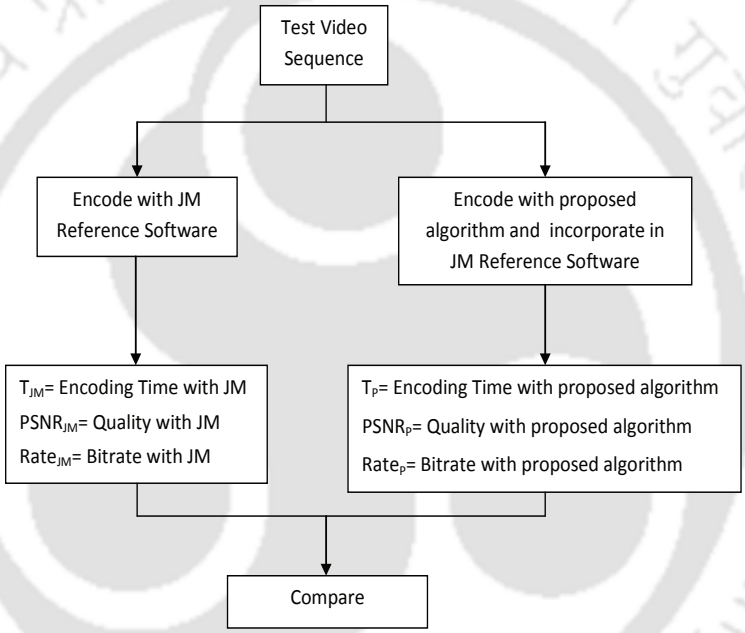


Figure 3.1: Test Scenario For the Implementations

3.3 Test Video Sequences Used for Experiments

The proposed low complexity algorithms are tested on some commonly used test video sequences. These test sequences exhibit different complexity levels, varying foreground and background information, different motion contents, texture and camera movements. The test sequences can be downloaded from http://trace.eas.asu.edu/yuv/.

Table 3.1: Different Classes of Sequences

Type	CIF Sequence	QCIF Sequence
Class A	News, Mother and daughter(MaD) Container, Hallmonitor	Suzie, Claire Missamerica
ClassB	Foreman, Coastguard Harbour, Ice	Foreman, Silent Crew
Class C	Mobile, Flower Tempete, Stefan	Mobile, Football Soccer

3.3.1 Classification of Sequences

Real video sequences have different characteristics in terms of texture, changing scenes, motion complexity, lighting changes, camera motion , etc. Different sequences have different motion complexity and thus they show different mode distribution statistics. The characteristics of some sequences according to the type of motion, spatial complexity and camera movement is given in [19]. After studying the motion content in various test video sequences, the different video sequences were grouped into three classes having different motion complexity. Table 3.1 shows video sequences classified into three classes: Class A having low and simple motion, Class B having medium to high motion and Class C having high motion complexity.

The characteristics of some of the test video sequences are given below:

(i) **News**: This video clip shows two newsreader talking to the camera in a studio environment. The video is shot with a still camera with very little motion of the newsreader. In a small area in the the background, there is couple performing ballet dance with moderate motion.

(ii) **Mother and Daughter**: In this clip, a woman and a child are sitting in a room. The woman is talking into a still camera and stroking the child's hair. There is low motion content in this sequence.

(iii) **Container**: In this scene, there is a container ship slowly moving on the water. The camera is still and the motion in this clip is slow.

(iv) **Foreman**: This clip shows a construction worker talking into a hand held camera with lot of hand gestures and facial expressions. Towards the end, the camera suddenly moves away from the man's face and shows the construction site. This video sequence has high motion and detail.

(v) **Coastguard**: Here, a patrol speedboat is cruising the banks of a water body at moderate to high speed.

(vi) **Ice**: This video consists of people skating on the ice. It has slow to medium motion of the skaters.

(vii) **Flower**: Here there is slow camera panning motion over the landscape. There is a lot of spatial detail and color in this video.

(viii) **Mobile and Calender**: This sequence is also commonly known as Mobile. In this sequence, there is a lot of detail in the background which is a calender. The calender has vertical motion. A train is pushing a ball in the horizontal direction. This sequence has a very complex texture with motion in different directions.

(ix) **Tempete**: The camera zooms into the scene. There is a lot of spatial detail and fast random motion.

Some sample frames from the sequences for testing the proposed algorithms are shown in Figure 3.2.

3.4 Performance Evaluation

The algorithms are implemented in the JM12.4 reference encoder. The baseline profile of the standard are used for the implementations. The QP levels used are 24,28,32 and 36. Intra coding is done on the first frame. The motion search range for both the CIF and the QCIF sequences is ±32. CAVLC is employed for the entropy coding. The simulations are carried according to the conditions given in [99]. To evaluate the average encoding

(a) News (b) Mother and Daughter (c) Foreman

(d) Ice (e) Tempete (f) Mobile and Calender

Figure 3.2: Sample frames from test video sequences in CIF resolution

performance over a range of QPs, the differences in PSNR (ΔPSNR) in dB and bitrate (ΔRate (%)) are calculated according to numerical averages between RD curves as given by Bjontegaard [100].

3.4.1 Rate Distortion Performance

The compression performance is evaluated using the "rate-distortion performance". This is done by plotting the PSNR against the bit rate. The results using different methods are plotted in the same figure. A typical Rate Distortion (RD) curve is shown in Figure 3.3. It shows a sample plot where algorithm 'A' performs better than algorithm 'B'. Here, the proposed algorithm (B) is compared with the reference software (A), hence the rate distortion curve for both of them are plotted in the same figure and the performance

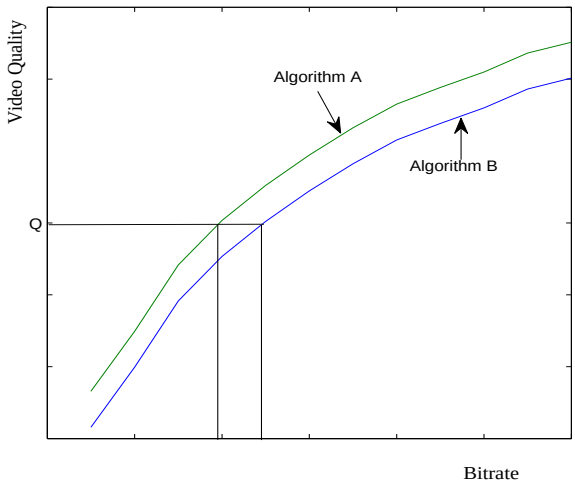


Figure 3.3: Rate Distortion Plot

is evaluated. It clearly shows that for the same video quality, the bitrate of algorithm 'A' is less than that of algorithm 'B'. Thus rate distortion performance of algorithm 'A' is better than algorithm 'B'.

3.4.2 Computational Complexity

The computational complexity can be measured in a number of ways. The measures used in this work are given below:

A. Encoding Time

The time taken by the encoder to encode the video sequence is the encoding time. The time spent by the encoder is taken as a measure of the computational complexity of the encoder. The encoding time of the encoder using the reference encoder and that of the proposed methods are used for comparision. The same test conditions are maintained for all the algorithms. The time saving (ΔT) of the proposed method compared to the reference software is calculated as follows:

$$\Delta T = \frac{T_{JM} - T_P}{T_{JM}} \times 100 \tag{3.1}$$

Here T_{JM} is the encoding time with the reference encoder and T_P is the encoding time with the proposed algorithm. ΔT is expressed as a percentage. A positive value of ΔT implies that the proposed method is faster than the reference encoder and vice versa.

B. Number of Search Points

The search efficiency (or speed-up factor) for the VBS-ME encountered in H.264/AVC standard cannot be evaluated by simply using the number of search points computed for performance evaluation. The computation of SAD is in direct proportion to the block size and contributes to the majority of computational load in computing the RD cost J. A new metric for estimating computational cost was proposed in [73] where the computational cost per search point for a 4 x 4 block is defined as one unit cost. The computation cost for each mode is defined as

Computational Costs = SP × W. SP is the number of search points per mode. W is the weight associated with a particular mode. The total searches per block can be estimated by finding the computational cost for each mode. Thus for the seven modes the total search points per MB, $\mathrm{SP_{MB}}$ is determined as

$$\mathrm{SP_{MB}} = \sum_{\mathrm{i=1}}^{7} \mathrm{W_i} (\sum_{\mathrm{j=1}}^{\mathrm{N_i}} \mathrm{SP_{i,j}}), \tag{3.2}$$

where 'i' is the mode for the MB, 'j' is the block serial number and $\mathrm{W_i}$ is the weight of the computational cost incurred for the i-th mode due to different block sizes involved. $SP_{i,j}$ is the search points for the i-th mode and the j-th block type. If the area of 16 x 16 block size is taken to be one unit, then $\mathrm{N_i}$ is the ratio of the area of 16 x 16 block to that of the current block mode. The values of $\mathrm{N_i}$ and $\mathrm{W_i}$ are given in Table 4.2 for different block sizes.

Table 3.2: Value of N and W for different block sizes

Parameter	16x16	16x8	8x16	8x8	8x4	4x8	4x4
N_i	1	2	2	4	8	8	16
W_i	1	1/2	1/2	1/4	1/8	1/8	1/16

3.4.3 Bit Rate

The average bitrate of the encoded bitstream is measured for both the reference encoder and the proposed encoder. The bitrate increase or decrease is found by using the following:

$$\Delta Rate = \frac{Rate_P - Rate_{JM}}{Rate_{JM}} \times 100 \tag{3.3}$$

Here $Rate_{JM}$ is the encoding bitrate with the reference encoder and $Rate_P$ is the encoding bitrate with the proposed algorithm. $\Delta Rate$ is expressed as a %. A positive value of $\Delta Rate$ implies that the proposed method has a higher bitrate than the reference encoder and vice versa.

3.4.4 Video Quality

The H.264/AVC is a lossy compression method. Since information is lost during the encoding process, the decoded video has reduced information content than the raw video. However, in the process of encoding, the visual quality of the video should not degrade. Video quality is a subjective measure. People perceive the same video in different ways. Therefore it is difficult to measure the video quality precisely. There are subjective and objective measures of the video quality.

A. Subjective Video Quality Measurement

This measurement is done for the perceptual quality of the video. In our work, the subjective measurement is carried out by viewing the videos encoded with the reference encoder and the proposed methods.

B. Objective Video Quality Measurement

The Peak Signal to Noise Ratio (PSNR) is the most commonly used objective measure for the video quality. The PSNR is already discussed in Section 2.2. While making the PSNR comparisions, the average of the PSNR of all the frames are taken. Higher PSNR

indicates a better video quality. The differences in the PSNR obtained from the reference encoder and the proposed method gives the degradation or improvement of the video quality. The PSNR difference $\Delta PSNR$ is calculated using

$$\Delta PSNR = PSNR_{JM} - PSNR_P \tag{3.4}$$

Here $PSNR_{JM}$ is the PSNR with the reference encoder and $PSNR_P$ is the PSNR with the proposed algorithm. $\Delta PSNR$ is expressed in dB. A positive value of $\Delta PSNR$ implies that the there is degradation in quality using the proposed method compared to the reference encoder and vice versa.

3.5 Conclusion

This chapter details the experimental methods used to test the fast encoding algorithms developed in this thesis. The H.264/AVC reference software JM12.4 is used as the benchmark software and the developed algorithms are tested with the results of this software. The different ways of measuring the performance are described in this chapter. These measurement criteria will be used throughout this work.

4

Edge Histogram Based Fast Mode Decision for H.264/AVC

Contents

4.1 Observations from Exhaustive Intra Prediction Process . . . 50
4.2 Fast Intra Mode Decision in EHFMD Algorithm 55
4.3 Experimental Observations 58
4.4 Observations from Full Search ME in H.264/AVC 61
4.5 Fast Inter Mode Decision in EHFMD Algorithm 67
4.6 Experimental Results 76
4.7 Conclusions 78

In the video coding process, the prediction modes depend largely on the characteristics of the video sequence. Neighboring video frames generally have a high degree of similarity between them. Natural videos have many regions with homogeneous textures. There are regions in which the video objects have strong edges. In this chapter, fast mode decision algorithms for both the intra and the inter prediction process are proposed. These algorithms limit the search to a subset of modes based on the characteristics of the video thereby reducing the total computational time.

The rest of the chapter is organized as follows. The chapter begins with some detailed observations gleaned from an exhaustive study of intra prediction and full search inter prediction process of the H.264/AVC encoder. An Edge Histogram based Fast Mode Decision (EHFMD) algorithm is proposed in this chapter. The EHFMD algorithm consists of two parts: one for intra prediction and the other for inter prediction. The chapter ends with results and conclusions.

4.1 Observations from Exhaustive Intra Prediction Process

The spatial redundancies within a video image is exploited in the intra coding process [2,12,13]. The resulting picture is said to be intra coded and is referred to as an I-picture. Intra prediction in the H.264/AVC utilizes the neighboring pixels of the previously coded blocks. The prediction block may be formed for each 4x4 block (I4MB prediction) or for an entire MB (I16MB prediction). As mentioned in Chapter 2, I4MB uses nine prediction modes whereas I16MB uses four prediction modes. An encoder chooses the appropriate intra mode that minimize the total number of bits in the prediction and the residual. Regions having complex texture and the MBs which are less spatially related are encoded with smaller block sizes (I4MB) whereas homogeneous regions of the frame where the texture is largely uniform is encoded with larger block size (I16MB). The QP also affects the mode in which a MB is encoded. At lower QP, more MBs are encoded with I4MB whereas at higher QP, I16MB is more prevalent. Certain observations made after an

exhaustive intra prediction search for various video sequences at CIF and QCIF resolutions are detailed next.

4.1.1 Mode dependency of sequences at different QP

The mode in which a MB is encoded is not the same for all values of QP. The mode distribution of MBs for intra prediction depend on the value of the QP. To study the distribution, exhaustive intra prediction was done for different test sequences at different QP. The mode selected when the QP is small consisted mainly of the I4MB modes. As QP is increased it is found that the number of modes encoded in the I16MB increased. The distribution for different QPs are obtained for some video sequences and is given in Table 4.1. From this distribution, we observe that for QP up to 24, more than 80% of the MBs are encoded in the I4MB mode. For values of QP above 24 and below 40, MBs are encoded in both the I4MB and I16MB modes. For QP above 45, we note that more than 80% of the MBs are encoded in the I16MB mode. Figure 4.1 shows the average distribution of modes with respect to the QP for both QCIF and CIF sequences.

Table 4.1: Mode Distribution (%) at different QP

Sequence QCIF	Type Of MB	Quantization Parameter					
		8	16	24	32	40	45
Foreman	I4MB	99	93	85	59	30	14
	I16MB	1	7	15	41	70	86
Crew	I4MB	99	96	87	68	41	21
	I16MB	1	4	13	32	59	79
Mobile	I4MB	96	95	88	71	39	19
	I16MB	4	5	12	29	61	81
Bus	I4MB	99	92	87	68	40	21
	I16MB	1	8	13	32	60	79
City	I4MB	98	93	81	63	25	21
	I16MB	2	7	19	37	75	79
Soccer	I4MB	97	94	81	65	24	20
	I16MB	3	6	19	35	76	80
Football	I4MB	98	95	85	75	40	23
	I16MB	2	5	15	25	60	77
Average	**I4MB**	**98**	**93**	**85**	**67**	**34**	**19**
	I16MB	**2**	**7**	**15**	**33**	**66**	**81**

Sequence CIF	Type Of MB	Quantization Parameter					
		8	16	24	32	40	45
Mobile	I4MB	96	94	89	75	46	22
	I16MB	4	6	11	25	54	78
Bus	I4MB	95	93	84	66	30	19
	I16MB	5	7	16	36	70	81
City	I4MB	95	93	85	70	36	19
	I16MB	5	7	15	30	64	81
News	I4MB	95	93	82	63	32	19
	I16MB	5	7	18	37	68	81
Tempete	I4MB	99	94	85	74	41	21
	I16MB	1	6	15	26	59	79
Mother-daughter	I4MB	91	88	83	62	29	18
	I16MB	9	12	17	38	71	82
Silent	I4MB	97	90	85	69	39	20
	I16MB	3	10	15	31	61	80
Average	**I4MB**	**95**	**92**	**84**	**68**	**35**	**19**
	I16MB	**5**	**8**	**16**	**32**	**65**	**81**

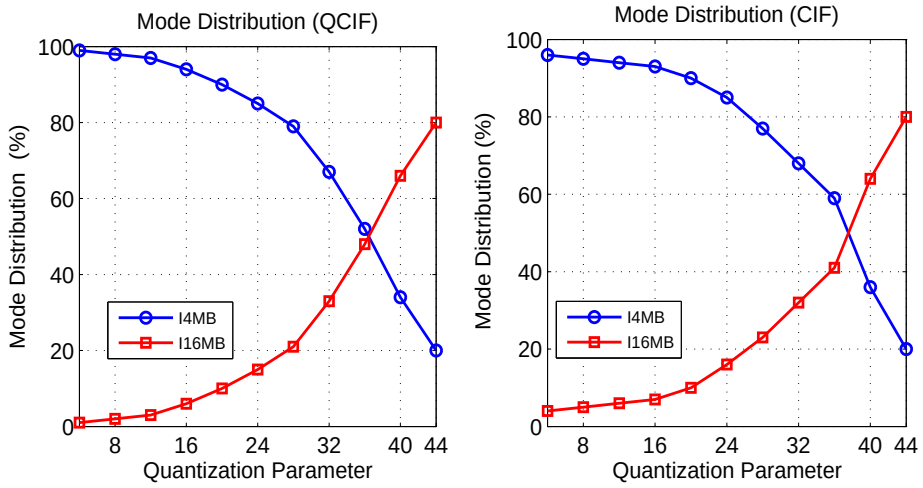


Figure 4.1: Distribution of MBs in I4MB and I16MB modes

4.1.2 Directional Correlation

H.264/AVC utilizes the directional intra prediction in the spatial domain. Both I4MB and I16MB have directional modes. This has been discussed in Section 2.5. The directions for the intra prediction modes are shown in Figure 4.2.

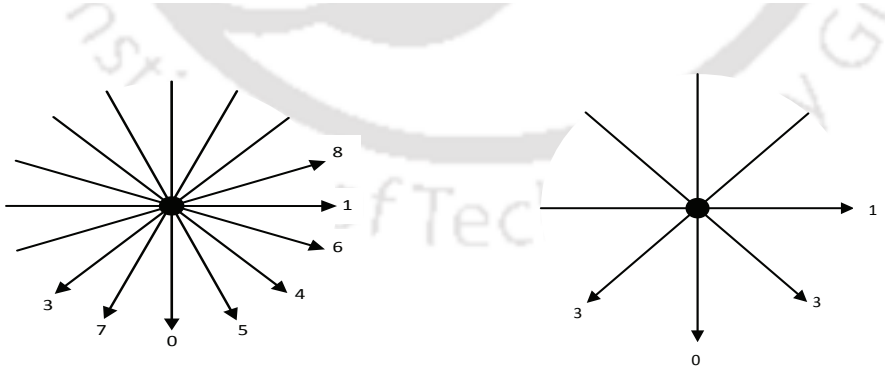
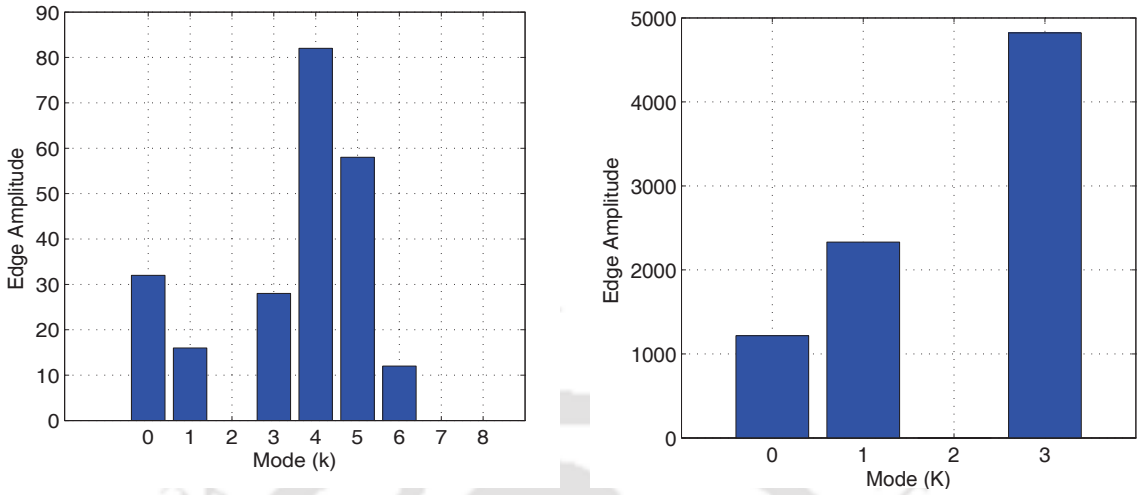


Figure 4.2: Directional Intra Prediction modes for I4MB and I16MB

To study the directional nature of the intra prediction process, for each MB in a frame the edge amplitude histogram was obtained. The sum of the edge amplitudes in the directions of the intra prediction modes for both I4MB and I16MB are determined.



(a) Edge Direction Histogram for I4MB (b) Edge Direction Histogram for I16MB

Figure 4.3: A typical Edge Histogram for a MB

Sobel Operators are used to obtain the edge information. Each pixel in the block will be associated with an edge vector containing edge direction and amplitude. An edge vector $D_{i,j} = \{dx_{i,j}, dy_{i,j}\}$ is defined where

$$dx_{i,j} = p_{i-1,j+1} + 2 \times p_{i,j+1} + p_{i+1,j+1} - p_{i-1,j-1} - 2 \times p_{i,j-1} - p_{i+1,j-1} \tag{4.1}$$

$$dy_{i,j} = p_{i+1,j-1} + 2 \times p_{i+1,j} + p_{i+1,j+1} - p_{i-1,j-1} - 2 \times p_{i-1,j} - p_{i-1,j+1} \tag{4.2}$$

Here $dx_{i,j}$ and $dy_{i,j}$ are the differences in the vertical and the horizontal directions. The amplitude of the edge vector is computed as

$$Amp(D_{i,j}) = |dx_{i,j}| + |dy_{i,j}| \,. \tag{4.3}$$

The direction of the edge is determined by the hyper-function

$$Ang(D_{i,j}) = \frac{180^\circ}{\pi} \times \arctan(\frac{dy_{i,j}}{dx_{i,j}}), \qquad |Ang(D_{i,j})| < 90^\circ \tag{4.4}$$

Figure 4.3(a) and Figure 4.3(b) shows a typical edge amplitude histogram of a MB for I4MB and I16MB predictions as a function of the mode directions. Here 'k' represents the different mode directions. DC mode is represented by k=2.

The figures show the edge amplitude histograms in the direction of different modes of a MB. In this example, from Figure 4.3(a) it is evident that the three dominant edges are directed along the mode 4, 5 and 0. The final mode obtained from the exhaustive intra prediction for this MB was mode 4. Similar study for other modes showed that the final mode from exhaustive intra prediction is any one of the three modes giving the dominant edges. However in some cases where the MB is homogeneous, DC mode was the chosen mode. Thus the final encoding mode for a MB is related to the direction of the dominant edges in that block.

4.1.3 Residual Complexity

In the intra prediction process, a block is predicted from spatially neighboring samples. The neighboring upper and left pixels of a block are used to extrapolate the predicted pixels for spatial redundancy removal. These neighboring pixels used for the prediction are sometime very different from the pixels to be predicted. In such cases the prediction becomes poor and the residuals obtained have large values. This means that more bits will be needed to encode the residuals. The difference between the reference (neighboring pixels for prediction) and the to-be-coded pixel values is an effective criterion to measure the predicted residual complexity. For any $N \times N$ block, we define a metric Residual Complexity (RC) as

$$RC = \frac{1}{(N+1)(N+1)} \sum_{i=-1}^{N-1} \sum_{i=-1}^{N-1} |I(x+i, y+i) - I_{mean}| \tag{4.5}$$

Here I_{mean} is the mean of the pixels in the statistical window $(N+1) \times (N+1)$ which includes the reference pixels and x and y are the positions of the MB in the horizontal and vertical directions. When the reference pixels used to extrapolate the predicted pixels are very different from the pixels in the block to be predicted, the RC becomes high. This occurs if the block is not homogeneous indicating a region with high spatial details. Lower values of RC indicate a homogenous region with uniform pixel values.

The RC parameter tends to overestimate the residual complexity of a MB if the

statistical window size is too large. When N=16, the window size is 17×17 and the RC is sometimes overestimated. It is noted in Section 4.1.1 that for QP values upto 32, I4MB modes are more prevalent than the I16MB modes. For QP values upto 40, many MBs are encoded in the I4MB mode. The RC parameter is thus used for the I4MB blocks. It is observed that regions with homogeneous textures with similar spatial properties have low values of RC and with the increasing spatial complexity RC increases. Thus RC is an effective measure for determining the homogeneity of a block.

4.2 Fast Intra Mode Decision in EHFMD Algorithm

From an analysis of the exhaustive intra prediction process and the statistics that we have obtained in the last section, we observe that the final mode selected for a MB depends upon the QP, the direction of the edge in the MB and the residual complexity. For fast intra mode selection these factors are considered.

Two thresholds T_1 and T_2 are defined. The threshold T_1 is the value of the QP below which nearly 85% of the MBs are coded in I4MB mode. From Figure 4.1 it is observed that this condition is satisfied for QP values upto 24. Hence threshold T_1 is taken equal to 24. The threshold T_2 is the value of the QP above which nearly 85% of the MBs are coded in I16MB mode. This is satisfied for QP values above 45 and hence threshold T_2 is taken as 45.

Each frame of a sequence is divided into MBs. For each MB, the QP is compared with T_1 and T_2. The decision is taken as follows :

$$\text{Decision} = \begin{cases} \text{Algorithm A if QP} \leq \text{T}_1 \\ \text{Algorithm B if QP} \geq \text{T}_2 \\ \text{Algorithm C if T}_1 < \text{QP} < \text{T}_2 \end{cases} \tag{4.6}$$

Each algorithm requires that the edge histogram for the MB be calculated. Thus, edge histogram is obtained for each MB. Algorithm A and C requires the calculation of the RC for the I4MB prediction. The determination of edge histogram and the RC is described

next.

4.2.1 Determination of Edge Histogram

The edges in the directions of the intra prediction modes are obtained from the procedure described in Section 4.1.2. The edge histogram amplitude is obtained by summing up the amplitude of the edges with similar edge direction. There are nine edge directions for the I4MB and four for I16MB (shown in Figure 4.3). Three dominant edge directions for a MB are selected and the information is used for the final mode selection of the MB.

4.2.2 Determination of Residual Complexity

The RC parameter is calculated using (4.5) only for those MBs which qualify for the I4MB. The MBs qualifying for I4MB mode will be decided from the value of the QP for the MB. The homogeneity of a 4×4 block is determined from the value of the RC. A threshold T_{RC} is defined and if the RC of a block is less than T_{RC} , the block is said to be homogenous. Homogeneous MB in most cases are encoded in mode 2 (DC mode). Thus, if the MB is homogeneous then mode 2 (DC mode) will be included in the RDO calculation.

Threshold Selection for T_{RC}

Exhaustive intra prediction is performed with all the directional I4MB modes. The DC mode is included in the prediction process when the RC value is below a threshold. We take the performance as 100 % for the threshold value of RC=0. In this case, the mode prediction will include all the modes. The threshold value is gradually incremented by one and its performance relative to the standard performance is noted. Figure 4.4 shows the performance of the Mobile sequence for various threshold values for RC. The plot clearly show that the performance is above 90% for threshold values of RC upto 5 and for values above this, the performance gradually deteriorates. Thus the threshold for RC, T_{RC}, is taken as 5 and it is observed that this value gives consistent and good results for all sequences considered. Thus, based on the above discussions, we set the value of

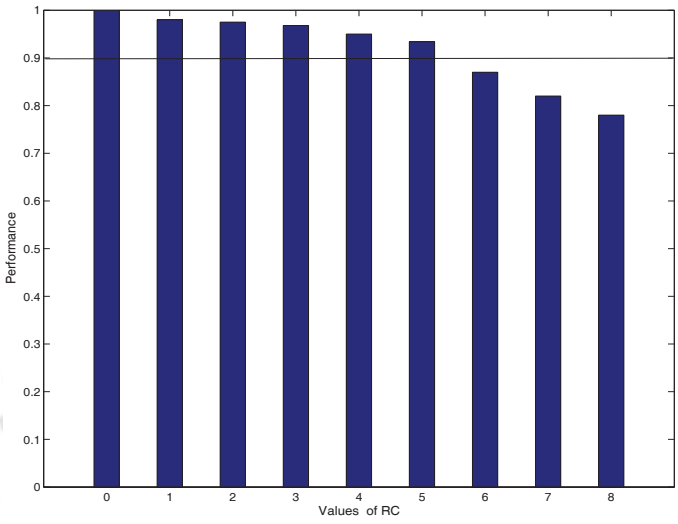


Figure 4.4: Performance of Mobile sequence at different values of T_{RC}

T_{RC} at 5 for I4MB prediction.

4.2.3 Overall Intra Prediction in EHFMD Algorithm

Each MB will qualify for any one of the three algorithms A, B or C depending upon the QP value. The decision is taken according to the conditions given in (4.6).

Algorithm A: Only I4MB prediction

Step 1: The MB is divided into sixteen 4×4 subblocks.

Step 2: For each subblock the following is done:

(i) The edge histogram is obtained. The modes pertaining to the three dominant edges are determined. These modes are denoted by M1, M2 and M3.

(ii) RC is determined for the block.

(iii) If RC $> T_{RC}$, RDO procedure is carried out for the modes M1, M2 and M3. Go to (v). Else,

(iv) RDO procedure is carried out for the M1 and DC modes. Go to (v).

(v) Select the best mode which gives the least RDO cost among the chosen 4×4 intra modes for the subblock.

Step 3: Repeat Step 2 for each subblock.

Step 4: Proceed with the next MB qualifying for Algorithm A.

Algorithm B: Only I16MB prediction

Step 1: The edge histogram for the MB is obtained. The modes pertaining to the dominant edge is determined and is denoted by M16.

Step 2: RDO procedure is carried out for the M16 and DC modes.

Step 3: Select the mode with the least RDO cost among these two modes.

Step 4: Proceed with the next MB qualifying for Algorithm B.

Algorithm C: Total Prediction

This algorithm is selected when the QP lies between the two thresholds T_1 and T_2. In this region the MBs get encoded both in the I4MB mode or the I16MB mode. Thus, this algorithms includes both the proposed fast I4MB and I16MB predictions. For each MB, the steps involved are

Step 1: Perform Steps 1, 2 and 3 of Algorithm A. Store the RD cost for the best mode for I4MB mode.

Step 2: Perform Steps 1, 2 and 3 of Algorithm B. Store the RD cost for the best mode for I16MB mode.

Step 3: The final mode is the one that gives least RDO cost from Step 1 and Step 2.

Step 4: Proceed with the next MB qualifying for Algorithm C.

4.3 Experimental Observations

The QP used for the simulations is varied from 4 to 48. The thresholds T_1 is taken at 24 and T_2 is taken at 45 and T_{RC} is fixed at 5. The performance of the proposed fast prediction algorithm is compared with the results from JM12.4 reference software

Table 4.2: Results For All Intra Frame Sequences

Format	Sequence	EHFMD			Pan's et al.'s [22]			Liu et al.'s [34]		
		ΔPSNR (%)	ΔRate (dB)	ΔT (%)	ΔPSNR (%)	ΔRate (dB)	ΔT (%)	ΔPSNR (%)	ΔRate (dB)	ΔT (%)
QCIF	Foreman	0.23	1.53	68.80	0.24	3.24	65.40	0.20	2.94	66.20
	City	0.21	1.30	62.03	0.35	2.12	55.23	0.27	1.89	58.78
	Mobile	0.23	1.14	67.24	0.25	2.94	61.34	0.26	2.98	63.43
	Bus	0.22	1.15	63.41	0.18	2.45	56.34	0.19	2.49	57.52
	Football	0.33	2.05	64.35	0.24	2.67	53.45	0.26	2.69	58.35
	Soccer	0.24	1.68	60.05	0.20	2.78	49.36	0.19	2.23	52.45
	Crew	0.35	1.23	53.62	0.32	1.67	45.67	0.29	1.47	50.34
	Average	**0.25**	**1.44**	**62.78**	**0.25**	**2.55**	**55.25**	**0.23**	**2.38**	**58.15**
CIF	Foreman	0.60	1.87	66.67	0.25	3.26	59.13	0.21	2.86	64.15
	City	0.37	1.98	61.15	0.35	1.80	46.54	0.29	1.62	52.43
	Mobile	0.31	2.85	65.34	0.26	2.96	55.35	0.26	2.96	60.35
	Bus	0.14	1.29	66.54	0.19	2.10	56.41	0.15	1.89	58.52
	News	0.19	1.26	61.14	0.28	3.01	48.56	0.26	2.76	54.19
	Silent	0.19	2.15	71.45	0.18	1.98	54.78	0.15	1.48	63.28
	Tempete	0.18	2.12	68.16	0.20	2.27	50.43	0.22	2.43	55.28
	Mother-daughter	0.24	2.01	70.06	0.25	2.16	49.76	0.24	2.10	58.53
	Crew	0.31	1.56	67.12	0.29	1.35	55.12	0.30	1.39	60.30
	Average	**0.28**	**1.89**	**66.40**	**0.25**	**2.32**	**52.90**	**0.23**	**2.16**	**58.55**

ΔPSNR(+/-): picture quality loss/gain measured in dB
ΔRate(+/-): bitrate increase/decrease measured as a %
ΔT(+/-): encoding time saving/loss measured as a %

and Pan's [22] and Liu's [34] algorithm . Results are presented as improvements over the standard H.264/AVC benchmark JM12.4. Simulations are carried with the setup described in Chapter 3. For testing fast intra prediction algorithm, all the frames are set as I frames, that is, the period of I frames is set to 1. Table 4.2 summarizes the average PSNR improvements/loss, the saving in encoding time and the loss in bitrate by the proposed method for coding the sequences. It is seen that the loss in the PSNR and bitrate over the conventional approach are modest but there is a large gain in the encoding time of an average of 62%.

In H.264/AVC encoding, MBs in P frame also choose Intra as the one of the prediction modes in the RDO operation. Experiments on IPPP... sequences show that there is no noticeable degradation in the PSNR and the bitrate but there is large saving in the computational time. Table 4.3 shows the performance comparison of the proposed method with JM12.4, [22] and [34].

Table 4.3: Results For IPPP . . . Sequences

Format	Sequence	EHFMD			Pan's et al.'s [22]			Liu et al.'s [34]		
		ΔPSNR (%)	ΔRate (dB)	ΔT (%)	ΔPSNR (%)	ΔRate (dB)	ΔT (%)	ΔPSNR (%)	ΔRate (dB)	ΔT (%)
QCIF	Foreman	0.05	1.24	54.40	0.07	1.45	24.45	0.09	1.86	39.56
	City	0.06	1.22	50.24	0.08	1.80	26.35	0.07	1.35	37.23
	Mobile	0.06	1.20	54.35	0.02	0.79	30.34	0.08	1.45	40.23
	Bus	0.02	0.82	51.15	0.01	0.42	27.43	0.05	1.21	37.17
	Football	0.06	1.33	54.34	0.07	1.34	27.34	0.07	1.36	38.62
	Soccer	0.08	1.51	58.16	0.07	1.28	28.97	0.08	1.42	40.25
	Crew	0.10	1.63	51.76	0.09	1.55	25.11	0.07	1.37	39.23
	Average	**0.06**	**1.27**	**53.48**	**0.06**	**1.23**	**27.11**	**1.43**	**1.10**	**38.89**
CIF	Foreman	0.07	1.32	55.45	0.07	1.35	24.12	0.08	1.39	40.21
	City	0.10	1.67	50.67	0.06	1.24	23.67	0.08	1.41	37.25
	Mobile	0.01	0.69	52.33	0.01	0.52	28.32	40.15	0.07	1.32
	Bus	0.02	0.82	50.56	0.01	0.64	26.82	0.08	1.41	38.21
	News	0.04	1.09	49.49	0.06	1.23	25.35	0.06	1.24	42.15
	Silent	0.01	0.63	56.16	0.03	0.85	23.39	0.05	1.18	43.05
	Tempete	0.02	0.83	53.67	0.02	0.81	25.68	0.10	1.84	39.15
	Mother-daughter	0.02	0.78	57.70	0.04	1.12	26.49	0.05	1.19	41.36
	Crew	0.05	1.26	52.47	0.10	1.64	24.46	0.08	1.39	39.81
	Average	**0.03**	**1.01**	**53.16**	**0.04**	**1.04**	**25.36**	**0.07**	**1.37**	**40.14**

ΔPSNR(+/-): picture quality gain/loss measured in dB
ΔRate(+/-): bitrate increase/decrease measured as a %
ΔT(+/-): encoding time saving/loss measured as a %

Complexity Analysis

Table 4.4 summarizes the number of candidates selected for RDO calculation for the proposed EHFMD algorithm. The number of candidate modes are compared with the JM 12.4 reference software and Pan's method [22]. If we consider only the luma component then for each MB the proposed algorithm needs less searches to be performed compared to Pan's method [22] and JM 12.4 software. Hence there is large reduction in the number of RDO calculation. For one frame in the QCIF sequence at the most 495 RDO calculations are required as against 14652 as per the JVT reference software.

Table 4.4: Number of candidate Modes

MB type (Y)	RDO Calculations per MB						
	JM 12.4	Pan's method	EHFMD (Intra Prediction)				
			QP< T_1		T_1 <QP< T_2		QP> T_2
			RC<=5	RC>5	RC<=5	RC>5	
4 × 4	9	4	2	3	2	3	0
16 × 16	4	2	0	0	2	2	2
Total	13	6	2	3	4	5	2

PSNR-Rate Curves

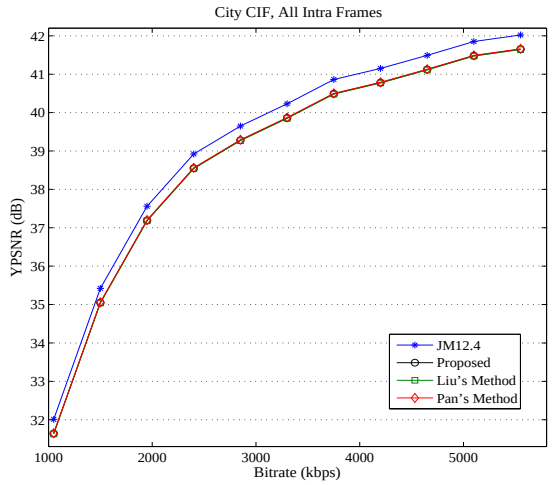
Figure 4.5 shows the RD curve for some sequences in the QCIF and the CIF resolution. It shows the RD performance of four algorithms, the exhaustive method, Pan's method [22], Liu's method [34]and the method proposed method. It is observed that the proposed method gives similar RD performance as compared to JM12.4 reference software and improves on [22] and [34].

4.3.1 Discussion

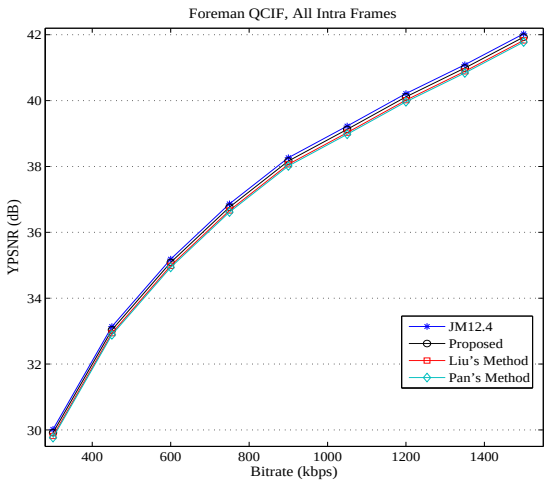
The proposed intra prediction algorithm is partly inspired by the work done by Pan et.al [22]. In their algorithm, the direction of the edges were considered for the mode selection in the intra coding decision. Our algorithm relies largely on the statistical analysis of the mode decision process. We have analyzed the dependency of the modes on the QP. The value of the QP and the edge direction information have been considered for the mode selection process. The homogeneity factor is included for fine tuning the mode selection process. Overall our method gives better time saving as compared with the reference software JM12.4 and improves on Pan's [22] and Liu's [34] algorithm.

4.4 Observations from Full Search ME in H.264/AVC

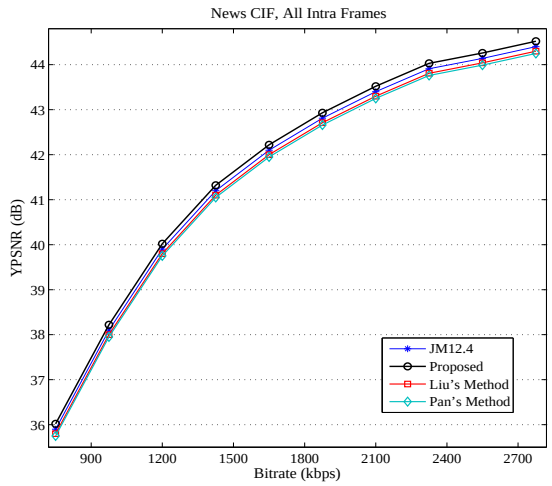
Inter prediction is a process of predicting a block of samples from a picture that has already been coded and transmitted (reference picture). The reference picture is chosen from a list of previously coded pictures. In H.264/AVC encoder, the current picture samples are predicted from the reference picture samples and are subtracted from the original samples to form a residual which is then coded and transmitted. The temporal redundancies between frames are exploited in the inter prediction process. The inter prediction process in H.264/AVC has been described in detail in Chapter 2. There are seven block sizes P_{16x16}, P_{16x8}, P_{8x16}, P_{8x8}, P_{8x4}, P_{4x8} and P_{4x4} as shown in Figure 4.6 that are used by H.264/AVC besides the SKIP and the INTRA modes [2, 13, 16]. In the full



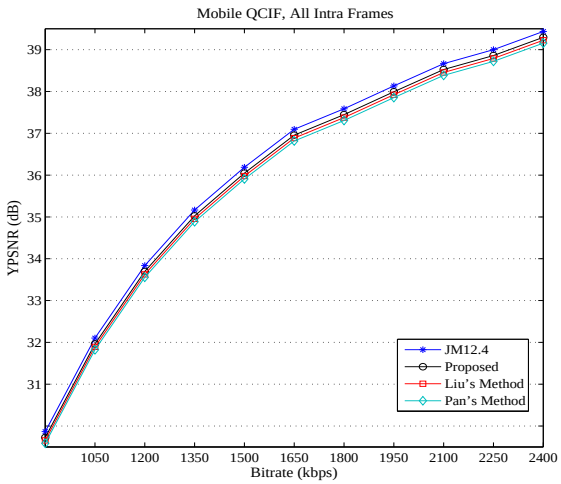
(a) RD Performance for City(CIF) Sequence



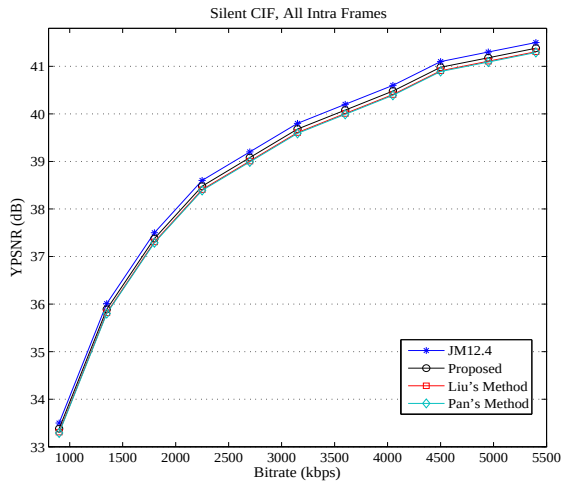
(b) RD Performance for Foreman(QCIF) Sequence



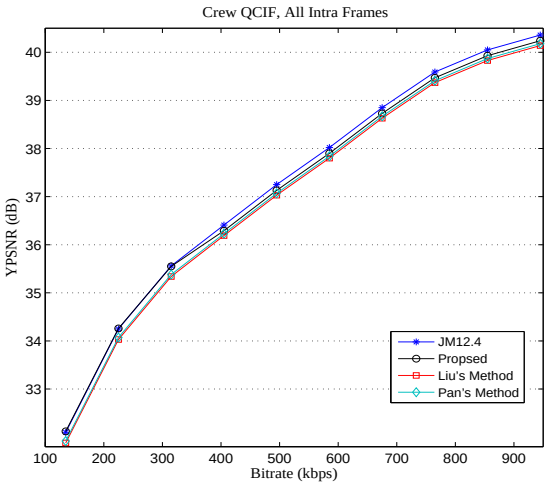
(c) RD Performance for News(CIF) Sequence



(d) RD Performance for Mobile(QCIF) Sequence



(e) RD Performance for Silent(CIF) Sequence



(f) RD Performance for Crew(QCIF) Sequence

Figure 4.5: RD Performance for various sequence in the CIF and the QCIF resolution.

search ME where every possible coding mode and every prediction type is checked, the encoder chooses the best mode in terms of the least RD cost (J_{MODE}) by employing the Lagrangian RDO process [19].

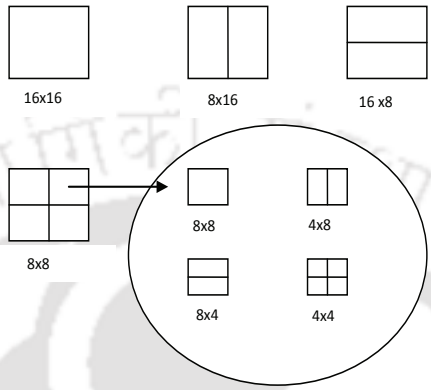


Figure 4.6: Inter prediction block sizes for a MB

The high computational burden on the encoder in performing the full rate-distortion optimized mode selection process is a practical constraint on the encoder. Neighboring video frames have large similarities between them. Motion estimation and compensation attempt to reduce the temporal redundancy by exploiting these similarities. The encoder chooses larger partitions for the homogenous regions with less details which are characterized by similar spatial property. Motion in some region of a frame could be very simple whereas in other parts of the same frame the motion could be complex. Regions having little or no motion are considered stationary and these usually get encoded in the SKIP mode. The encoder chooses larger partitions for regions with less motion content and smaller partitions in areas of complex motion. A better prediction signal may be obtained by using 4×4 sub partition sizes in scenes with high motion and detail. Smaller partition implies larger number of MVs, hence more bits are required to indicate the MVs and the choice of partitions. Making the best choice for the motion compensation partition can have a significant impact on the compression efficiency of the encoder. In real video sequences the distribution of inter prediction modes is not uniform across the MBs [12]. It depends upon the characteristics of the video sequences. To study the mode

distribution of the MBs, we perform a full search ME for different test sequences and analyze the results.

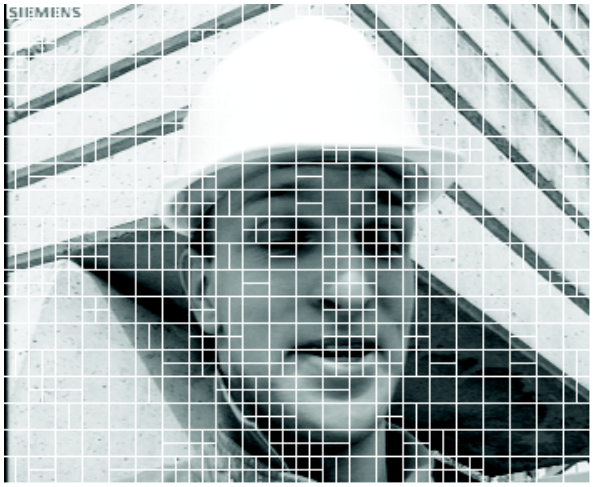
4.4.1 Mode Distribution for Different Test Sequences

In encoding a video sequences, inter prediction is performed with a number of modes to obtain the best mode for encoding. The modes in which the MBs in a frame are encoded are usually not same. It usually depends on the characteristics of the video sequences.

Figure 4.7 shows how frames from different sequences are coded as inter (P) MB with the choice of partitions overlaid on the frame. Typically, SKIP mode and P_{16x16} modes are selected for homogeneous regions and for regions with lesser motion while smaller partitions are chosen for areas with complex motion. Table 4.5 shows the distribution of MBs (in %) encoded in different modes for a QP value of 32 obtained from a full search ME. In the table, the P_{8x8} mode includes the P_{8x8}, P_{8x4}, P_{4x8} and P_{4x4} modes. The results clearly show that Class A sequences that have low motion content have most of the MBs encoded with larger block sizes. For medium motion Class B sequences, the MBs are encoded with both large and small block sizes. When the sequences have complex motion as in Class C sequences, smaller block sizes are more prevalent. From these observations it may be concluded that for all types of sequences there is a significant percentage of MBs that are encoded in the SKIP mode and P_{16x16} mode. If such MBs can be detected at an early stage, it will result in large savings in computation. Thus, if the stationary and homogenous regions in a frame can be determined prior to ME, early decision on the encoding mode can be taken. Further, it is noted that the INTRA mode takes up less than 1.5 % of the total mode distribution. Thus calculating the RD cost for the INTRA mode is wasteful of the computational resource in most of the cases.

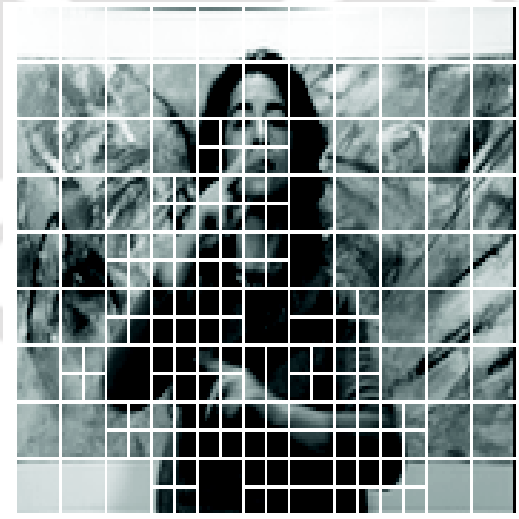
4.4.2 Correlation Between Prediction Block Size and Motion Direction

It is observed that the inter prediction block sizes in Figure 4.7 have either horizontal or vertical partitions as shown in Figure 4.6. The MB partitions P_{16x16}, P_{16x8} and P_{8x16}



(a) Partition Choice for Foreman CIF

(b) Partition Choice for News CIF



(c) Partition Choice for Silent QCIF

Figure 4.7: Partition choices for different sequences overlaid on a frame.

Table 4.5: Distribution of MBs (%) in different Modes at QP=32

Sequence		SKIP	P_{16x16}	P_{16x8}	P_{8x16}	P_{8x8}	INTRA
CIF	News	82.57	6.55	2.25	1.85	6.68	0.00
Class A	Container	83.46	15.42	0.21	0.23	0.67	0.01
	Hallmon	64.76	23.30	2.92	3.25	5.57	0.20
	Coastguard	20.58	44.76	6.70	8.35	19.38	0.23
Class B	Foreman	29.02	34.41	8.90	7.97	18.50	1.20
	Ice	64.76	11.20	4.82	3.92	14.85	0.45
	Flower	24.92	51.82	3.06	2.70	17.47	0.03
Class C	Mobile	12.04	40.54	8.62	9.82	28.98	0.42
	Stefan	22.69	39.67	4.71	5.86	26.04	1.03
QCIF	Claire	91.17	5.95	0.49	0.66	1.73	0.00
Class A	Suzie	60.63	23.12	7.41	5.03	3.54	0.27
	MissAmerica	64.89	28.46	2.09	2.01	2.49	0.06
	Foreman	34.53	37.88	6.54	7.34	12.76	0.95
Class B	Silent	75.70	8.10	1.93	1.32	12.88	0.07
	Crew	16.92	25.37	12.63	5.97	38.01	1.10
	Football	15.20	20.48	6.26	11.45	28.35	18.26
Class C	Mobile	11.57	42.22	10.85	12.14	22.87	0.35
	Soccer	18.90	29.00	12.63	15.97	22.40	1.10

usually are suitable for the homogeneous regions and regions with less motion. The P_{16x16} mode is suitable for encoding areas that do not contain multiple objects or for areas lying on the boundary of a moving object with continuous motion in both the horizontal and vertical directions. The block size of P_{16x8} is more suitable for areas with continuous horizontal motion and complex vertical motion while block size of P_{8x16} is more suitable for areas with continuous vertical motion and complex horizontal motion. Among the submacroblock partition, P_{8x8} block size is more suitable for areas that contain multiple objects and lie on the boundary of moving objects, have complex motion in both the horizontal and vertical directions. The P_{8x4} block size is suitable for complex motion in the vertical direction whereas P_{4x8} block size is suitable for complex motion in the horizontal direction. Thus, if the direction of the moving boundaries in a video can be determined early, then only thoses modes in the respective direction can be considered for encoding.

From the discussions in the preceding sections, the following observations may be made:

- All MBs in a frame are not encoded in the same mode.
- The mode partitions depend upon the motion content and the homogeneity of the MBs.
- MBs with lower motion content and having homogeneous region usually are encoded in larger partitions whereas MBs with complex motion are usually encoded in smaller partitions.
- The prediction partition is consistent with the direction of motion in the MB.

Based on these observations, the fast inter prediction process of EHFMD is described next.

4.5 Fast Inter Mode Decision in EHFMD Algorithm

We have seen that the mode in which a MB is encoded depends on the characteristic of the MB in term of its motion content, homogeneity and motion direction. These properties of the MB are determined prior to performing the ME process so as to narrow down the search space of the encoding modes. In order to do so, we examine each frame of the video carefully. Every frame is divided into MBs. Each MB is first examined for the motion content. The MB is further examined for homogeneity. If the MB is found to be non-homogeneous and has high motion content, then the direction of the possible motion in the MB is next studied. Depending upon the motion content, the homogeneity and the direction of possible motion in the MB, a decision on the encoding mode is taken.

4.5.1 Determination of Stationarity

Stationarity refers to the stillness between consecutive frames in the temporal direction. Further, regions having similar motion in consecutive frames are also considered stationary. The MBs which are temporally stationary usually get encoded in the SKIP or in the P_{16x16} mode. The simplest method of temporal prediction is to use the previous frame as the prediction for the current frame.

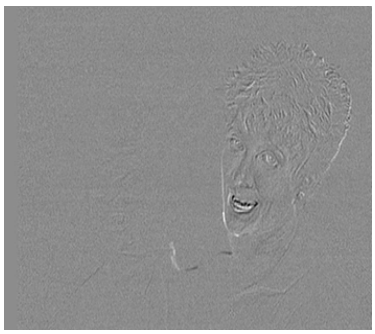


Figure 4.8: Difference image of frame 10 and 11 of Mother and Daughter sequence

Figure 4.8 shows the difference frame formed by subtracting frame 10 from frame 11 of the Mother and Daughter (MaD) sequence. Regions which have little or no motion have zero or low frame difference residual values represented by the grey regions and are encoded with larger block sizes. The light and dark grey areas correspond to the positive and the negative differences representing higher motion activity and hence use smaller block sizes. The frame difference residue block R_{DF} is first obtained by subtracting the MB in the previous frame (MB_P) from the collocated MB in the current frame (MB_C). The sum of absolute values of the difference S_{DIFF} is obtained as,

$$S_{DIFF} = \sum_{i=1,j=1}^{16,16} abs(R_{DF}) \tag{4.7}$$

where

$$R_{DF}(i,j) = MB_C(i,j) - MB_P(i,j), \quad i,j = 1,\ldots,16 \tag{4.8}$$

It is observed that if the MB is stationary, then the frame difference residues in R_{DF} have very low values resulting in a small value of S_{DIFF}. These MBs usually are encoded with larger block sizes. Figure 4.9 shows the distribution of the number of MBs encoded in the SKIP and the P_{16x16} mode for different values of S_{DIFF}. These statistics are obtained from full search ME on the first 10 frames of three sequences at QP value of 24. One

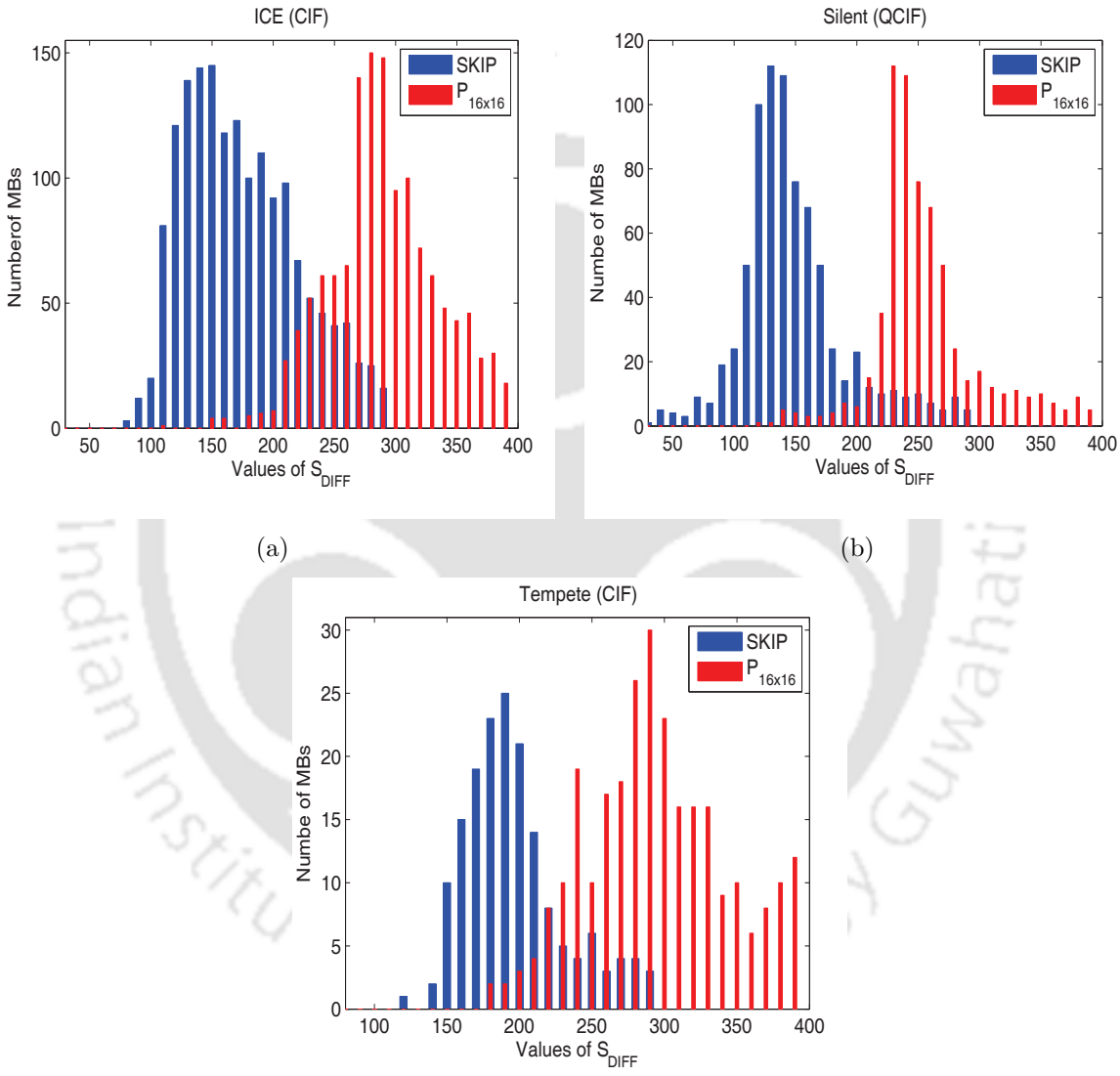


Figure 4.9: Relation between MBs in SKIP Mode and the S_{DIFF}

sequence from each class was considered. Similar results have been obtained for other sequences also. It is clear from the distribution that MBs encoded in the SKIP mode have small values of S_{DIFF}. The MBs encoded in the P_{16x16} mode have higher S_{DIFF} as compared to SKIP mode. There is a region of overlap where some MBs encoded in the SKIP mode have the same value of S_{DIFF} as the MBs encoded in the P_{16x16} mode. It is evident that the number of MBs in the P_{16x16} mode below the S_{DIFF} value of 200 is much less compared to the number of MBs encoded in the SKIP mode for $S_{DIFF} \leq 200$. This value of S_{DIFF} is taken as a threshold T_S and the MBs that have S_{DIFF} below T_S are considered as candidates for SKIP mode. For Ice sequence, the number of MBs in the P_{16x16} mode with S_{DIFF} value below T_S(=200) is just 3.5% of the total MBs in the P_{16x16} mode. Similarly for the other two sequences, the percentage of the MBs in the P_{16x16} mode with the values of S_{DIFF} below 200 are very low. If these few MBs in the P_{16x16} mode are encoded in the SKIP mode the degradation in quality is marginal compared to the reduction in complexity as no ME is performed for these MBs. Hence, the value T_S=200 is taken as a conservative estimate and MBs with S_{DIFF} values below T_S are encoded in the SKIP mode.

In a MB, motion is sometimes concentrated in a small region and this region will have a small number of large valued residuals, however the S_{DIFF} for the MB could be below T_S. Such MBs usually get encoded in the P_{16x16} mode as is seen in the overlap region. Hence the MB is further examined to check whether all the residues in R_{DF} are low and may be considered temporally stationary. The residues in R_{DF} are further examined and if all the residues in R_{DF} have absolute values less than or equal to 3, then the MB qualifies for the SKIP mode and further ME is terminated. If this condition is not met but still the S_{DIFF} is below T_S, then the MB is encoded in the P_{16x16} mode. MBs that do not satisfy both these conditions are further tested for homogeneity.

4.5.2 Determination of Homogeneous Region

Homogeneous regions have similar spatial properties. Natural scenes often contain many objects and background on which different textures or patterns are observed. The degree of uniformity in these regions determine the homogeneity of the region. A homogeneous region is defined by a common unifying characteristic throughout the whole area: there are no abrupt changes in the gray levels in the image. There exist many techniques for detecting homogenous regions in a frame. Region growing, similarity measures among pixel values, edge information of the image are some of the many techniques available for homogeneous region detection. In this work, the detection of homogeneous region is based on the edge information in the frame as video objects have strong edges. Homogeneous regions will have low values of the edge amplitude. The edge map of a frame using the the Sobel operator has already been described in Section 4.1.1. The edge amplitudes ($Amp(D_{i,j})$ are calculated using (4.3). Next, the sum of the edge amplitudes ($S_{Amp(N)}$) for a MB is considered, where

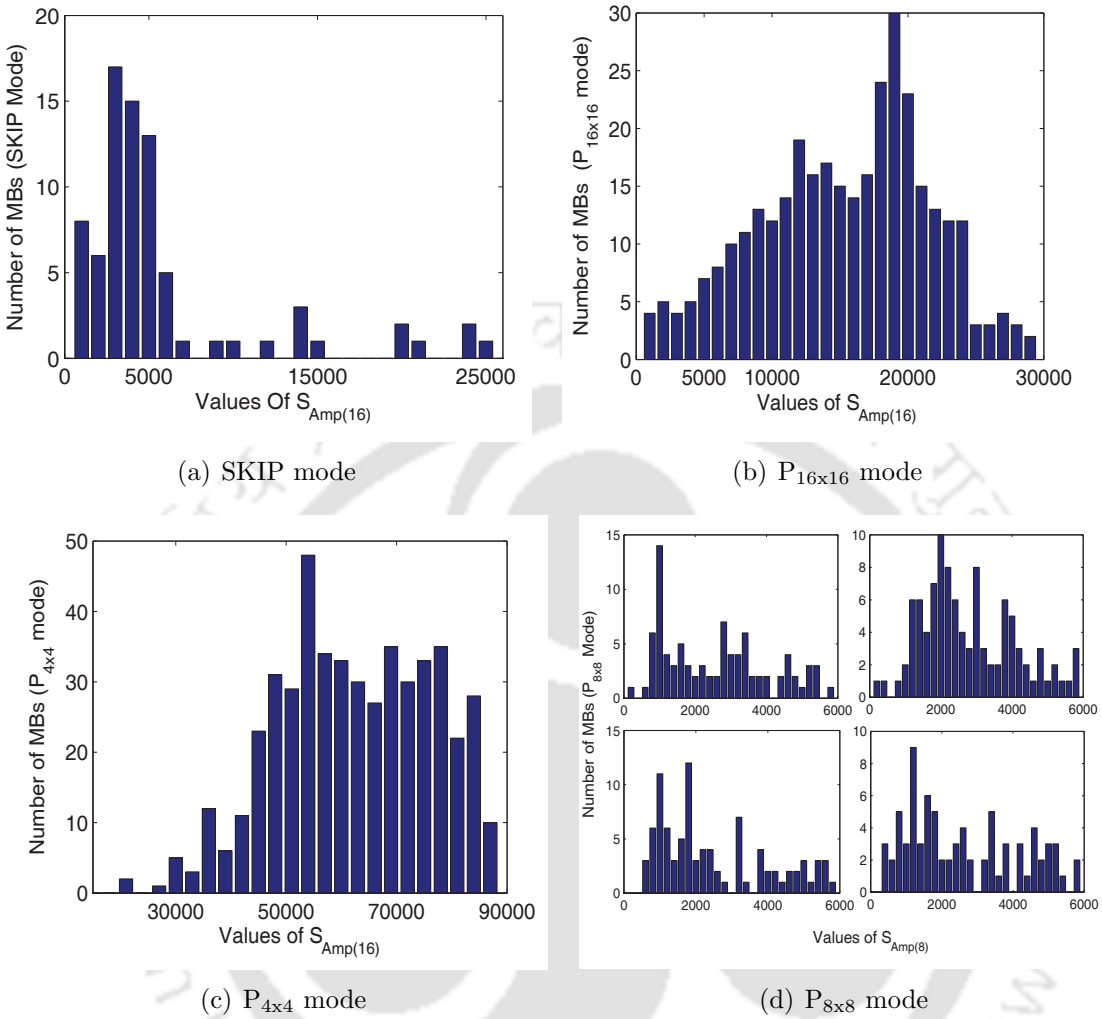
$$\mathrm{S_{Amp(N)}} \triangleq \sum_{\mathrm{i=1,j=1}}^{\mathrm{N,N}} \mathrm{abs(Amp(D_{i,j}))}, \quad \mathrm{N} = 16 \ \ \mathrm{or} \ \ 8 \tag{4.9}$$

Homogeneous regions have low values of edge amplitudes. Thus $S_{Amp(N)}$ will have low values for such regions. If $\mathrm{S_{Amp(N)}}$ is below a certain threshold ($\mathrm{T_{th(N)}}$) it is designated as a homogenous block otherwise it is nonhomogeneous. The decision is taken as follows:

$$Decision = \begin{cases} Homogeneous, & if \quad S_{Amp(N)} < T_{th(N)} \\ Non-homogeneous, & otherwise \end{cases}$$

Selection of Threshold $\mathrm{T_{th(N)}}$

The number of MBs in different modes as a function of $\mathrm{S_{Amp(N)}}$ for the first 10 frames of the Mobile sequence are shown is Figure 4.10. When encoded in the SKIP mode, most of the MBs lie to the left of $\mathrm{S_{Amp(16)}}$ value of 5000. For the $\mathrm{P_{16x16}}$ mode, most of the



(a) SKIP mode

(b) P_{16x16} mode

(c) P_{4x4} mode

(d) P_{8x8} mode

Figure 4.10: Distribution of number of MBs in different encoding modes with $S_{Amp(N)}$ for Mobile Sequence

MBs have values of $S_{Amp(16)}$ between 5000 and 25000. There are however a small number of MBs encoded with P_{4x4} mode that have $S_{Amp(16)}$ values below 25000. Therefore as a conservative measure, we take the threshold value $T_{th(16)}$ as 20000. Thus the MBs that are not skipped and having $S_{Amp(16)}$ value less than $T_{th(16)}$, are encoded in the mode that gives the lowest RD cost amongst P_{16x16}, P_{8x16} and P_{16x8} modes. Otherwise the MB is tested for homogeneity at the submacroblock level.

MBs that do not qualify for the larger block partitions as described above are divided into four submacroblocks each of 8×8 size. The homogeneity of each submacroblock is next examined. Figure 4.10(d) shows the MBs encoded in the P_{8x8} mode for all the

four submacroblocks. Most of the submacroblocks have the $S_{Amp(8)}$ values below 5000 and thus the $T_{th(8)}$ is taken as 5000. For a submacroblock with $S_{Amp(8)}$ less than T_{th8} is considered homogeneous at the submacroblock level and is encoded in the P_{8x8} mode. If for a submacroblock this is not satisfied, then directional ME is performed on the submacroblock and is described next.

4.5.3 Directional Motion Estimation

In Section 4.4.2 the relation between the prediction block size and the direction of motion in the frame was discussed. From Figure 4.6 it is seen that the partitioning of the MBs into different sizes are either horizontal or vertical. Hence if the direction of the motion in the MB can be determined early, only those modes pertaining to the motion direction may be considered for ME. Since the computational complexity for smaller block size ME is more, the directional approach is applied to those submacroblocks which are nonhomogeneous and have not satisfied the conditions for stationarity and homogeneity. The R_{DF} of the collocated MB in the difference frame will contain the information of the changes between two consecutive frames. The direction of edges in R_{DF} give an indication of the possible motion direction in the MB. The R_{DF} is divided into four 8×8 submacroblocks. The edge detection is done on the collocated residual submacroblock in R_{DF}. The direction of the edge is determined for this submacroblock. The direction of the edge is determined by the hyper-function

$$Ang(D_{i,j}) = \frac{180^\circ}{\pi} \times \arctan(\frac{dy_{i,j}}{dx_{i,j}}); \qquad |Ang(D_{i,j})| < 90^\circ \tag{4.10}$$

We divide the whole area of the submacroblock into three regions namely horizontal, vertical and plane region as shown in Figure 4.11. The sum of the amplitudes of the edge histogram in the these three directions are determined. The region in which this amplitude is maximum is taken as the dominant direction. The modes selected for different regions are as given in Table 4.6.

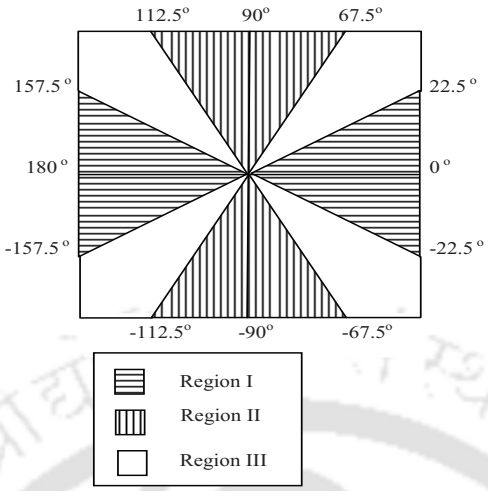


Figure 4.11: Partitioning into different regions

Table 4.6: Mode Selection for Different Regions

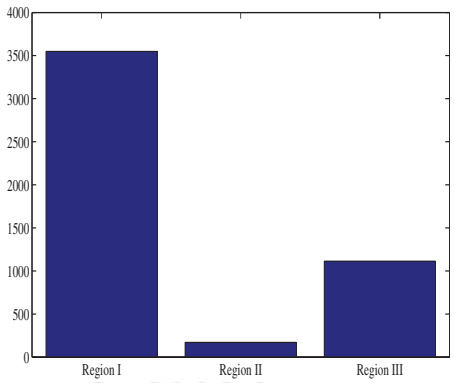
Region	Modes Selected
I	P_{8x8}, P_{4x8}
II	P_{8x8}, P_{8x4}
III	P_{8x8}, P_{8x4} P_{4x8}, P_{4x4}

A sample histogram generated for different regions in a submacroblock for the difference frame is given in Figure 4.12. Region I pertain to horizontal edge, Region II to vertical edge and Region III pertain to diagonal edges respectively.

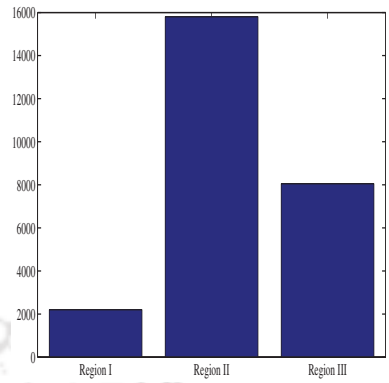
4.5.4 Overall Inter Prediction in EHFMD Algorithm

In the proposed EHFMD algorithm, the mode decision process starts with the determination of stationarity of a MB. If the MB is not stationary, then the homogeneity is next tested. If the MB is found to be non homogeneous then directional motion estimation is performed for the smaller block sizes. The overall procedure for the proposed method for fast inter mode decision is summarized below:

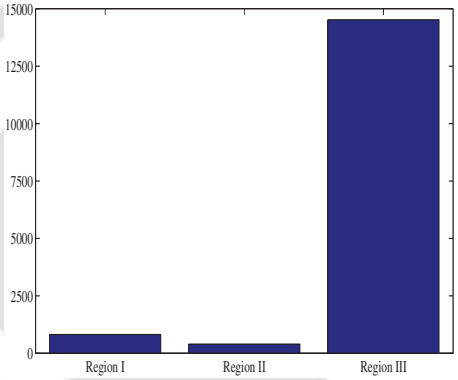
(i) For each MB in the current frame, subtract the collocated MB in the reference frame to get the residual difference R_{DF} and calculate S_{DIFF}.



(a) Edge pertaining to Region I



(b) Edge pertaining to Region II



(c) Edge pertaining to Region III

Figure 4.12: Typical edge histograms for the frame difference image

(ii) If $S_{DIFF} < T_S$ and all residues in R_{DF} are ≤ 3, encode MB in the SKIP mode. Go to step (x) . Else

(iii) If $S_{DIFF} < T_S$, but one or more residues in R_{DF} are > 3, encode the MB in P_{16x16} mode. Go to step (x). Else

(iv) As the MB is non stationary, generate the edge amplitude histogram.

(v) If $S_{Amp(16)} < T_{th(16)}$ then MB is homogenous. Perform ME with P_{16x16}, P_{16x8} and P_{8x16} modes. Select the mode that minimize the RDO cost. Go to step (x). Else

(vi) Divide the MB into four 8×8 submacroblocks.

(vii) For each 8×8 submacroblock, if $S_{Amp(8)} < T_{th(8)}$ then the submacroblock is homogenous. Perform ME with P_{8x8} mode.

(viii) Otherwise, perform edge detection on the collocated submacroblock in R_{DF} and calculate the edge amplitudes in the three defined directions. Let the maximum of these three edge amplitudes be denoted by M1.

(a) If M1 occurs in Region I, perform ME with P_{8x8} and P_{4x8} modes. Select the mode that minimizes the RDO cost.

(b) If M1 occurs in Region II, perform ME with P_{8x8} and P_{8x4} modes. Select the mode that minimize the RDO cost.

(c) If M1 occurs in Region III, perform ME with P_{8x8}, P_{8x4}, P_{4x8} and P_{4x4}modes. Select the mode that minimizes the RDO cost.

(ix) Repeat steps (vii) and (viii) for all subblocks.

(x) Proceed with the next MB.

4.6 Experimental Results

The overall EHFMD algorithm consists of both fast intra and inter prediction processes. A MB in I frame is encoded in the prediction process as described in Section 4.5.3 whereas a MB in the P frame is encoded as per the algorithm in Section 4.5.4 . This Section details the results of the experiment performed to evaluate the proposed EHFMD algorithm.

4.6.1 *Distortion and Compression Ratio Comparisons*

Table 4.7 lists the performance of the proposed algorithm in comparison to JM12.4 and the algorithms due to Wu [87] and Liu [72]. The results are arranged for different classes of sequences. The proposed method shows very little degradation in the PSNR compared to the other methods. The average bitrate has increased in all the methods as compared to the JM12.4 reference software. However the increase in bitrate with our method is 0.75% for the CIF sequences as against 1.15% [87] and 1.14% [72]. Similar

Table 4.7: Performance Comparison For Different Sequences

Class	Sequence	Performance Comparison								
		EHFMD			Wu et al.'s [87]			Liu et al.'s [72]		
		ΔPSNR dB	ΔRate (%)	ΔT (%)	ΔPSNR dB	ΔRate (%)	ΔT (%)	ΔPSNR dB	ΔRate (%)	ΔT (%)
CIF Class A	News	-0.10	-0.08	76.00	0.02	0.84	39.23	0.11	2.41	48.53
	MaD	0.02	0.72	73.98	0.01	0.62	43.21	0.09	1.76	55.25
	Container	0.04	1.06	63.32	0.05	1.21	46.18	0.08	1.61	52.59
	Hall	0.05	1.14	63.90	0.06	1.27	34.67	0.07	1.35	46.27
Class B	Foreman	0.05	1.18	64.96	0.07	1.32	34.90	0.05	1.04	39.86
	Coastguard	-0.01	-0.04	43.48	0.05	1.18	26.30	0.02	0.71	33.56
	Ice	0.03	0.86	77.11	0.07	1.29	45.68	0.05	1.09	54.12
	Harbour	0.01	0.64	30.36	0.05	1.15	21.56	0.06	1.12	23.26
Class C	Flower	0.02	0.82	52.41	0.05	1.17	36.85	0.06	1.16	42.86
	Stefan	0.09	1.52	40.72	0.02	0.84	32.25	0.08	1.59	39.54
	Tempete	-0.01	-0.02	42.68	0.09	1.48	27.21	0.09	1.63	35.42
	Mobile	0.04	1.30	38.62	0.09	1.54	12.35	0.07	1.45	19.28
	Average	**0.01**	**0.75**	**55.62**	**0.05**	**1.15**	**33.36**	**0.06**	**1.41**	**41.71**
QCIF Class A	Claire	-0.08	-1.01	83.82	-0.01	-0.95	47.35	0.01	0.51	62.45
	MissAmerica	-0.01	-0.56	79.67	0.02	0.81	48.23	0.03	0.89	60.27
	Suzie	0.05	1.14	75.05	0.05	1.24	42.91	0.06	1.32	59.65
Class B	Foreman	0.04	1.10	52.24	0.02	0.85	30.25	0.05	1.21	41.18
	Silent	0.09	1.59	73.12	0.06	1.41	42.62	0.03	0.91	55.24
	Crew	0.07	1.42	62.90	0.05	1.20	19.64	0.04	1.12	30.43
Class C	Football	0.04	1.05	53.18	0.05	1.24	28.84	0.05	1.26	31.75
	Mobile	0.07	1.31	28.08	0.07	1.39	15.32	0.08	1.71	21.29
	Soccer	0.04	1.08	61.15	0.03	0.95	20.19	0.04	1.10	27.65
	Average	**0.03**	**0.79**	**63.24**	**0.03**	**0.90**	**33.21**	**0.04**	**1.11**	**43.32**

ΔPSNR(+/-): picture quality loss/gain measured in dB
ΔRate(+/-): bitrate increase/decrease measured as a %
ΔT(+/-): encoding time saving/loss measured as a %

observations hold for the QCIF sequences. Figure 4.13 shows the RD curves for different sequences of each Class in the QCIF and the CIF resolution.

4.6.2 *Computational Speedup*

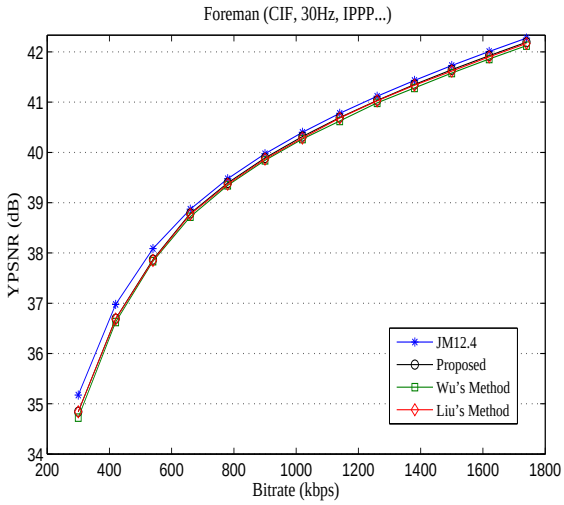
Table 4.7 also shows the percentage reduction in encoding time ΔT(%) for sequences of different classes. The time saving obtained depends upon the type of sequence. An increased saving (upto 83%) is noted for Class A sequences (QCIF) whereas the time saving for Class C sequences is comparatively low. This is due to the fact that Class A sequences have low motion complexity and hence a large number of MBs get encoded with larger block sizes. Our method gives a better overall encoding time saving compared to the reference software and the two algorithms with which our results have been compared. Although the degree of saving varies with the class of sequence, the proposed algorithm

exhibits a good overall computational saving. Class A and B sequences benefit more from the SKIP mode detection by detecting the stationary regions whereas Class C sequences benefit more from further examination of the directional features for smaller subblocks.

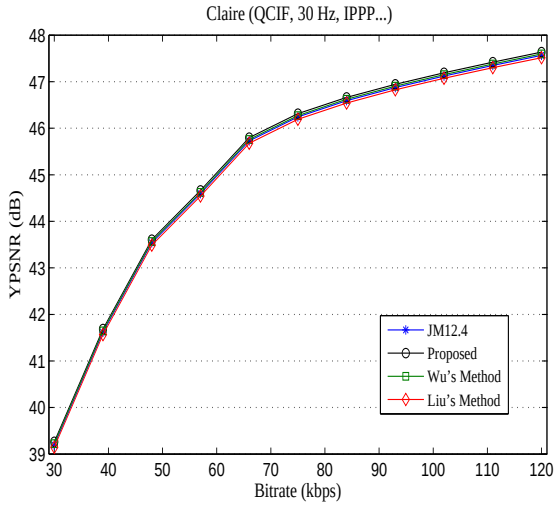
4.7 Conclusions

An attempt has been made in this chapter to reduce the encoding time and the complexity of the H.264/AVC standard using the edge features of the image. The encoding complexity of the H.264/AVC standard depends upon the nature of the video sequence. A detailed study of the full search intra and inter prediction process has been presented in the chapter. From an analysis of the various factors that affect the mode decision process, this chapter proposes a spatial-domain approach for fast mode decision for H.264/AVC. We have observed that the edge information present in a image gives a good indication of the nature of the image. This chapter proposed an Edge Histogram based Fast Mode Decision (EHFMD) algorithm which consists of two parts: one for fast encoding of the intra (I) frames and the other for fast mode selection process for the inter (P) frames. The intra prediction of EHFMD is dependent on QP. The mode decision was taken based on the direction of the edges in the MB and the determination of the homogeneity from the RC calculation. The inter prediction process of EHFMD finds the skipped MBs from the residue images. From the edge amplitude histogram the homogeneous regions were determined. These factors were utilized for the fast ME process. The experimental results and the RD curves have shown the effectiveness of the EHFMD algorithm.

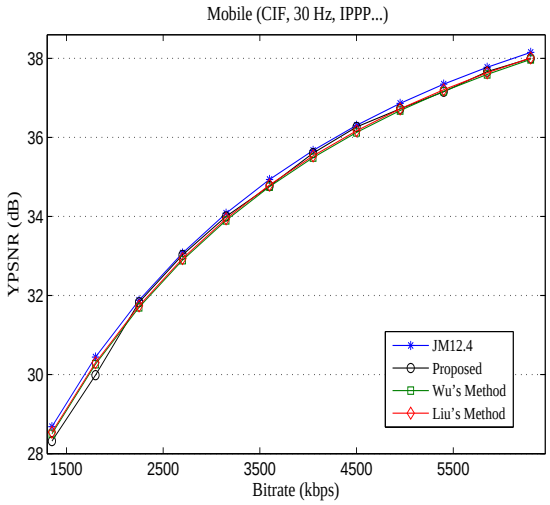
The overall analysis of the videos showed that many MBs are encoded in the SKIP mode. The next chapter presents a method for fast mode decision based on early detection of the SKIP mode.



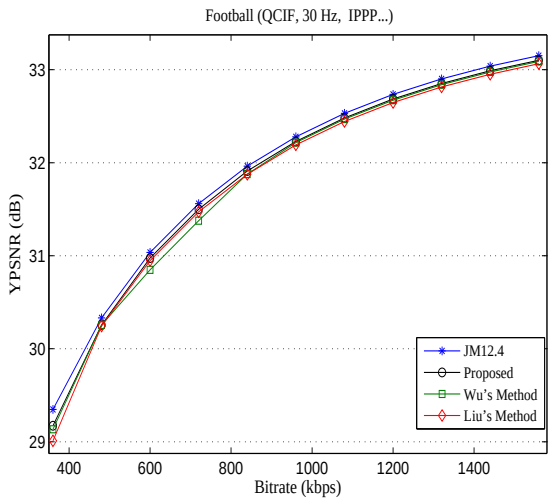
(a) RD Performance for Foreman (CIF) Sequence



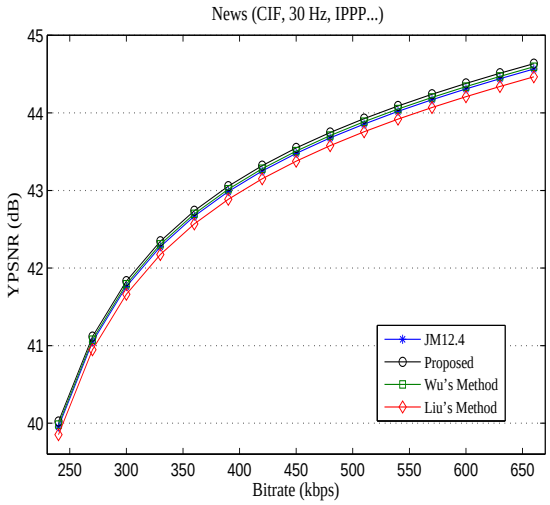
(b) RD Performance for Claire(QCIF) Sequence



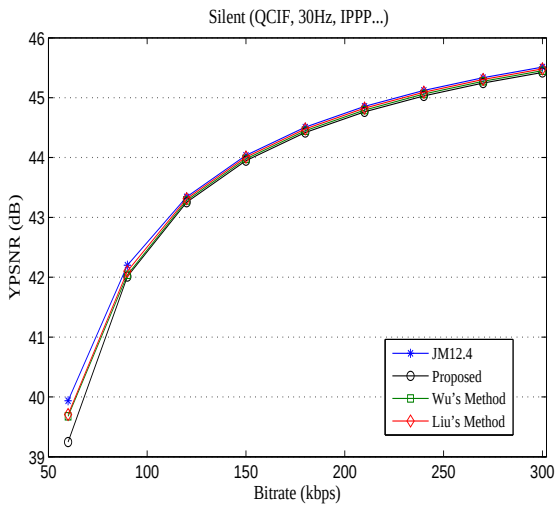
(c) RD Performance for Mobile(CIF) Sequence



(d) RD Performance for Football(QCIF) Sequence



(e) RD Performance for News(CIF) Sequence



(f) RD Performance for Silent(QCIF) Sequence

Figure 4.13: RD Performance for various sequence in the CIF and the QCIF resolution

5

Fast Mode Decision based on Early Zero Block Detection and Weighted Prediction

Contents

5.1 Early Zero Block Detection based Fast Mode Decision (EZBD-FMD) 83
5.2 EZBD-FMD Algorithm 89
5.3 Experimental Observations 95
5.4 Mode Decision Using Weighted Prediction 103
5.5 Experimental Observations 108
5.6 Conclusions 109

5. Fast Mode Decision based on Early Zero Block Detection and Weighted Prediction

In the previous chapter, we observe that there are many blocks encoded in the SKIP mode. Encoding a MB in the SKIP mode requires very little computations. In this chapter further attempt is made to study in detail the factors that help in recognizing the MBs that are the probable candidates for the SKIP mode. Since this will be done prior to performing the ME process, computational savings are achieved.

Macroblocks encoded in the SKIP mode have all the transform coefficients quantized to zero. H.264/AVC uses 4×4 transform and if all the transform coefficients of the 4×4 block are quantized to zero, it is called a Zero Block (ZB). If this is true for all the sixteen subblocks in a MB, it is called an All Zero Coefficient Block (AZCB). Early prediction of SKIP mode employs early detection algorithms to estimate the number of ZBs in a MB. These algorithms try to predict the SKIP mode beforehand and save unnecessary mode calculations. This approach is appealing as the computation of the SKIP mode is very simple as compared to the other modes. Several early detection based fast algorithms have been reported in literature. Sousa [82] proposed a general method to determine ZBs early for 8x8 blocks. Kim et al. [61], [83] proposed a threshold based early detection of SKIP mode using theoretical analysis of the integer transform and quantization of the residuals. In [85], Wang et al. proposed adaptive thresholds to detect AZCB and terminate the ME early. Park et al. in [75] proposed a fast mode decision algorithm based on the residue image. An efficient method for early detection of ZBs depending upon the QP exceeding a threshold value is described by Zhang et al. [90].

The rest of the chapter is organized as follows. The chapter is divided into two parts. The first part describes a fast inter mode decision process based on early detection of SKIP mode. The sufficient condition for the detection of an AZCB is described in detail. The second part of the chapter focuses on a weighted inter mode decision process. The chapter ends with results and conclusions.

5.1 Early Zero Block Detection based Fast Mode Decision (EZBD-FMD)

5.1.1 Further Observations from Full Search ME

The full search statistics for H.264/AVC has already been dealt in detail in Chapter 4. The encoding mode for the inter prediction process depends largely on the motion content between subsequent frames, the texture of the frame and the QP selected for encoding. Natural videos have large areas with homogeneous motion that results in a large number of MBs being encoded with larger block sizes. The dependence of the intra prediction mode on the QP was illustrated in the last chapter. For high values of QP, more MBs tend to be encoded with larger block sizes.

From the observations on inter prediction process, we have seen in Table 4.5 the distribution of MBs (in %) encoded in different modes for a QP value of 32 for sequences of different classes. Through analysis of the encoding modes at different QP, it is seen that the distribution of the modes is dependent on the QP. As the QP increases, the percentage of the MBs encoded with larger block size increases. At high values of QP, more MBs are encoded with larger block sizes. Figure 5.1 shows the distribution of modes for QP values of 20 and 32. For Class A sequence News, it is seen that there are many MBs in SKIP mode even at low QP. The % of MBs in SKIP mode are lower for sequences from the other two classes. Further, the distribution clearly indicates that as QP increases, the percentage of MBs encoded with larger block sizes also increases.

From these observations we note that there are sizeable number of MBs for sequences of all classes that are encoded in the SKIP mode. Since SKIP mode does not entail any motion estimation computation, there is large saving in complexity if the MBs to be encoded in the SKIP mode is determined from the characteristics of the MB and skipping the full search for the MB. Thus, detecting SKIP mode early thereby saving unnecessary mode calculations is one way to reduce the encoder complexity. The SKIP mode detection

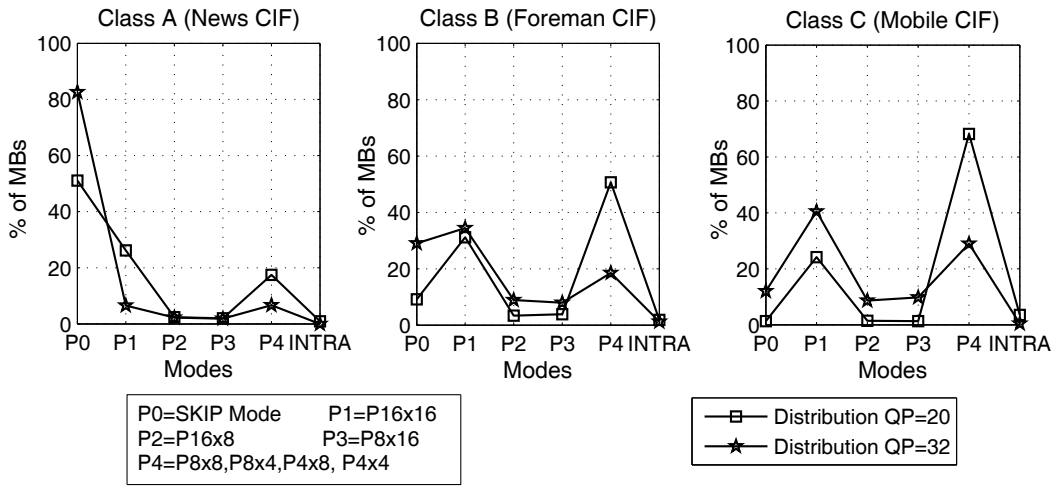


Figure 5.1: Distribution of modes for different QP

is dealt in detail in the next section.

5.1.2 SKIP mode detection from frame difference residues

The H.264/AVC encoder identifies certain MBs as skipped during encoding. For a MB to be encoded in the SKIP mode the following conditions are to be satisfied [48]:

(a) The best motion compensation block size is 16×16.

(b) The reference frame is the previous frame.

(c) The motion vector is zero or equal to the predicted motion vector.

(d) The transform coefficients for the MB are all quantized to zero.

If a MB is skipped by the encoder, the decoder reconstructs the MB by using the predicted motion vector for the MB by motion-compensated prediction from the current reference picture. In doing so, the skip mode decision is made only after 16×16 block motion estimation.

Condition (d) given above can be utilized for early detection of the SKIP mode. If all the transformed and quantized coefficient in a MB are zero, then it is an indication that it may be a skipped MB. For this purpose, the transformed and quantized coefficients

are estimated. H.264/AVC uses 4×4 integer transforms and if all the coefficients in the block are zero, we call it a zero block (ZB). Early prediction of SKIP mode employs early detection algorithms to estimate the number of ZBs in a MB. If all the sixteen blocks in a MB are ZBs it is called an all zero coefficients block (AZCB) and the MB will be a candidate for the SKIP mode. These algorithms try to predict the SKIP mode beforehand and save unnecessary mode calculations. This method is appealing as the computation of the SKIP mode is very simple as compared to the other modes.

We have already observed that the stationary regions and regions with very little motion usually get encoded in the SKIP mode. From condition (b), we see that the reference frame for the SKIP mode is the previous frame. Thus the simplest method of temporal prediction is to use the previous frame as the prediction for the current frame.

Figure 4.8 shows the residual frame formed by subtracting frame 10 from frame 11 of the Mother and Daughter (MaD) sequence. No motion estimation is performed to produce the difference frame. The frame difference residual block R_{DF} was obtained in Section 4.5.1 of Chapter 4 and is used here for early determination of the SKIP mode.

After performing full search on different classes of sequences, the MBs which are encoded in the SKIP mode and the corresponding residues in R_{DF} are studied at different QPs . Table 5.1 shows the distribution of these residues whose values are below a certain quantity for different QP values (16, 24 and 36). The entries in the table are the percentage of residues averaged over all MBs in SKIP mode for the sequences of a particular class. We have already observed in Chapter 4 that the residues in R_{DF} for skipped MBs have low values. Here we further show that more than 90 % of the values of the residues in R_{DF} for the skipped MBs are below 3 for low QPs.

Thus from the study of the residuals and the MBs encoded in different modes, following observations are made:

1. For MBs encoded in SKIP mode

- The residual values in R_{DF} for skipped MBs are low.
- Most of the residues (>90% on an average) in R_{DF} have values below '3'.

Table 5.1: Relation between Residues per MB and SKIP mode

Type	QP	% of residues per MB in R_{DF} with residues					
		0	≥1 and <2	≥2 and <3	≥3 and <4	≥4 and <5	≥5
Class A	16	29.25	39.45	23.76	4.15	2.15	1.20
	24	28.44	36.31	20.12	9.16	4.36	1.61
	36	24.77	34.35	20.16	10.11	5.74	4.87
Class B	16	29.46	38.32	21.15	5.74	3.14	2.19
	24	28.79	31.53	19.80	13.58	1.01	5.29
	36	19.81	34.71	18.13	9.09	3.95	13.31
Class C	16	28.56	38.36	24.79	5.21	1.66	1.42
	24	31.51	39.74	18.06	6.83	2.67	1.19
	36	30.34	25.96	15.15	7.65	12.42	8.48

• Few residues in skipped MBs have values greater than 5.

• As the QP increases, more residues have values above 3. In other words, as QP increases, a MB may be encoded in the SKIP mode even if there are residues with large values.

2. For MBs encoded in other modes

• Even when a MB is encoded in other modes, there are many residues with values ≤ 3, however their percentage is much lower compared to that for the MB in the SKIP mode.

• There are quite a few residues with very high values specially for the light and dark grey regions in the difference frame.

We have used the frame difference residues as a very basic filtering criterion to determine the MBs that could be candidates for the SKIP mode. But this criteria cannot solely be used for ascertaining the MBs to be skipped. The MBs that pass through this filtering operation are further analyzed to ascertain whether they should indeed be encoded in the SKIP mode.

5.1.3 Early Detection of AZCB for H.264/AVC

The conditions for a MB to be in the SKIP mode is mentioned in Section 5.1.2. Condition (d) states that one of the criteria for a MB to be encoded in the SKIP mode is that

the transform coefficients of the residual block are all quantized to zero. In H.264/AVC, the transformation and quantization processes are designed to minimize computational complexity. The transformation and quantization process has been dealt in detail in Chapter 2. The basic transform in H.264/AVC is a 4×4 integer transform which is a scaled approximation to the Discrete Cosine Transform (DCT). If all the transform coefficients of the residual block are zero, the block is classified as a AZCB. This satisfies the condition (d) wherein a MB encoded in the SKIP mode has all the transform coefficients quantized to zero.

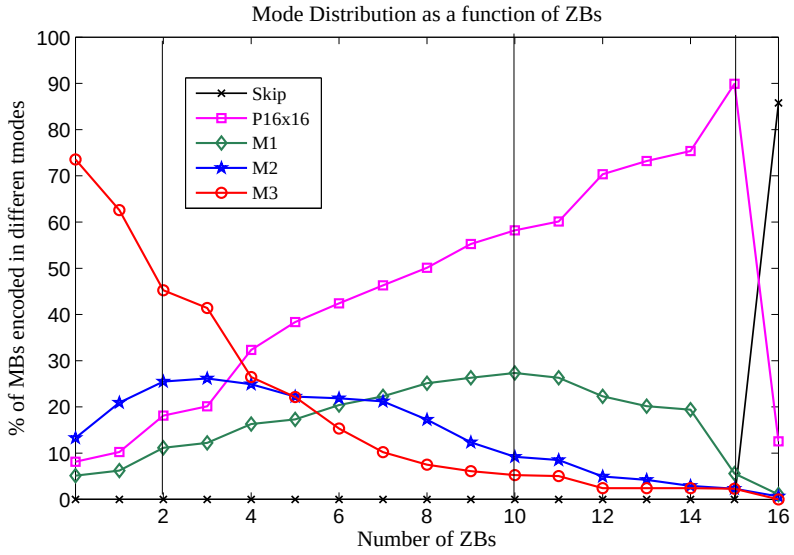


Figure 5.2: Relation between number of ZBs and selected mode

From full search ME on sequences of different classes, the percentage of MBs that are encoded in different modes and the corresponding number of ZBs in these MBs is studied. We note that there is a relation between the number of ZBs and the final mode selected for encoding. Figure 5.2 shows the percentage of MBs encoded in different modes as a function of the number of ZBs. Here, M1 includes the P_{16x8} and P_{8x16} modes, M2 includes the P_{8x8}, P_{8x4} and P_{4x8} modes and M3 includes the P_{4x4} and INTRA modes. Figure 5.2 clearly shows that when the number of ZBs is 16 which is a AZCB, the dominant mode is the SKIP mode and there are only a few MBs encoded in the P_{16x16} and other modes. Further, the occurrence frequency of the M2 and M3 modes is very low. Between the range

with 10 to 15 ZBs, the P_{16x16} and the M1 modes compete but the occurrence frequency of the M2 and M3 modes is still low. When the number of ZBs are between 2 to 9, all the modes have almost equal probability of occurrence. For ZBs lower than 2, the MBs tend to get encoded in the M1 mode. Thus, the number of ZBs in a MB may give an indication of the mode or a subset of the modes to be searched for ME. If the ZBs in the residual MB are detected early, then decision for the SKIP and other modes become less time consuming. The sufficient condition for a MB to be a ZB is described next.

The 4×4 integer transform used in H.264/AVC is described here. For a residual block e(x, y), the integer transform can be defined as

$$E_I(u, v) = \sum_{x=0}^{3}\sum_{y=0}^{3} e(x, y) \cdot A(x, u) \cdot A(y, v), \tag{5.1}$$

where

$$A(m, n) \triangleq \left\lceil \frac{2.5C(n)}{\sqrt{2}} \cos \frac{(2m + 1)n\pi}{8} \right\rceil, \tag{5.2}$$

$\lceil\ \rceil$ is the rounding off operation, $C(n) = 1/\sqrt{2}$ for $n = 0$ and $C(n) = 1$ otherwise.

The quantized coefficient $E_q(u, v, r, QP)$ for QP ranging from 0-51 is given by

$$E_q(u, v, r, QP) = \text{sign}\{E_I(u, v)\} \times \frac{|E_I(u, v)| \cdot M(QP\%6, r) + f}{2^{qbits}}, \tag{5.3}$$

Here $qbits = 15 + \text{floor}(QP/6)$ and $f = \text{floor}(2^{qbits/6})$ for inter blocks. $M(QP\%6, r)$ is the multiplying factor associated with the coefficient position and the QP. It is a part of the transform and is predefined for each coefficient position.

$$M(QP\%6, r) = \begin{bmatrix} 5273 & 8066 & 13107 \\ 4660 & 7490 & 11916 \\ 4194 & 6554 & 10082 \\ 3647 & 5825 & 9362 \\ 3355 & 5243 & 8192 \\ 2893 & 4559 & 7282 \end{bmatrix}, \tag{5.4}$$

where $r = 2 - (u\%2) - (v\%2)$.

The absolute value of the transform coefficient $E_I(u, v)$ is limited by

$$|E_I(u, v)| \leq \sum_{x=0}^{3} \sum_{y=0}^{3} |e(x, y)|. |C(x, y, u, v)|, \tag{5.5}$$

$C(x, y, u, v)$ is the basis part of (5.1). From (5.3), we note that if $E_q(u, v, r, QP)$ is less than 1, then it will be zero valued after quantization. Hence the following inequality is easily obtained.

$$|E_q(u, v, r, QP)| \leq \frac{f + 4 \cdot M(QP\%6, 0) \cdot \sum_{x=0}^{3} \sum_{y=0}^{3} |e(x, y)|}{2^{qbits}}. \tag{5.6}$$

All the transform coefficients will be quantized to zero if the upper bound in (5.6) is smaller than 1. Hence the sufficient condition for all transform coefficients to be quantized to zero is given by

$$\sum_{x=0}^{3} \sum_{y=0}^{3} |e(x, y)| \leq \frac{2^{qbits} - f}{M(QP\%6, 0)} \cdot \frac{1}{4} \triangleq T_S. \tag{5.7}$$

The left-hand-side of (5.7) is the sum of all the residuals in the 4×4 block. Thus, if this sum is below the threshold T_S, then all the transform coefficients will be zero. Thus for a subblock if this condition is satisfied, it is regarded as a ZB. Hence the early detection of ZB is possible.

5.2 EZBD-FMD Algorithm

We propose an Early Zero Block Detection based Fast Mode Decision (EZBD-FMD) algorithm for reducing the complexity and time of the encoding process. The decision process involves the identification of the MBs that have the properties to be encoded in the SKIP mode. The EZBD-FMD algorithm is implemented at two levels - Level A and Level B. Level A deals with SKIP mode identification process based on the frame difference residues. Level B comprises of early detection of ZBs on an approximated

motion compensated residual MB using predicted MVs. The mode decision depends on the number of ZBs in a MB.

According to condition (b) for SKIP mode as given in Section 5.1.2, the previous frame is the reference frame. Thus, the previous frame is used for the frame difference and this residues give an indication of the changes in the neighboring frames. These residues are studied to find the MBs which have very low residue values. If all the residues are below a threshold, the MB is encoded in the SKIP mode. Otherwise, the number of ZBs in a MB is determined using the analysis given in the last section and decision on the encoding mode is taken according to the number of ZBs.

5.2.1 *Level A: Early SKIP mode Decision*

Level A of the EZBD-FMD algorithm attempts to identify the MBs that can be coded in the SKIP mode. This decision is taken without utilizing the ME process. The strategy behind SKIP mode decision for a MB is to find the number of residues that are below a threshold in the collocated frame difference block R_{DF}. For MBs encoded in the SKIP mode, it is noted in Section 5.1.2 that more then 90% of the residues in R_{DF} have values less than or equal to '3'. This value is taken as the threshold for finding the MBs that are candidates for the SKIP mode. For low values of QP, there are more residues below this value of threshold. For all classes of sequences this value is a conservative estimate as it gives consistent results for all ranges of QP. There are several instances when more than 90% of the residues are below this threshold but some of the remaining residues some may have large values which indicates that the MB may contain some motion and is not a skipped MB. Thus we define an upper bound (< 5) for the remaining residues and only those MBs which have all residuals below this value are considered for the SKIP mode. Thus, the MBs with all residues <5 and 90% or more residues ≤ 3 are identified for encoding in the SKIP mode. Figure 5.3 shows the flow chart for Level A.

Let C be the number of residues in R_{DF} whose absolute values ≤ 3 and M be the maximum value of the residues in $|R_{DF}|$. If C is greater than 230 ($\sim$ 90% of the total

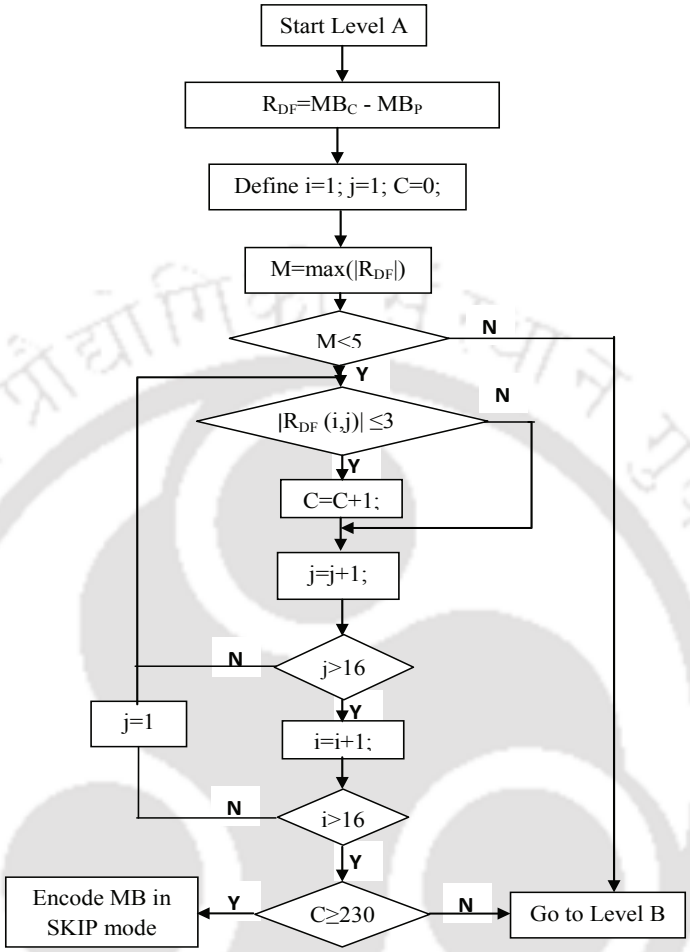


Figure 5.3: Flowchart for Level A for SKIP mode detection

residues) and M is < 5, then the MB is encoded in the SKIP mode and the ME is terminated early. Otherwise, we proceed to the next level. The decision process for SKIP detection is defined as

$$\text{Decision}_{\text{SKIP}} = \begin{cases} \text{SKIP} & \text{if} \quad \text{C} \geq 230 \quad \& \quad \text{M} < 5 \\ \text{LevelB} & \text{otherwise.} \end{cases}$$

A fast mode decision approach based on frame difference residues is reported in [101] where the decisions on modes are taken based on the mean of absolute difference for a MB compared to the mean of absolute difference for the entire frame. If the mean of absolute difference for a MB is less than the mean of absolute difference for a frame, larger sizes

are selected. Else, full search is done on the MB. The drawback of this method is that a MB containing high values of residuals (indicating large motion) in only a small region may give low value of the mean. These MBs sometime get encoded incorrectly with larger block sizes. To overcome this problem, we have analyzed each of the MBs independently. The threshold value of 3 is taken to ensure that MBs with only low residuals get encoded in the SKIP mode. Since larger residues indicate large motion, the second condition that all residues be below M further ensures that a MB with more motion concentrated in only a small region will also not go undetected. It may be noted that in the full search in H.264/AVC, as the QP increases, there will be some MBs encoded in SKIP mode that have large valued residues in R_{DF}. Sometime, quite a few residues have values greater than the threshold M. We have used the same threshold for all QPs. Thus MBs which otherwise would have been in the SKIP mode do not qualify for the SKIP mode at this level. These MBs spill over to Level B that further examines whether they should indeed be encoded in the SKIP mode. Thus with the EZBD-FMD algorithm, no degradation is observed for low bit rate encoding.

5.2.2 *Level B: Early Detection of ZBs*

According to condition (c), the best motion vector is the predicted motion vector i.e the motion vector difference is zero. Condition (d) states that the transform coefficients for the MB are all quantized to zero. When a MB does not qualify for the SKIP mode at Level A, then the process of ensuring whether it should be in SKIP mode is done in Level B. At this level, early detection of ZBs in the MB is done. Since the best MV for the SKIP mode is the predicted MV, the motion compensated block is obtained using the predicted MV. The predicted motion vectors (pmv) for a MB are obtained from the MVs of the neighboring MBs that have been encoded and transmitted. It may be noted that no ME is performed at this level. An approximate motion compensated residual block ($\hat{E}_{MC}$) is first obtained by subtracting the motion compensated MB using the pmvs and the original MB as given below

$$\hat{E}_{MC}(i,j) = \left\| f_t(x+i, y+j) - \hat{f}_{t-1}(x+i-pmv_x, y+j-pmv_y) \right\|, \tag{5.8}$$

.

where f_t is the frame at time t and $\hat{f}_{t-1}$ is the reconstructed previous frame, i and j are the pixel positions in the MB, x and y are the positions of the MB in the horizontal and vertical directions and pmv_x and pmv_y are the respective predicted MVs.

In H.264/AVC, transformation and quantization is performed on 4×4 blocks. Thus $\hat{E}_{MC}$ is divided into sixteen 4x4 subblocks e_k, k=1,...,16. For each k, threshold T_S is calculated using (5.7). For the k^{th} subblock, the sum of the residuals S_k for the subblock e_k, is determined by

$$S_k = \sum_{x=0}^{3} \sum_{y=0}^{3} |e_k(x,y)|, \quad k = 1, \ldots, 16. \tag{5.9}$$

The decision on S_k being a ZB is taken as per the condition in (5.7) and is given by,

$$\text{Decision} = \begin{cases} \text{ZB} & \text{if} \quad S_k \leq T_S \\ \text{Not a ZB} & \text{otherwise.} \end{cases}$$

Let N be the number of S_k that are ZBs. Based on the observations made in Section 5.1.3 the following is proposed:

Case 1: Number of ZB in the MB, N=16 and the MB is a AZCB.

From Figure 5.2 we note that when the MB is an AZCB, there is a high probability that the MB may qualify for the SKIP mode. But, there are some MBs (around 10%) that get encoded in other modes. Out of these modes, probability of occurrence of small block size modes is negligible and the MBs get encoded mostly in the P_{16x16} mode. The degree of stationarity of the MB which is an AZCB is obtained by considering the SAD at zero MV ($SAD_{MV=0}$) given by

$$SAD_{MV=0} = \sum_{i=1}^{16} \sum_{j=1}^{16} \left| f_t(x+i, y+j) - \hat{f}_{t-1}(x+i, y+j) \right| \tag{5.10}$$

If this value of SAD is large, it indicates that this MB may have regions that contain some motion. Thus if encoded in SKIP mode, the distortion will increase. If instead P_{16x16} is chosen for this MB, the quality of the video is not affected. Thus, a threshold may be defined such that if the $SAD_{MV=0}$ is less than this threshold, the MB is encoded in the SKIP mode. Otherwise, P_{16x16} ME is performed on the MB.

From the condition given in (5.7), for each subblock the S_k should be less than T_S. As there are sixteen subblocks, a threshold T_{skip} is defined where

$$T_{skip} = 16 \times T_S \tag{5.11}$$

Thus, the decision for a MB which is an AZCB to be encoded in the SKIP or P_{16x16} mode is taken as

$$\text{Decision} = \begin{cases} \text{SKIP} & \text{if} \quad SAD_{MV=0} < T_{skip} \\ P_{16x16} & \text{otherwise.} \end{cases} \tag{5.12}$$

A MB is to be encoded in the SKIP mode, if it is an AZCB and satisfies (5.12). Furthermore, as T_S is a function of QP, the value of T_{skip} will vary with QP and will increase with increasing QP. Thus, MBs qualifying for SKIP mode at high QP may not qualify for SKIP mode at lower QP.

Case 2: Number of ZB in the MB, N=15

When N=15, there is only one subblock which have non zero values. From Figure 5.2 it is observed that the probability that the MB is encoded in the P_{16x16} mode is more than 90%. The probability of occurrence of the other modes is very less. Hence, only P_{16x16} ME is performed and the MB is encoded in the 16x16 mode.

Case 3: Number of ZB in the MB is in the range $10 < N < 15$

Here there are a few subblocks which have non zero coefficients. In this region,

we observe that the P_{16x16} and M1 mode which includes the P_{16x8} and P_{8x16} are prevalent. Hence ME is performed with P_{16x16}, P_{16x8} and P_{8x16} modes. The MB is finally encoded in the mode that minimize the RD cost amongst these modes.

Case 4: Number of ZB in the MB is in the range $2 < N \leq 10$

It may be mentioned here that these MBs will have many non zero coefficients. This will indicate the presence of motion in the MB. Under this condition, all the modes contribute to the mode decision process except the INTRA mode. Thus, ME is performed with all modes except the INTRA mode and the final mode is the one that minimizes the RD cost.

Case 5: Number of ZB in the MB $N \leq 2$

When at the most there are only two ZBs for the MB, there is a high likelihood that there is some amount of motion associated with this MB. Full search ME is done for these MBs. All the modes in addition to the INTRA mode is used for the prediction purpose. The selected mode is the one which gives the lowest RD cost.

5.3 Experimental Observations

Common test video sequences and the test conditions used for the implementation of the EZBD-FMD algorithm are as mentioned in Chapter 3. To evaluate the average encoding performance over a range of QPs, the differences in PSNR (ΔPSNR) in dB and bitrate (ΔRate (%)) are calculated according to numerical averages between RD curves as given by Bjontegaard [100].

5.3.1 *Distortion and Compression Ratio Comparisons*

The simulations on different sequences show that the mode decision process described here have similar rate distortion performance as that of the standard software JM12.4. The overall PSNR loss is negligible. The EZBD-FMD algorithm's distortion and bitrate performance is comparable to Wu's [87] and Liu 's [72] algorithms. However, the encoding

time saving for the given method is much higher compared to the other two algorithms mentioned here. Table 5.2 lists the performance of the EZBD-FMD in comparison to JM12.4 implementation and Wu's and Liu 's algorithms. The results are arranged for different classes of sequences. The search efficiency (or speed-up factor) is evaluated using the search point calculations following the procedure detailed in Chapter 3 (Section 3.4.2). The time saving obtained depends upon the type of sequence. An increased saving is noted for Class A sequences whereas time saving obtained for Class C sequences is comparatively low. For all sequences, the EZBD-FMD algorithm exhibits a good computational saving in terms of time and search points. Class A and B sequences benefit more from the SKIP mode detection whereas Class C sequences benefit more from the further examination of ZBs at Level B.

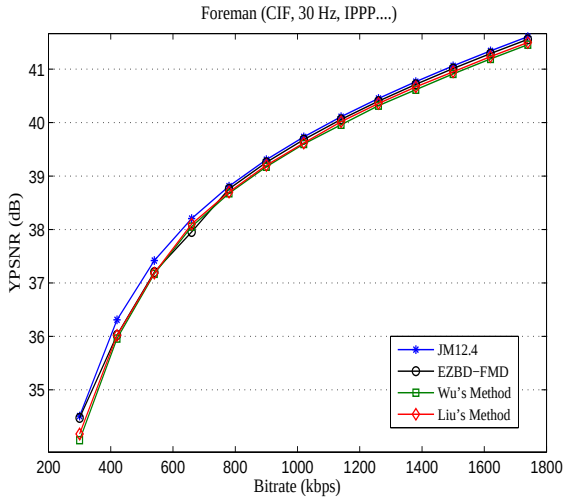
Table 5.2: Performance Comparison For Different Sequences

Class	Sequence	Performance Comparison											
		EZBD-FMD Algorithm				Wu *et al.*'s [87]				Liu *et al.*'s [72]			
		ΔPSNR dB	ΔRate (%)	ΔT (%)	ΔSP (%)	ΔPSNR dB	ΔRate (%)	ΔT (%)	ΔSP (%)	ΔPSNR dB	ΔRate (%)	ΔT (%)	ΔSP (%)
CIF Class A	News	0.04	0.96	84.24	84.54	0.02	0.84	39.23	32.15	0.11	2.41	48.53	50.83
	MaD	0.04	1.03	93.91	91.07	0.01	0.62	43.21	37.23	0.09	1.76	55.25	53.67
	Container	0.05	1.09	88.44	90.10	0.05	1.21	46.18	41.89	0.08	1.61	52.59	49.41
	Hall	0.07	1.27	82.12	80.15	0.06	1.27	34.67	36.24	0.07	1.35	46.27	42.89
Class B	Foreman	0.08	1.38	55.96	44.65	0.07	1.32	34.90	35.23	0.05	1.04	39.86	36.58
	Coastguard	0.05	1.05	35.13	32.34	0.05	1.18	30.30	30.53	0.02	0.71	33.56	31.42
	Ice	0.04	1.04	71.17	62.38	0.07	1.29	45.68	42.15	0.05	1.09	54.12	56.38
	Harbour	0.02	0.76	29.18	24.86	0.05	1.15	21.56	20.08	0.06	1.12	23.26	20.49
Class C	Flower	0.03	0.85	48.58	49.23	0.05	1.17	36.85	39.71	0.06	1.16	42.86	41.27
	Stefan	0.05	1.16	45.19	40.62	0.02	0.84	32.25	35.01	0.08	1.59	39.54	34.15
	Tempete	0.06	1.26	39.55	40.13	0.09	1.48	27.21	26.58	0.09	1.63	35.42	32.54
	Mobile	0.02	0.81	35.41	22.19	0.09	1.54	12.35	14.04	0.07	1.45	19.28	18.52
	Average	**0.04**	**1.05**	**59.07**	**62.40**	**0.05**	**1.15**	**33.36**	**32.57**	**0.06**	**1.41**	**40.87**	**39.01**
QCIF Class A	Claire	-0.04	-1.01	91.89	90.88	-0.01	-0.95	47.35	45.04	0.01	0.51	62.45	65.48
	MissAmerica	-0.02	-0.82	87.07	85.85	0.02	0.81	48.23	46.75	0.03	0.89	60.27	59.52
	Suzie	0.04	1.05	80.63	84.02	0.05	1.24	42.91	42.07	0.06	1.32	59.65	57.46
Class B	Foreman	0.05	1.14	41.25	39.87	0.02	0.85	30.25	28.85	0.05	1.21	41.18	37.51
	Silent	0.07	1.25	77.69	76.94	0.06	1.41	42.62	38.75	0.03	0.91	55.24	53.26
	Crew	0.03	0.93	40.32	36.40	0.05	1.20	19.64	16.41	0.04	1.12	30.43	32.15
Class C	Football	0.07	1.32	39.90	37.06	0.05	1.24	28.84	25.23	0.05	1.26	31.75	30.42
	Mobile	0.06	1.27	34.14	29.80	0.07	1.39	15.32	11.18	0.08	1.71	21.29	19.34
	Soccer	0.05	1.09	40.52	41.84	0.03	0.95	20.19	20.26	0.04	1.10	27.65	24.21
	Average	**0.03**	**0.69**	**58.18**	**58.07**	**0.03**	**0.90**	**33.21**	**30.48**	**0.04**	**1.11**	**43.32**	**42.15**

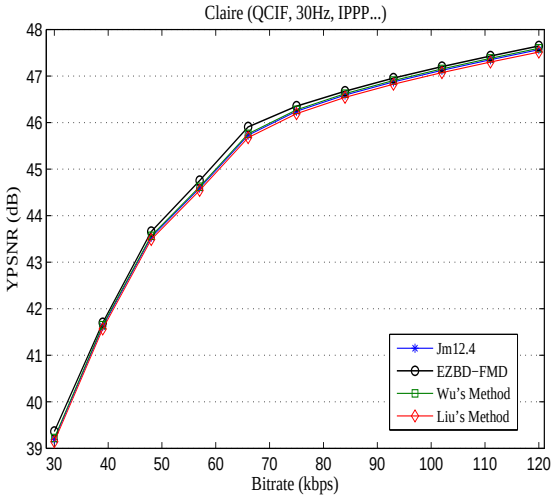
ΔPSNR(+/-): picture quality loss/gain measured in dB
ΔRate(+/-): bitrate increase/decrease measured as a %
ΔT(+/-): encoding time saving/loss measured as a %
ΔSP(+/-): search point saving/loss measured as a %

5.3.2 *PSNR-Rate Curves*

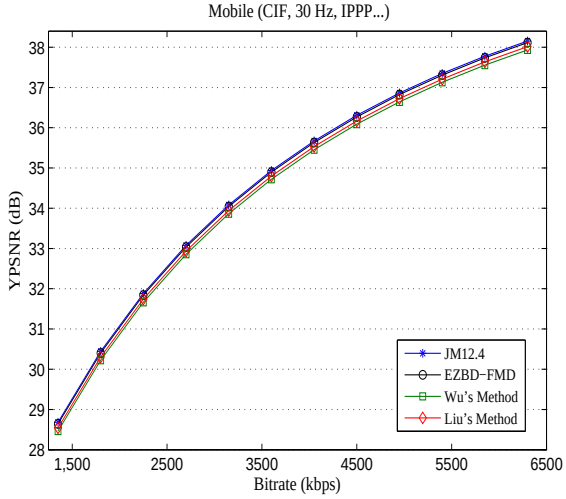
The performance of the EZBD-FMD algorithm is compared with the JM12.4 encoder for a wide range of QP. Figure 5.4 shows the RD curves of different sequences for all classes and resolutions. The RD curves show that the EZBD-FMD algorithm performs better than Wu's and Liu's algorithm for all classes of sequences. The performance of the proposed method is virtually identical to that provided by the JM12.4 software.



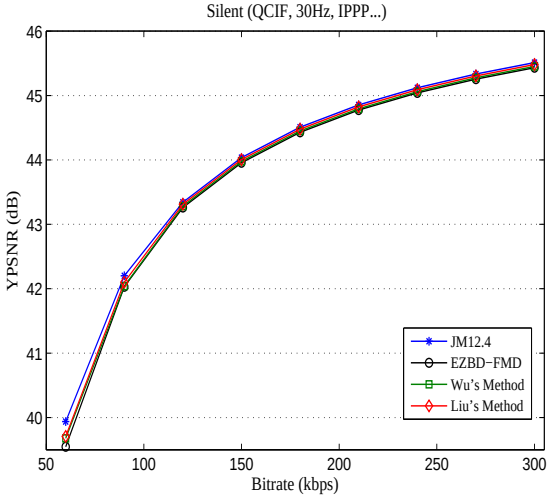
(a) RD Performance for Foreman (CIF) Sequence



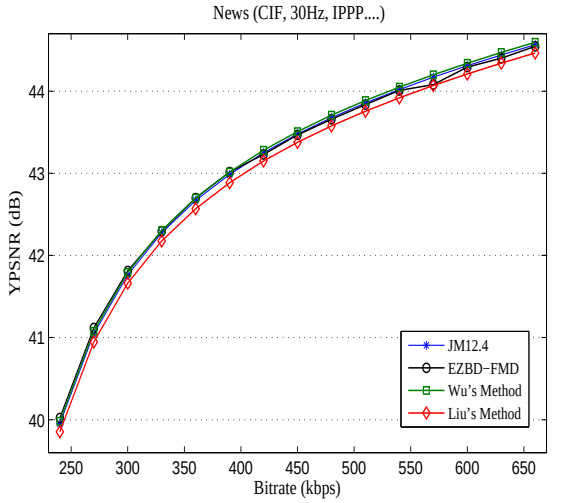
(b) RD Performance for Claire(QCIF) Sequence



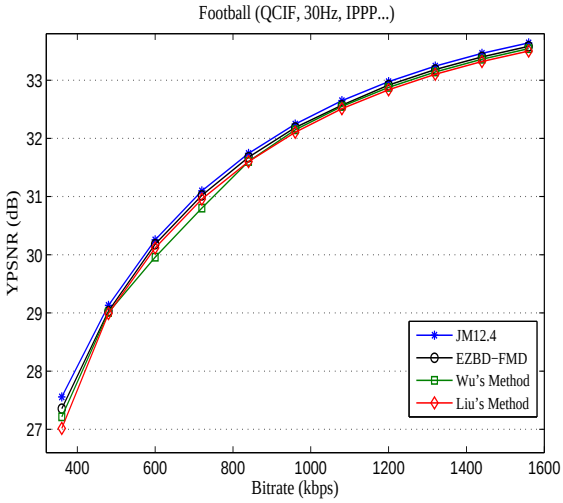
(c) RD Performance for Mobile(CIF) Sequence



(d) RD Performance for Silent(QCIF) Sequence



(e) RD Performance for News(CIF) Sequence



(f) RD Performance for Football(QCIF) Sequence

Figure 5.4: RD Performance for various sequence in the CIF and the QCIF resolution

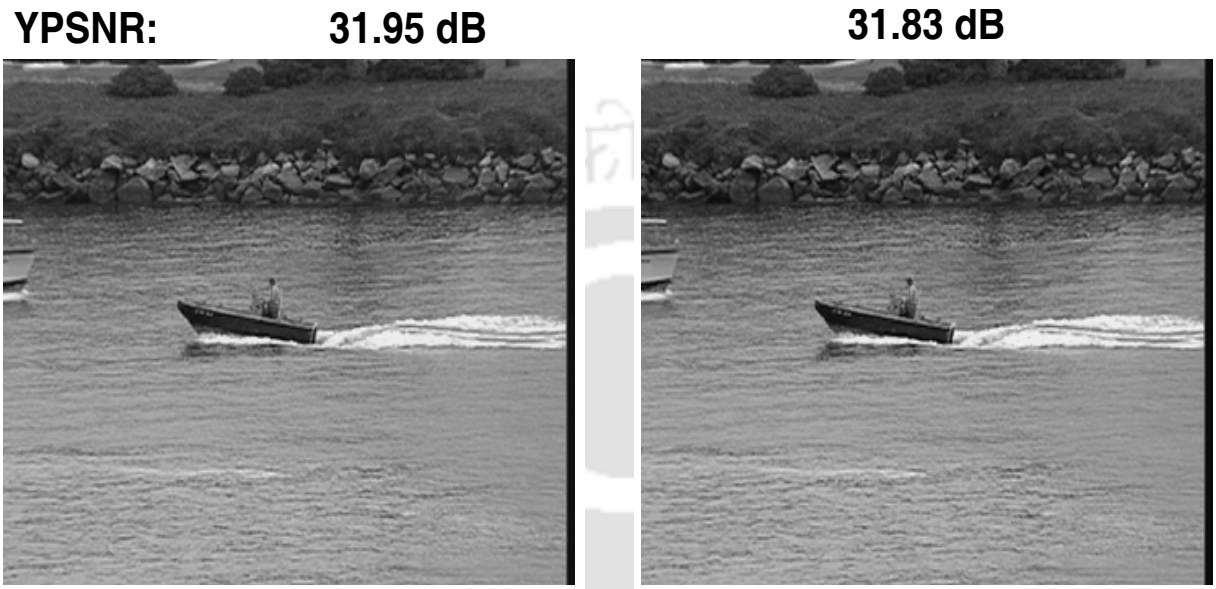
5.3.3 *Subjective Comparison*

For subjective comparison, the snapshots taken from sequences of different classes generated by the JM12.4 encoder and the EZBD-FMD algorithm are illustrated in Figure 5.8. No significant degradation in perceptual visual quality is observed.

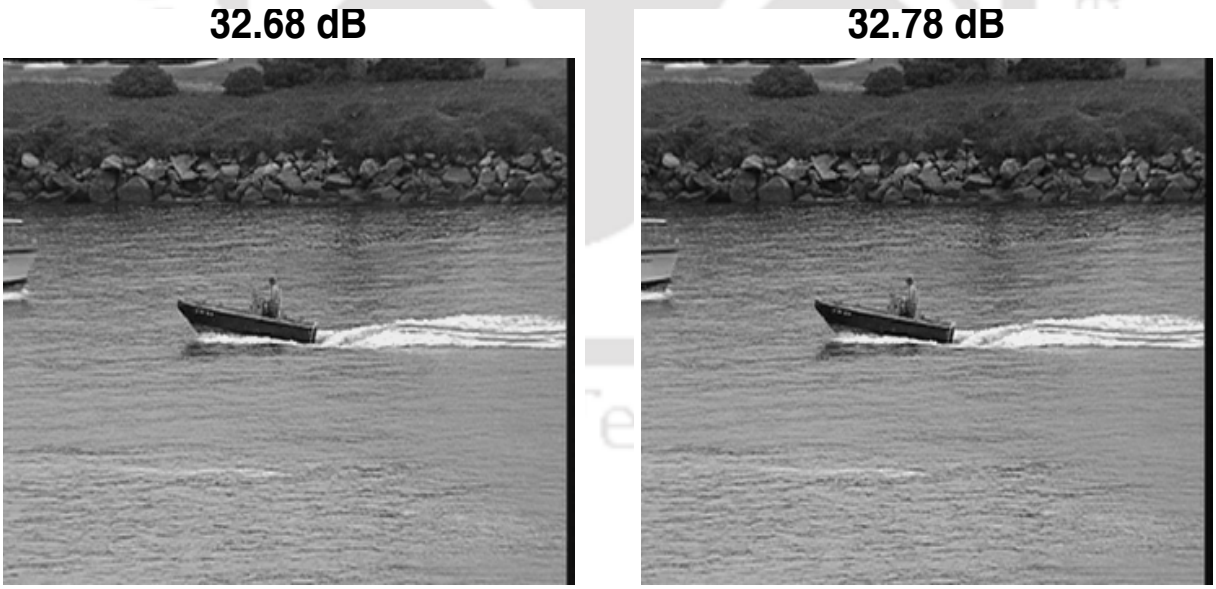
(a) Frame 2 and 3 of *News* encoded by JM12.4

(b) Frame 2 and 3 of *News* encoded by EZBD-FMD Algorithm

Figure 5.5: Snapshots of News:(a) Encoded by JM12.4 (b) Encoded by EZBD-FMD Algorithm

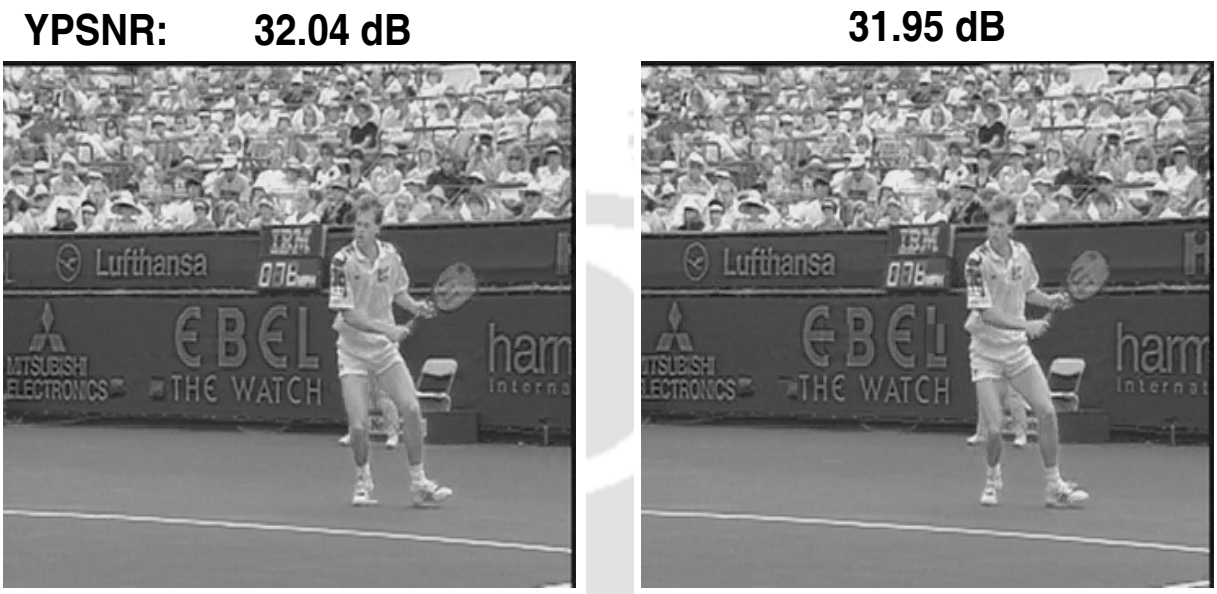


(a) Frame 2 and 3 of *Coastguard* encoded by JM12.4

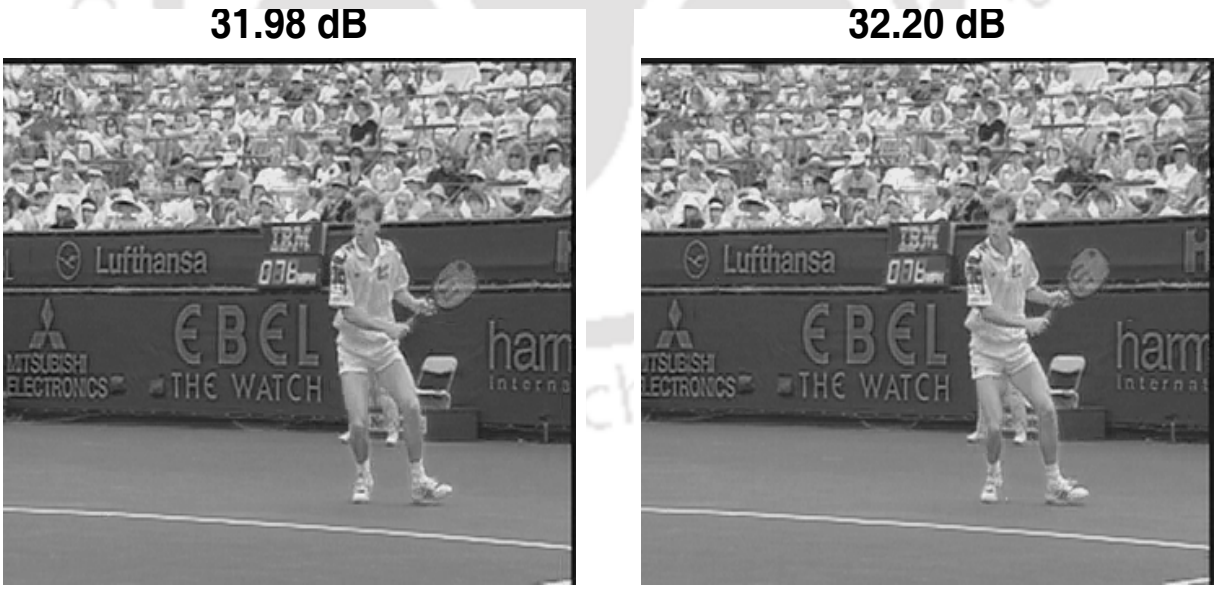


(b) Frame 2 and 3 of *Coastguard* encoded by EZBD-FMD Algorithm

Figure 5.6: Snapshots of Coastguard:(a) Encoded by JM12.4 (b) Encoded by EZBD-FMD Algorithm



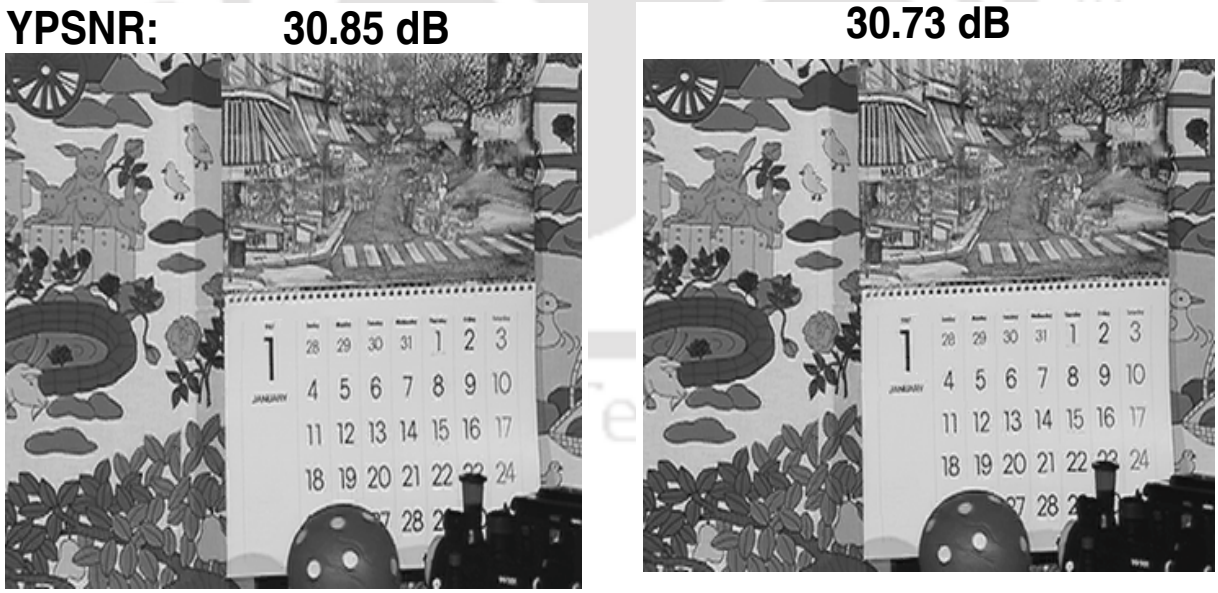
(a) Frame 2 and 3 of *Stefan* encoded by JM12.4



(b) Frame 2 and 3 of *Stefan* encoded by EZBD-FMD Algorithm

Figure 5.7: Snapshots of Stefan:(a) Encoded by JM12.4 (b) Encoded by EZBD-FMD Algorithm

(a) Frame 2 and 3 of *Mobile* encoded by JM12.4



(b) Frame 2 and 3 of *Mobile* encoded by EZBD-FMD Algorithm

Figure 5.8: Snapshots of Mobile:(a) Encoded by JM12.4 (b) Encoded by EZBD-FMD Algorithm

5.4 Mode Decision Using Weighted Prediction

All images have certain characteristics that are inherent to them. Natural videos have many homogeneous regions. In video sequences, there are stationary regions between frames and regions with moderate to complex motion. From an in-depth analysis of the full search algorithm in which properties of each MB are studied in details, a method is proposed where each MB is given a weight depending upon its characteristics. The motion content and the homogeneity parameters of each MB is determined prior to the ME process. The MBs correlation with neighboring MBs in respect of predicted MVs and encoding modes are studied. Weights are assigned for these parameters and the final mode is selected based upon these weights. We propose a Weight Based Fast Mode Decision (WBFMD) process where we define four weights for each MB based on its motion content, homogeneity, value of the predicted MV and the encoding modes of the neighboring MBs. Smaller weights are assigned when these parameters have low values indicating a simple MB whereas larger weights are assigned when these parameters have high value indicating a more complex MB.

5.4.1 Determination of Motion Content in a MB

Stationarity refers to the stillness between consecutive frames in the temporal direction. Regions having similar motion in consecutive frames are also considered temporally stationary. The MBs which are temporally stationary usually get encoded in the SKIP or in the P_{16x16} mode whereas MBs with large motion get encoded with smaller block sizes. Stationarity is determined from the residual block R_{DF} as described in the EZBD-FMD algorithm. From the analysis (Section 5.1.2) of the values in R_{DF}, we observe that at low values of QP, a MB encoded in SKIP mode has very low values of residuals in the corresponding R_{DF}. As QP increases, MBs in the SKIP mode have larger values of residuals in R_{DF}. Usually when MBs are encoded in SKIP mode, a large number of residuals in R_{DF} have values that are below 3 (Table 5.1). MBs having large motion have high valued residuals in R_{DF}. From Table 5.1, we note that the residuals (for QP values upto 24) of

MBs encoded in the SKIP mode have more than 90% residuals with absolute values that are usually below 3. In the WBFMD, this variation in the value of the residuals with the QP for each mode is taken into consideration while assigning the weights. For each MB a weight is assigned depending upon the QP and the values of the residuals in R_{DF}. MBs with low values of residuals are given lower weights indicating a region with little motion whereas MBs with large values of residuals are assigned higher weights indicating high to complex motion. Thus a threshold TH_{24} for the residuals is taken for QP values upto 24 and TH_{24} is taken equal to 2. For higher values of QP, the threshold is defined as

$$TH_{QP} = \begin{cases} TH_{24} & \text{if} \quad QP \leq 24 \\ \text{floor}[\frac{1}{4}(QP - 24) + TH_{24}] & \text{if} \quad QP > 24 \end{cases},$$

Hence with the increase in QP the threshold also increases. For each MB, let there be N residues in R_{DF} that are below the TH_{QP}. We define a residual ratio 'R' for each MB as the fraction of the residues 'N' in R_{DF} that are below the TH_{QP} to the total number of residues in R_{DF}.

$$R = \frac{\text{Number of Residues in } R_{DF} \text{ below } TH_{QP}}{\text{Total number of residues in } R_{DF}} = \frac{N}{256}.$$

A difference frame weight DF_{wt} is assigned to each MB based on the value of R and is given in Table 5.3 below.

Table 5.3: Difference Frame Weights DF_{wt}

R range	> 0.9	0.8-0.9	0.7-0.8	0.6-0.7	< 0.6
DF_{wt}	0	1	2	3	4

Larger values of DF_{wt} indicate a MB with higher motion content. Lower weights are assigned for MBs which have larger number of residuals below the threshold.

5.4.2 Determination of Homogeneity in a MB

Natural videos have many homogeneous regions. Homogeneous regions have similar spatial properties. These regions in most cases get encoded with larger block sizes. Re-

gions with more complex texture get encoded with smaller block sizes. If the homogeneity of a MB is detected early, then a decision on the possible encoding modes can be taken. There exist many techniques for detecting homogenous regions. The determination of the edge amplitudes in a video frame has been studied in Chapter 4. The sum of the edge amplitudes, $S_{Amp(N)}$ is calculated using (4.9). If $S_{Amp(N)}$ has a low value, it suggests that the block is homogenous.

The the number of MBs in different modes was plotted against the value of $S_{Amp(N)}$ (N=16) and was shown in Figure 4.10 for the Mobile sequence. It clearly shows that for the SKIP mode the value of $S_{Amp(16)}$ is below 5000. For large block sizes, $S_{Amp(16)}$ is generally below 30000. For smaller block sizes, this value is significantly large. In WBFMD algorithm, weights are assigned based on the values of $S_{Amp(16)}$ which are obtained from the study of the histogram characteristics for different sequences. Depending upon these statistics the weights assigned are given in Table 5.4. Lower weight is given for the MB if $S_{Amp(16)}$ is low and higher weight for larger values of $S_{Amp(16)}$.

Table 5.4: Homogeneous MB Weights Hom_{wt}

$S_{Amp(16)}$ Range	< 5000	5001-15000	15001-20000	20001-25000	> 25000
Hom_{wt}	0	1	2	3	4

5.4.3 Determination of Predicted MV

In H.264/AVC the predicted MVs (pmv) for a MB are obtained from the MVs of the neighboring MBs [12]. Motion vectors for neighboring partitions are often highly correlated. Thus the calculated pmvs give a good indication of the degree of possible motion in the MB. Higher pmvs indicate the possibility of higher motion for the MB and vice versa. In WBFMD algorithm, a pmv weight (pmv_{wt}) is introduced where weights are assigned for an MB depending upon the pmv for the MB. For low values of pmvs smaller weights are assigned and for higher values larger weights are assigned. This is given in Table 5.5 where pmv_x and pmv_y respectively are predicted MVs in in the horizontal and

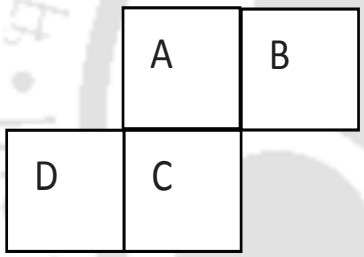
vertical directions.

Table 5.5: Predicted MV Weights pmv_{wt}

max $\|pmv_x\|$ or max $\|pmv_y\|$ or both	0	1-2	3-4	5	> 5
pmv_{wt}	0	1	2	3	4

5.4.4 Determination of Predicted mode from Neighboring Modes

It is observed that the encoding modes of the neighboring MBs are often correlated.



(a) Neighboring MBs

Mode	Value Assigned
SKIP	0
P16x16	1
P16x8	2
P8x16	3
P8x8	4
P8x4	5
P4x8	6
P4x4	7
INTRA	8

(b) Values assigned

Figure 5.9: Neighboring MBs: C is the current MB and Values Assigned for different Modes

Referring to Figure 5.9(a), let C be the current MB and A, B and D be the neighboring MBs that have already been encoded. Let the encoding modes of A, B and D be A_{MODE}, B_{MODE} and D_{MODE} which take values assigned corresponding to the modes as given in Table in Figure 5.9(b). The following relation is used to determine the likely mode for C from the modes of A,B anc D:

$$Neigh_{MODE} = \text{median}\{A_{MODE}, B_{MODE}, D_{MODE}\}$$

Depending upon the value of $Neigh_{MODE}$ for a MB, a neighboring mode weight (NM_{wt}) is defined. If $Neigh_{MODE}$ for the MB indicate large block size partition then smaller weight is assigned to NM_{wt} and higher weight is assigned if block partition is small. The NM_{wt} for the MB is as given in the Table 5.6.

Table 5.6: Neighboring Mode Weights NM_{wt}

$Neigh_{MODE}$	SKIP	P_{16x16}	P_{16x8} P_{8x16}	P_{8x8} P_{8x4},P_{4x8}	P_{4x4} INTRA
NM_{wt}	0	1	2	3	4

5.4.5 Overall WBFMD Algorithm

In WBFMD algorithm, for each MB, the four different weights are first determined and a total weight $Total_{wt}$ is defined for each MB as

$$Total_{wt} = [\ DF_{wt} \quad Hom_{wt} \quad pmv_{wt} \quad NM_{wt}\ ]$$

- If at least three weights in $Total_{wt}$ are same and is equal to say x (where x=0,1,2,3,4) then $Final_{wt}$=x
- If any two weights in $Total_{wt}$ are equal, then $Final_{wt}$=ceil (median ($Total_{wt}$))
- If all the weights in $Total_{wt}$ are unequal, then $Final_{wt}$=max($Total_{wt}$)

Based on this value of $Final_{wt}$, the decision on the encoding mode is taken. A low value of $Final_{wt}$ for a MB suggest that the MB is homogeneous with little motion and will be encoded with larger block sizes. A high value of $Final_{wt}$ indicate higher motion and complexity and will be encoded using smaller block sizes. The mode selection for each MB based on the value of the $Final_{wt}$ is given in Table 5.7.

Table 5.7: Final Encoding Modes for MBs

$Final_{wt}$	0	1	2	3	4
Mode	SKIP	P_{16x16}	P_{16x8} P_{8x16}	P_{8x8} P_{8x4},P_{4x8}	P_{4x4} INTRA

5.5 Experimental Observations

Simulations carried out with the WBFMD algorithm has shown considerable time savings while maintaining the PSNR and the bitrate quality. The results are discussed in detail in this section.

5.5.1 *Distortion and Compression Ratio Comparisons*

Table 5.8 lists the performance of the WBFMD algorithm in comparison to JM12.4 implementation and the algorithms due to Wu [87] and Liu [72]. The results show that for all the sequences, there is only a marginal loss in the PSNR performance. There is an average loss of 0.04 dB in PSNR while the average bitrate increase is 0.96%. The results demonstrate the effectiveness of the proposed algorithm.

Table 5.8: Performance Comparison For Different Sequences

Class	Sequence	Performance Comparison								
		WBFMD Algorithm			Wu et al.'s [87]			Liu et al.'s [72]		
		ΔPSNR dB	ΔRate (%)	ΔT (%)	ΔPSNR dB	ΔRate (%)	ΔT (%)	ΔPSNR dB	ΔRate (%)	ΔT (%)
CIF Class A	News	-0.05	-0.90	88.42	0.02	0.84	39.23	0.11	2.41	48.53
	MaD	0.02	0.86	81.38	0.01	0.62	43.21	0.09	1.76	55.25
	Container	0.01	0.74	60.33	0.05	1.21	46.18	0.08	1.61	52.59
	Hall	0.09	1.42	76.84	0.06	1.27	34.67	0.07	1.35	46.27
Class B	Foreman	0.08	1.32	72.79	0.07	1.32	34.90	0.05	1.04	39.86
	Coastguard	0.05	1.12	65.07	0.05	1.18	26.30	0.02	0.71	33.56
	Ice	0.05	1.14	85.17	0.07	1.29	45.68	0.05	1.09	54.12
	Harbour	0.06	1.28	65.37	0.05	1.15	21.56	0.06	1.12	23.26
Class C	Flower	0.08	1.35	76.43	0.05	1.17	36.85	0.06	1.16	42.86
	Stefan	0.08	1.29	70.59	0.02	0.84	32.25	0.08	1.59	39.54
	Tempete	0.02	0.82	71.88	0.09	1.48	27.21	0.09	1.63	35.42
	Mobile	0.04	1.08	62.41	0.09	1.54	12.35	0.07	1.45	19.28
	Average	**0.04**	**0.96**	**73.05**	**0.05**	**1.15**	**33.36**	**0.06**	**1.41**	**40.87**
QCIF Class A	Claire	-0.01	-0.81	90.69	-0.01	-0.95	47.35	0.01	0.51	62.45
	MissAmerica	-0.07	-1.40	89.87	0.02	0.81	48.23	0.03	0.89	60.27
	Suzie	0.04	0.95	77.73	0.05	1.24	42.91	0.06	1.32	59.65
Class B	Foreman	0.08	1.34	67.73	0.02	0.85	30.25	0.05	1.21	41.18
	Silent	0.03	0.83	82.46	0.06	1.41	42.62	0.03	0.91	55.24
	Crew	0.08	1.41	63.86	0.05	1.20	19.64	0.04	1.12	30.43
Class C	Football	0.10	1.80	65.30	0.05	1.24	28.84	0.05	1.26	31.75
	Mobile	0.14	1.92	28.09	0.07	1.39	15.32	0.08	1.71	21.29
	Soccer	0.04	1.17	63.67	0.03	0.95	20.19	0.04	1.10	27.65
	Average	**0.04**	**0.80**	**77.18**	**0.03**	**0.90**	**33.21**	**0.04**	**1.11**	**43.32**

ΔPSNR(+/-): picture quality loss/gain measured in dB
ΔRate(+/-): bitrate increase/decrease measured as a %
ΔT(+/-): encoding time saving/loss measured as a %

5.5.2 *Computational Speedup*

Table 5.8 also shows the percentage reduction in encoding time $\Delta T(\%)$ for sequences of different classes. The time saving obtained depends upon the type of sequence. The average savings in the encoding time is 73.05% for CIF sequences. An increased saving is noted for Class A sequences whereas time saving obtained for Class C sequences is comparatively low. The saving in time is achieved as the decision on the final mode for encoding is taken prior to the ME and as seen from Table 5.7 for each MB, at the most, only three modes are searched.

5.5.3 *Comparison with JM12.4 Modes*

Table 5.9 shows the percentage of MBs that are encoded using the WBFMD algorithm in the same mode as the JM12.4 for different QP. The results show that the WBFMD algorithm is effective as it has been able to maintain the same final encoding modes as the JM12.4 to a large extent. The results are not similar for the Mobile and the Stefan sequences at lower QP values.

Table 5.9: % of MBs encoded in the same mode w.r.t JM12.4

Sequence (CIF)	Quantization Parameter			
	24	28	32	36
News	82.10	75.12	84.12	86.01
MaD	66.12	87.86	89.6	90.66
Container	76.50	86.01	89.30	89.15
Hall	59.11	68.69	72.45	83.33
Foreman	46.11	58.71	64.90	76.23
Coastguard	62.33	67.83	78.84	79.47
Ice	67.42	73.99	71.97	72.98
Harbour	46.21	52.66	55.08	63.33
Flower	69.70	70.71	69.44	72.14
Stefan	22.73	57.32	66.31	68.99
Tempete	52.27	59.75	68.23	70.43
Mobile	29.80	34.34	49.39	66.19

5.6 Conclusions

In video coding, it is quite common that all the DCT coefficients in a block are quantized to zero. For this condition only a special symbol indicating this to be an all-

zero block is sent to the decoder. In many cases blocks which have all zero coefficients are encoded in the SKIP mode. In this chapter an attempt has been made to detect the SKIP mode based on the frame difference residues and early detection of the all-zero blocks thereby reducing the computations and encoding time.

The first part of the chapter presented an Early Zero Block Detection based Fast Mode Decision algorithm (EZBD-FMD Algorithm) for fast inter mode decision where the decision was taken at two levels. The first level identified the SKIP modes for the MBs by inspecting the frame difference residues. To achieve further reduction in complexity, the next level made use of the number of ZBs in a MB to arrive at a mode decision. The experimental results showed that EZBD-FMD method gives considerable saving in time-complexity.

The second part of the chapter presented a Weight Based Fast Mode Decision (WBFMD) method based on the analysis of the full ME and weights assigned to the different characteristics of the image. This algorithm described fast mode decision process where each MB in a frame is tested for its motion content and homogeneity, the predicted motion vectors for the MB and the modes of the neighboring MBs. Each MB will have different degrees of these characteristics. The WBFMD algorithm presents a method where each MB is first analyzed for its characteristics without going through the complex ME calculation. Based on the nature of the MB, the encoding mode is chosen. This method show good rate distortion performance and reduction in encoding time.

During the detailed study of the transform coefficients for the early SKIP mode detection, it was observed that the transform coefficients carry some information regarding the encoding modes. These coefficients are available to us since for the RDO process, availability of the bitrate requires that the transformation and quantization be performed for each mode. The next chapter describes new algorithms where the information available directly from the transform coefficients are used for speeding up the encoding process.

6

Fast Mode Decision in the Transform Domain

Contents

6.1 Transform Domain based Fast Mode Decision (TDFMD) . . 113
6.2 Fast Intra Prediction of TDFMD 116
6.3 Fast Inter Prediction of TDFMD 118
6.4 Experimental Observations 120
6.5 Selective Coefficient Fast Mode Decision (SCFMD) 123
6.6 Proposed SCFMD Algorithm 125
6.7 Experimental Observations 126
6.8 Conclusions 128

Rate Distortion Optimization is utilized mainly for the selection of the prediction mode, the motion vectors in ME and the block size. The RDO cost computation and mode selection process is shown in Figure 6.1. For both intra and inter mode decision process in H.264/AVC, the bits required to encode the residuals and the header information are needed for the calculation of the RD cost for each mode. This requires that the transformation and quantization be performed for each mode. Thus the transform coefficients (TC) for each mode are available in the encoder even before the final mode is decided for encoding.

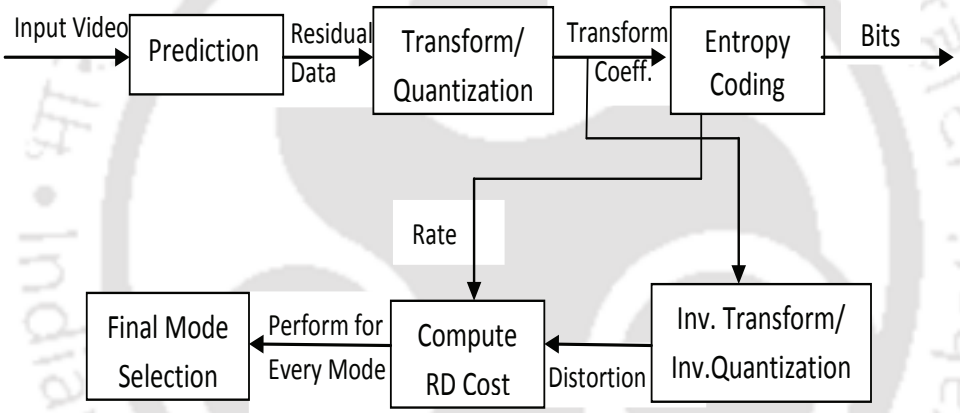


Figure 6.1: RDO Cost Computation Process

The distribution of the transform coefficients for a MB are different for different encoding modes. The coefficients have both zero and non-zero values. For low bit rate encoding (high QP) more transform coefficients are zero-valued. At lower values of QP, there are more non zero transform coefficients. Chapter 5 describes a condition where a ZB is identified prior to performing actual transformation and quantization. Depending upon the number of ZBs in a MB, the decision on the final mode was taken. However, in this chapter, we first perform the transformation and quantization process and take the decision based on the values of the transform coefficients so obtained.

This chapter proposes two algorithms in the transform domain : Transform Domain based Fast Mode Decision (TDFMD) and Selective Coefficient Fast Mode Decision

(SCFMD). In a recent work [102] a Sum of Absolute Transformed Difference (SATD) based encoding algorithm for intra prediction has been proposed. The authors have proposed a rate estimation process based on the variance of the SATD. The SATD and the variance information is incorporated in the RDO process to speed up the encoding process. The method achieves a significant improvement in computation time. But the complexity of the algorithm is high. The methods proposed here are simple to implement and give a good rate distortion performance.

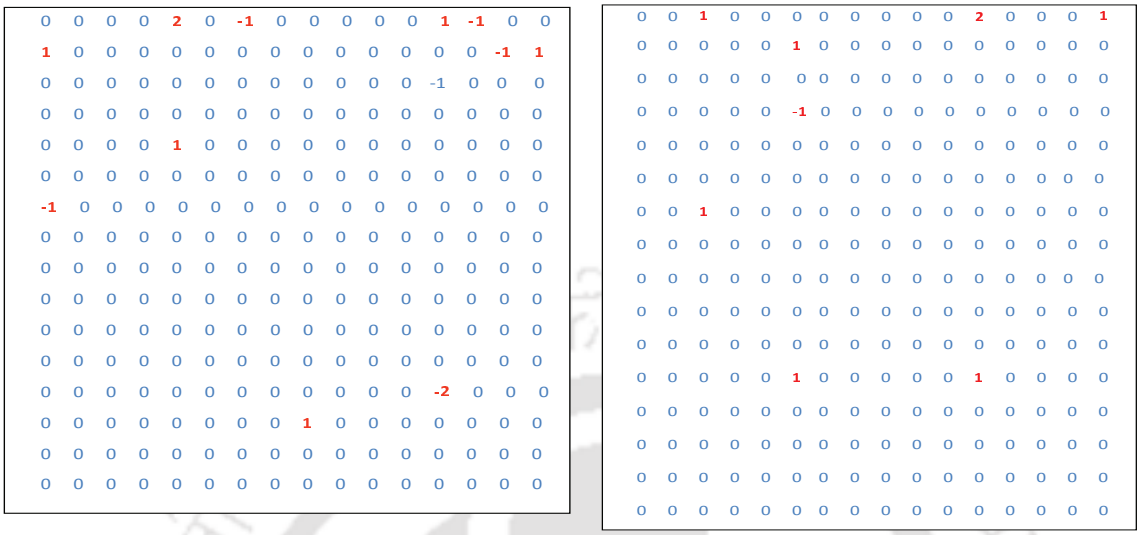
6.1 Transform Domain based Fast Mode Decision (TDFMD)

Transform coefficients are available at the encoder after performing transformation and quantization of the residuals of a block. The process of transformation and the quantization has been explained in detail in Section 2.7.1 of Chapter 2. These steps are performed on the residuals obtained from prediction process for every mode as is shown in Figure 6.1. We perform a detailed study of the transform coefficients obtained from intra and inter prediction processes.

6.1.1 Observations from Transform Coefficients in Intra Prediction Process

The intra prediction process for a MB includes the I16MB and the I4MB modes. The computational complexity involved in the I4MB prediction process is higher than that in I16MB prediction process. First, the transform coefficients of a MB are obtained from the exhaustive intra prediction process where it may be encoded either in the I4MB or the I16MB mode. Next, for the MB, the I16MB prediction is performed and the transform coefficients only for the I16MB prediction are obtained. The following observations are made based on the transformed coefficients obtained from I16MB and the exhaustive intra prediction process.

- The transform coefficients of the MBs from the I16MB prediction process are stud-



(a) Transform Coefficient for I16MB prediction (b) Transform Coefficient for I4MB prediction

Figure 6.2: Transform Coefficients for I16MB and I4MB prediction for a MB in Foreman sequence (SAD difference between I16MB and I4MB mode is 20)

ied. It is found in almost all the cases that when more than 95% of these transform coefficients are zero, the MB is finally encoded in the I16MB from the exhaustive intra prediction process. Thus, it may be concluded that if more than 95% of the transform coefficients in a MB from I16MB prediction are zero, in most cases the final mode is the I16MB mode.

- The SAD for a MB encoded in I4MB mode (SAD4) from exhaustive prediction and SAD for the same MB from I16MB prediction (SAD16) are obtained using (2.14). The SAD4 value is compared with SAD16 value. The MBs for which the difference between SAD4 and SAD16 is small and is generally below 30, the transform coefficients with I16MB prediction and I4MB prediction both have only a few non zero coefficients. Figure 6.2 shows two such transform coefficients with I16MB and I4MB prediction with the final mode being I4MB. The difference in SAD for the two modes here is 20. It is seen that the transform coefficients for both the prediction process are very similar.

- When difference between SAD4 and SAD16 for a MB is small with the final mode

being I4MB, it is seen that if the MB is encoded in the I16MB then there is negligible degradation in quality.

- When the SAD difference is small and the transformed block from I16MB (Figure 6.2) is divided into 16 subblocks, it is observed that many subblocks are ZBs. The subblocks which are not ZBs have only one or two non zero coefficients.
- When the final mode is the I4MB mode and the SAD difference is large then for that MB the transform coefficients obtained from I16MB prediction have many non zero coefficients.

From the above discussions it is clear that the transform coefficients obtained after the I16MB prediction gives an indication of the final mode that may be selected for encoding.

6.1.2 Observations from Transform Coefficients in Inter Prediction Process

For each MB, motion compensated residual and transform coefficients are obtained from full search inter prediction. The transform coefficients obtained after exhaustive inter prediction and those obtained for the P_{16x16} ME are compared and the following observations are made:

- When all the transform coefficients from the P_{16x16} ME are zero, the MB is encoded in the SKIP or P_{16x16} mode.
- When the final mode for a MB from exhaustive ME is the P_{16x16} mode, it is found that for such MBs around 95% of the transform coefficients are equal to zero. In other words, if more than 95% of the transform coefficients from P_{16x16} ME are zero, then it may be concluded that the MB will eventually get encoded in the P_{16x16} mode thus terminating further ME process.
- When the transform coefficients obtained after performing P_{16x16} ME on the MB have many non-zero coefficients (more than 5%), it is noted that for such MBs, the final mode after exhaustive ME have smaller block sizes.

- As the QP increases, the number of non zero coefficients decreases. At higher QPs, there are only a few non zero coefficients and thus more MBs are encoded in the P_{16x16} mode.

From the above analysis of the transform coefficients for both the intra and inter prediction processes, we propose the TDFMD algorithm.

6.2 Fast Intra Prediction of TDFMD

Based on the observations made in the previous section a fast intra prediction algorithm is proposed. The steps for the proposed algorithm are given below:

(i) Perform exhaustive I16MB prediction for each MB and obtain the residual block R by subtracting the predicted pixels from the original pixels.

(ii) Perform the transformation and quantization on R and obtain the transform coefficients block R_T. Find the number of zero valued coefficients ZC in R_T.

(iii) If the number of zero valued coefficients are greater than 95% of the total transform coefficients in the MB (ZC >245), final mode is I16MB. Else, check the MB further for ZBs.

(iv) The transformed block R_T is divided into sixteen 4×4 sub-blocks. The transform coefficients for each subblock is studied. The number of subblocks (N_Z) that have at the most only one non-zero coefficient is determined. If N_Z is more than 90% for a MB i.e $N_Z \geq 14$, then the MB is encoded in the I16MB. There is no additional computation required for this MB as the predicted MB is already obtained in step 1. Else,

(v) Perform I4MB prediction for that MB and the final decision is taken on the minimum RDO cost.

(vi) Choose the final mode that minimize the RDcost of the I4MB and the I16MB prediction.

(vii) Proceed with the next MB.

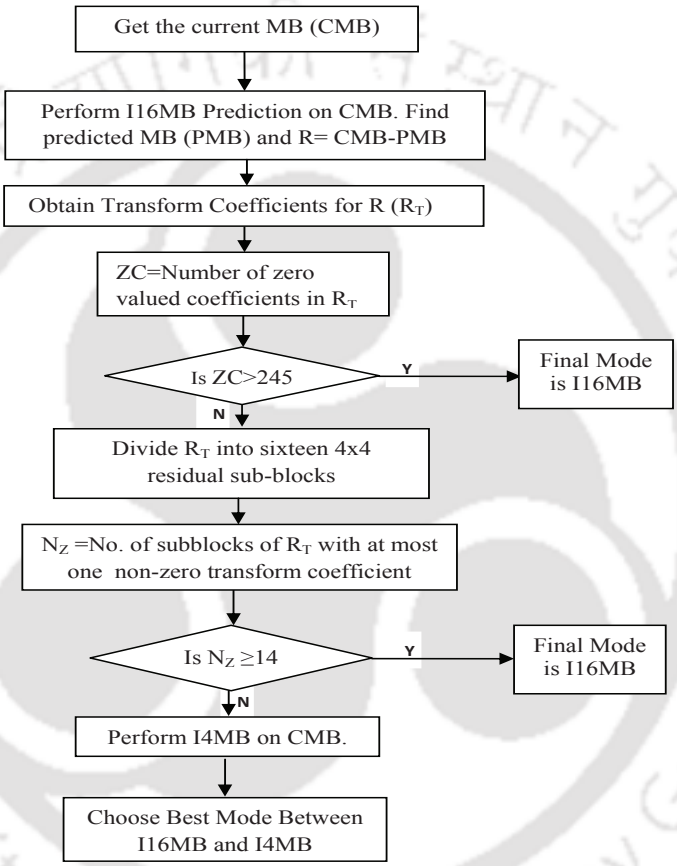


Figure 6.3: Flowchart for the Fast Intra Prediction of TDFMD

As the QP increases, number of zero valued transform coefficients increases and the number of MBs encoded in the I16MB increases leading to substantial savings in the complexity. The flowchart for the fast intra prediction is given in Figure 6.3.

6.2.1 Experimental Observations with All Intra Frames

Results of simulations carried out on different sequences with all intra frames are given in Table 6.1. The average savings in time is 62.73% and 67.75% for CIF and QCIF sequences respectively with corresponding PSNR loss of 0.04 dB and 0.03 dB and bitrate

increase of 1.19% and 1.25%. The results indicate that with the TDFMD algorithm for intra prediction, there is a marginal quality degradation and negligible increase in the bitrate compared to JM12.4.

Table 6.1: Results for All Intra Frame Sequences

Format	Sequence	TDFMD			Pan's et al.'s [22]			Liu et al.'s [34]		
		ΔPSNR (dB)	ΔRate (%)	ΔT (%)	ΔPSNR (dB)	ΔRate (%)	ΔT (%)	ΔPSNR (dB)	ΔRate (%)	ΔT (%)
QCIF	Foreman	0.06	1.23	64.40	0.24	3.24	65.40	0.20	2.94	66.20
	City	0.06	1.30	63.25	0.35	2.12	55.23	0.27	1.89	58.78
	Mobile	0.02	1.01	65.10	0.25	2.94	61.34	0.26	2.98	63.43
	Bus	0.01	0.85	62.15	0.18	2.45	56.34	0.19	2.49	57.52
	Football	0.04	1.15	69.34	0.24	2.67	53.45	0.26	2.69	58.35
	Soccer	0.03	0.99	55.17	0.20	2.78	49.36	0.19	2.23	52.45
	Crew	0.10	1.83	59.76	0.32	1.67	45.67	0.29	1.47	50.34
	Average	**0.04**	**1.19**	**62.73**	**0.25**	**2.55**	**55.25**	**0.23**	**2.38**	**58.15**
CIF	Foreman	0.06	1.87	69.87	0.25	3.26	59.13	0.21	2.86	64.15
	City	0.07	1.98	62.76	0.35	1.80	46.54	0.29	1.62	52.43
	Mobile	0.01	0.85	66.89	0.26	2.96	55.35	0.26	2.96	60.35
	Bus	0.02	1.09	63.51	0.19	2.10	56.41	0.15	1.89	58.52
	News	0.03	1.16	69.42	0.28	3.01	48.56	0.26	2.76	54.19
	Silent	0.01	0.84	73.15	0.18	1.98	54.78	0.15	1.48	63.28
	Tempete	0.02	1.02	66.67	0.20	2.27	50.43	0.22	2.43	55.28
	Mother-daughter	0.04	1.21	75.01	0.25	2.16	49.76	0.24	2.10	58.53
	Crew	0.05	1.26	62.47	0.29	1.35	55.12	0.30	1.39	60.30
	Average	**0.03**	**1.25**	**67.75**	**0.25**	**2.32**	**52.90**	**0.23**	**2.16**	**58.55**

ΔPSNR(+/-): picture quality loss/gain measured in dB
ΔRate(+/-): bitrate increase/decrease measured as a %
ΔT(+/-): encoding time saving/loss measured as a %

6.3 Fast Inter Prediction of TDFMD

The fast inter prediction of TDFMD is divided into three levels. The first level identifies MBs to be encoded in the SKIP mode prior to performing ME. The second level consists of identifying the MBs qualifying for the P_{16x16} mode. The third level is the directional mode decision (DMD). Each of the levels are discussed below.

Level I: SKIP mode detection

Level I identifies the MBs for the SKIP mode. Early detection of SKIP mode prior to performing ME saves computations. An approximate motion compensated residual, $\hat{E}_{MC}$, is obtained the using predicted MVs (pmv_x and pmv_y) by following the procedure

in Section 5.2.2 of Chapter 5. Transformation and quantization are performed on the residuals of $\hat{E}_{MC}$. If all the transform coefficients are quantized to zero (condition (d) of Section 5.1.2 for SKIP mode), then the MB is encoded in the SKIP mode. Otherwise, further ME are performed on the MB.

Level II: P_{16x16} mode detection

The second level applies for those MBs which do not qualify for the SKIP mode in the first level. For such MBs, P_{16x16} ME is performed and the motion compensated residual $\hat{E}_{MC_{16}}$ and RD cost J_{16x16} is obtained. Transformation and quantization is performed on $\hat{E}_{MC_{16}}$. If 95% or more of the transform coefficients are zero, the MB is encoded in the P_{16x16} mode and the ME is terminated. Otherwise, directional mode detection is performed for the MB.

Level III: Directional Mode Detection

The third level is performed on the MBs that fail to qualify for the SKIP and the P_{16x16} modes in the previous levels. The motion calculation is directed along the direction which gives lower J_{MODE}. In the proposed TDFMD algorithm, ME is performed with the P_{16x8} and P_{8x16} modes and corresponding costs J_{16x8} and J_{8x16} are obtained. If J_{16x16} obtained from the previous level is lower than both J_{16x8} and J_{8x16}, the final mode is P_{16x16}. If J_{16x8} is lower than J_{8x16}, then further ME is performed in the vertical partitions (P_{8x8}, P_{8x4}, P_{4x4}). The final mode is the one which minimizes the RD cost amongst these modes. If J_{8x16} is lower than J_{16x8}, then further ME is performed in the horizontal partitions (P_{8x8}, P_{4x8}, P_{4x4}). The final mode is the one with the minimum RD cost among these modes. The flow chart for the directional mode detection is given in Figure 6.4.

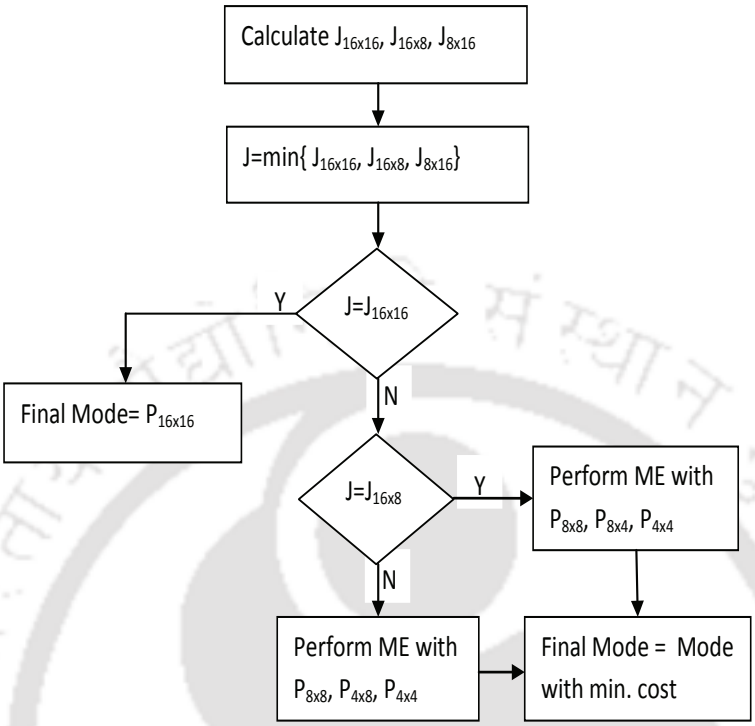


Figure 6.4: Flowchart for the DMD of TDFMD

6.4 Experimental Observations

In the implementation of the TDFMD algorithm, both the fast intra and inter prediction process is incorporated. The results of the TDFMD algorithm is compared with the exhaustive mode decision of the H.264/AVC benchmark JM12.4 and the algorithms due to Wu [87] and Liu [72].

6.4.1 *Distortion and Compression Ratio Comparisons*

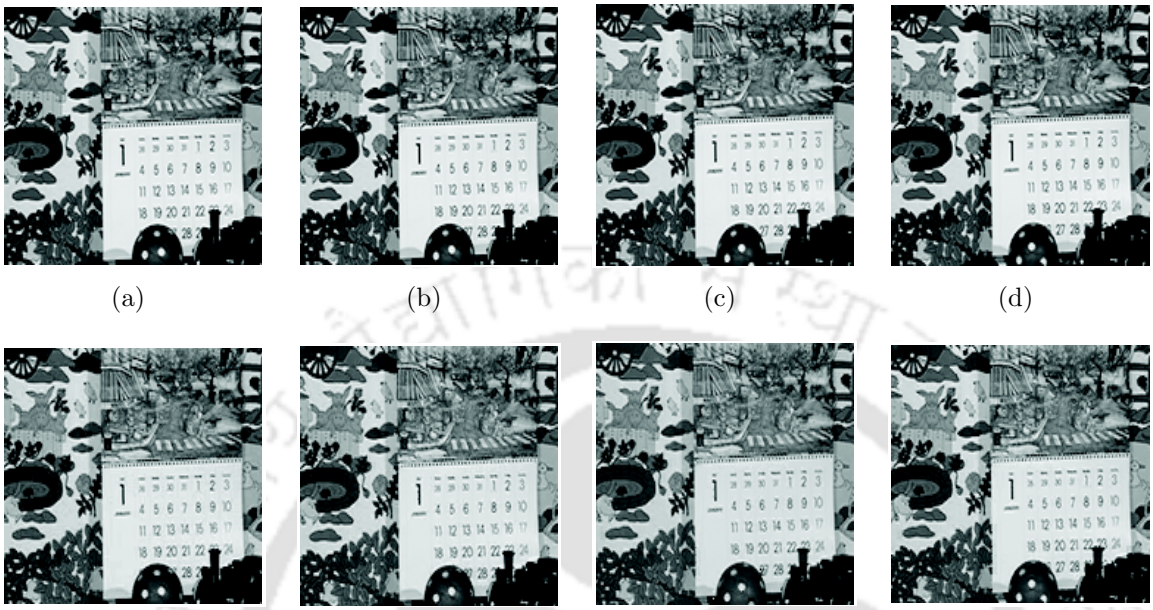
Table 6.2 shows the TDFMD algorithm gives an average encoding time savings of 77%(CIF) and 74% (QCIF) with PSNR loss of 0.04 dB for both. The average bitrate increase is 1.12% and 0.94% respectively. For the intra prediction the decision on whether the I4MB prediction will be performed or not will depend upon the total number of zero valued coefficients obtained after I16MB prediction. Since at higher QPs, there are more zero valued coefficients, hence I4MB prediction will have to be performed on fewer MBs

Table 6.2: Performance Comparison For Different Sequences

Class	Sequence	Performance Comparison								
		TDFMD			Wu et al.'s [87]			Liu et al.'s [72]		
		ΔPSNR dB	ΔRate (%)	ΔT (%)	ΔPSNR dB	ΔRate (%)	ΔT (%)	ΔPSNR dB	ΔRate (%)	ΔT (%)
CIF Class A	News	0.07	1.38	86.32	0.02	0.84	39.23	0.11	2.41	48.53
	MaD	0.05	1.22	86.74	0.01	0.62	43.21	0.09	1.76	55.25
	Container	0.02	1.06	87.96	0.05	1.21	46.18	0.08	1.61	52.59
	Hall	0.07	1.34	84.17	0.06	1.27	34.67	0.07	1.35	46.27
Class B	Foreman	0.07	1.38	80.54	0.07	1.32	34.90	0.05	1.04	39.86
	Coastguard	0.01	0.89	73.99	0.05	1.18	26.30	0.02	0.71	33.56
	Ice	0.10	0.86	81.76	0.07	1.29	45.68	0.05	1.09	54.12
	Harbour	0.04	1.21	65.58	0.05	1.15	21.56	0.06	1.12	23.26
Class C	Flower	0.04	1.16	77.35	0.05	1.17	36.85	0.06	1.16	42.86
	Stefan	0.04	1.11	70.58	0.02	0.84	32.25	0.08	1.59	39.54
	Tempete	0.02	1.02	71.47	0.09	1.48	27.21	0.09	1.63	35.42
	Mobile	0.01	0.90	67.63	0.09	1.54	12.35	0.07	1.45	19.28
	Average	**0.04**	**1.12**	**77.84**	**0.05**	**1.15**	**33.36**	**0.06**	**1.41**	**40.87**
QCIF Class A	Claire	-0.01	-1.01	90.91	-0.01	-0.95	47.35	0.01	0.51	62.45
	MissAmerica	0.01	0.76	91.00	0.02	0.81	48.23	0.03	0.89	60.27
	Suzie	0.05	1.24	86.56	0.05	1.24	42.91	0.06	1.32	59.65
Class B	Foreman	0.05	1.20	80.53	0.02	0.85	30.25	0.05	1.21	41.18
	Silent	0.09	1.45	86.67	0.06	1.41	42.62	0.03	0.91	55.24
	Crew	0.09	1.42	64.84	0.05	1.20	19.64	0.04	1.12	30.43
Class C	Football	0.06	1.31	46.72	0.05	1.24	28.84	0.05	1.26	31.75
	Mobile	0.01	0.82	63.57	0.07	1.39	15.32	0.08	1.71	21.29
	Soccer	0.06	1.29	60.34	0.03	0.95	20.19	0.04	1.10	27.65
	Average	**0.04**	**0.94**	**74.57**	**0.03**	**0.90**	**33.21**	**0.04**	**1.11**	**43.32**

ΔPSNR(+/-): picture quality loss/gain measured in dB
ΔT(+/-): encoding time saving/loss measured as a %
ΔRate(+/-): bitrate increase/decrease measured as a %

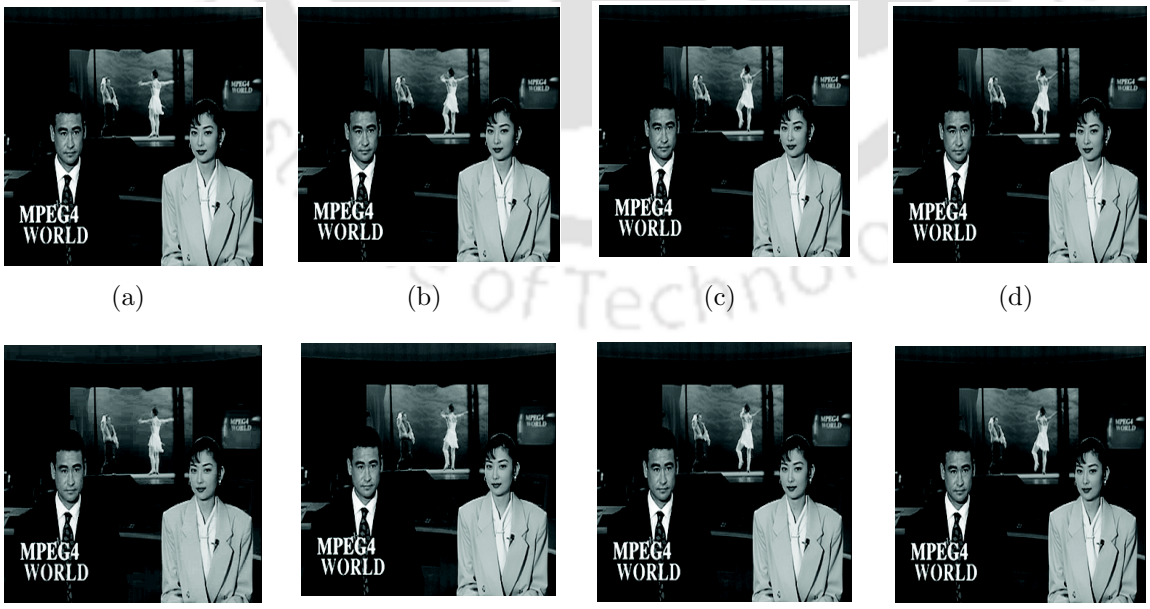
leading to a large encoding time saving. The high degree of encoding time saving may be explained as follows. For the inter prediction, as the QP increases, more transform coefficients tend to be zero valued. Hence the MBs will have larger number of zero coefficients leading to more MBs being encoded in the SKIP mode or P_{16x16} mode thus reducing the time complexity. Figures 6.5 and Figure 6.6 show the snapshots of the first four frames of Mobile (QCIF) and News (CIF) encoded with JM12.4 and TDFMD. There are no visual degradations observed with TDFMD algorithm.



(a) (b) (c) (d)

(e) (f) (g) (h)

Figure 6.5: Snapshots of frames 1-4 for Mobile(QCIF): (a)-(d) Encoded with JM12.4. (e)-(h) Encoded with TDFMD



(a) (b) (c) (d)

(e) (f) (g) (h)

Figure 6.6: Snapshots of frames 1-4 for News (CIF): (a)-(d) Encoded with JM12.4. (e)-(h) Encoded with TDFMD

6.5 Selective Coefficient Fast Mode Decision (SCFMD)

In the first part of this chapter we have seen that the transform coefficients of the residual pixels give an indication of the possible encoding mode in a MB. Depending upon the number of zero coefficients in the transformed block, the mode decision is made. The SCFMD algorithm is based on the calculation of only a few selected transform coefficients from a block. Depending upon the values of these transform coefficients, the fast encoding algorithm is proposed.

When all the transform coefficients in a MB are zeros, it is an AZCB. It is seen that there are many instances where after the transformation and quantization on a block of a residual, there is only one non zero transform coefficient in that block. Such MBs do not qualify for an AZCB. This single coefficient might require 10-12 bits for encoding with very little improvement in the rate distortion consideration. If this coefficient is ignored and the MB is considered to be an AZCB and in addition, the other conditions for the SKIP mode are satisfied, then the MB can be encoded in the SKIP mode. Such MBs when encoded in the SKIP mode do not significantly affect the visual quality of the video.

6.5.1 Observation On Selective Transform Coefficients

0	1	5	
2	4		
3			

Figure 6.7: First six low frequency transform coefficients in zig-zag order

The H.264/AVC uses 4×4 integer DCT transform on the residual block. In the DCT transform, it is well known that the energy of the transformed coefficients is concentrated in the lower frequency coefficients. Human eye is more sensitive to the low frequency de-

tails rather than the higher frequency details. From experiments, it is observed that if the low frequency transform coefficients are zero valued then in most cases the high frequency transform coefficients are also zero valued. To do a thorough study, we performed $P_{16\times16}$ ME on different classes of sequences. Transformation and quantization were performed on the motion compensated residual to obtain a block containing sixteen 4×4 transformed subblocks. The first six low frequency coefficients in the zig-zag order (Figure 6.7) hereby called " the first six coefficients " of each 4×4 subblock were studied. The following observations were made:

(i) If the first six coefficients of a subblock are zero, there is on an average 95% probability that the remaining coefficients of the subblock are also zero

(ii) If the above is true for all the sixteen 4×4 subblocks, then in almost all cases the MB has all zero coefficients and therefore it is an AZCB

(iii) There are some MBs where despite the first six coefficients of all sixteen subblocks being zeros, are not an AZCB indicating there are a few non-zero valued high frequency transform coefficients. Out of such MBs, it is observed that in 98% of the cases, the MB contains only one non-zero high frequency component. If this coefficient is ignored (i.e set to zero), then the MB can be considered to be an AZCB

(iv) If only one coefficient out of the first six coefficients is non zero, then in about 85% of the cases the MB is encoded in the $P_{16\times16}$ mode when compared with the final encoded mode from full search ME

Figure 6.8 illustrates these observations. From the above analysis, we conclude that if the first six coefficients of each of the 16 transform subblock is zero, then it is most probably an AZCB or a block having at most one non zero coefficient. On the other hand, if only one coefficient in the first six coefficient is non-zero and this is true for all subblocks, then it is more likely to be encoded in the $P_{16\times16}$ mode.

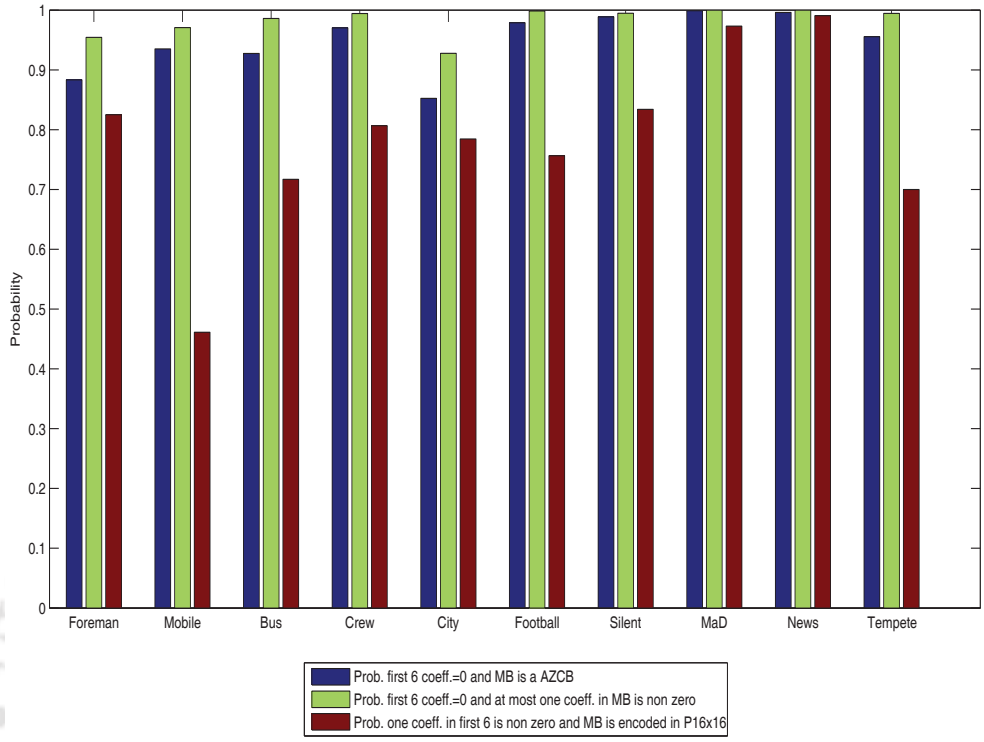


Figure 6.8: Observations on the first six transform coefficients

6.6 Proposed SCFMD Algorithm

From the discussions in the previous section we note that by computing the first six coefficients of each of the sixteen subblocks, inference regarding possible encoding modes may be made. The residuals of the motion compensated prediction is required for the transformation process. In SCFMD, ME is performed with $P_{16\times16}$ mode and the motion compensated residual $\hat{E}_{MC}$ is obtained. $\hat{E}_{MC}$ is divided into sixteen 4x4 subblocks, SB_k, k=1,. . . ,16.

For each SB_k, the unscaled forward transformed block are obtained from $W = C(SB_k)C^T$ using the procedure described in Section 2.7. The quantized and the scaled coefficients for the subblock is obtained from (2.10) and is given as

$$|Z_{ij}| = sign(W_{ij})[(|W_{ij}|\,.MF + f) \gg qbits)], \tag{6.1}$$

where f is $2^{qbits}/6$ and $qbits = 15 + floor(QP/6)$. In the SCFMD algorithm, Z_{ij} is cal-

culated only for the six positions given in Figure 6.7 instead of all the sixteen coefficients for the subblock. Depending on the values of these six coefficients, the mode decision is taken.

For each SB_k, find Z_{ij} for the first six low frequency components. Let C(k) be the number of non-zero coefficients in the first six coefficients. Depending upon the value of C(k), we define ZC(k) as

$$ZC(k) = \begin{cases} 0 & if \quad C(k) = 0 \quad (all \quad zero \quad coefficients) \\ 1 & if \quad C(k) = 1 \quad (one \quad non-zero \quad coefficient) \\ \alpha & if \quad C(k) > 1 \quad (more \quad than \quad one \quad non-zero \quad coefficient) \end{cases}$$

where α is any number greater then 16. If ZC(k) is zero, it indicates a ZB. We define ZC_{Tot} as,

$$ZC_{Tot} = \sum_{k=0}^{15} ZC(k)$$

From the value of ZC_{Tot}, decision on a MB qualifying for an AZCB, or a 16×16 (CB16) block or for further ME is taken according to the flowchart given in Figure 6.9. The final encoding mode for an AZCB is the SKIP mode and for CB16 the encoding mode is $P_{16\times16}$. For these two blocks, ME need not be performed as $P_{16\times16}$ ME has already been performed to find the E_{MC}. In case the MB goes for further ME, the DMD method as described in Section 6.3 is chosen.

6.7 Experimental Observations

Simulations carried out on different sequences show the effectiveness of the approach. The results obtained are analyzed in this section.

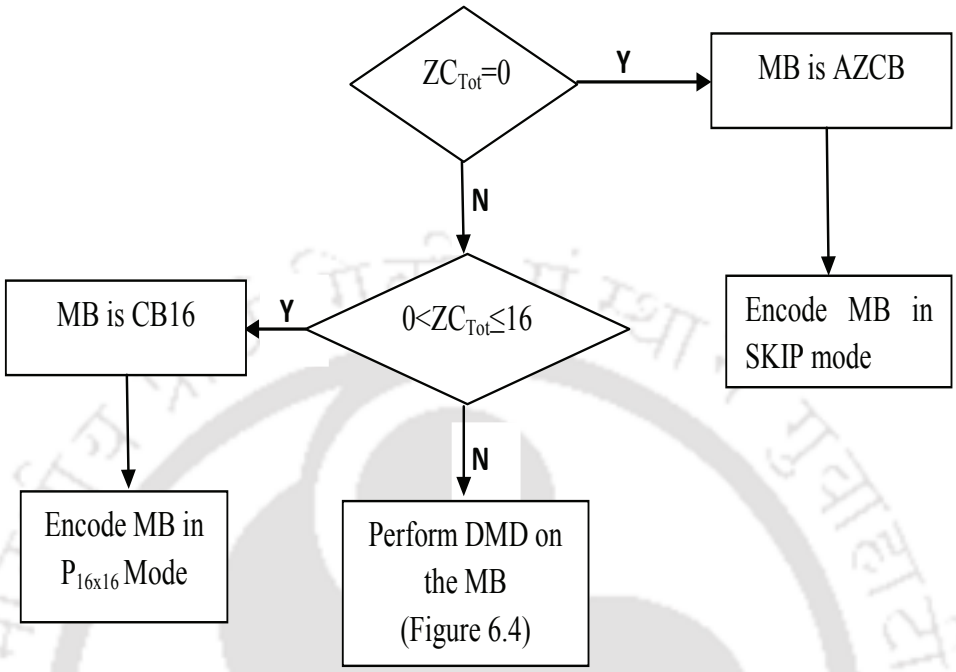


Figure 6.9: Final Encoding Mode Decision

6.7.1 *Distortion and Compression Ratio Comparisons*

Table 6.3 lists the performance of the proposed algorithm. The results show an encoding time saving of about 77% with 0.03 dB PSNR loss and 1.35 % increase in bitrate.

6.7.2 *PSNR-Rate Curves*

The performance of the proposed method is compared with the JM12.4 encoder for a wide range of QP. Figure 6.10 shows the rate distortion in different sequences. The performance of the proposed method is similar to that provided by the JM12.4 software.

Table 6.3: Performance Comparison For Different Sequences

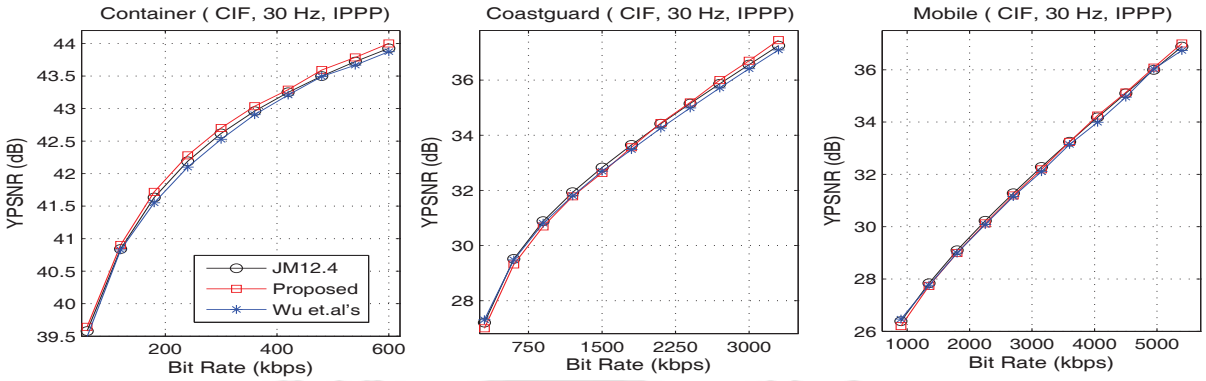
Class	Sequence	Performance Comparison								
		SCFMD Algorithm			Wu et al.'s [87]			Liu et al.'s [72]		
		ΔPSNR dB	ΔRate (%)	ΔT (%)	ΔPSNR dB	ΔRate (%)	ΔT (%)	ΔPSNR dB	ΔRate (%)	ΔT (%)
CIF Class A	News	0.05	1.23	89.20	0.02	0.84	39.23	0.11	2.41	48.53
	MaD	0.04	1.19	90.26	0.01	0.62	43.21	0.09	1.76	55.25
	Container	0.07	1.31	92.64	0.05	1.21	46.18	0.08	1.61	52.59
	Hall	0.09	1.51	88.01	0.06	1.27	34.67	0.07	1.35	46.27
Class B	Foreman	0.07	1.33	79.31	0.07	1.32	34.90	0.05	1.04	39.86
	Coastguard	0.04	1.16	74.08	0.05	1.18	26.30	0.02	0.71	33.56
	Ice	0.09	1.52	82.47	0.07	1.29	45.68	0.05	1.09	54.12
	Harbour	0.09	1.54	62.59	0.05	1.15	21.56	0.06	1.12	23.26
Class C	Flower	0.10	1.70	71.11	0.05	1.17	36.85	0.06	1.16	42.86
	Stefan	0.06	1.28	68.60	0.02	0.84	32.25	0.08	1.59	39.54
	Tempete	0.07	1.36	65.40	0.09	1.48	27.21	0.09	1.63	35.42
	Mobile	0.03	1.07	72.22	0.09	1.54	12.35	0.07	1.45	19.28
	Average	**0.06**	**1.35**	**77.93**	**0.05**	**1.15**	**33.36**	**0.06**	**1.41**	**40.87**
QCIF Class A	Claire	0.10	1.68	92.66	-0.01	-0.95	47.35	0.01	0.51	62.45
	MissAmerica	0.08	1.43	91.83	0.02	0.81	48.23	0.03	0.89	60.27
	Suzie	0.08	1.49	92.29	0.05	1.24	42.91	0.06	1.32	59.65
Class B	Foreman	0.09	1.53	79.53	0.02	0.85	30.25	0.05	1.21	41.18
	Silent	0.07	1.36	90.29	0.06	1.41	42.62	0.03	0.91	55.24
	Crew	0.05	1.21	77.53	0.05	1.20	19.64	0.04	1.12	30.43
Class C	Football	0.06	1.30	71.58	0.05	1.24	28.84	0.05	1.26	31.75
	Mobile	0.09	1.54	70.99	0.07	1.39	15.32	0.08	1.71	21.29
	Soccer	0.03	1.08	75.89	0.03	0.95	20.19	0.04	1.10	27.65
	Average	**0.07**	**1.40**	**82.51**	**0.03**	**0.90**	**33.21**	**0.04**	**1.11**	**43.32**

ΔPSNR(+/-): picture quality loss/gain measured in dB
ΔRate(+/-): bitrate increase/decrease measured as a %
ΔT(+/-): encoding time saving/loss measured as a %

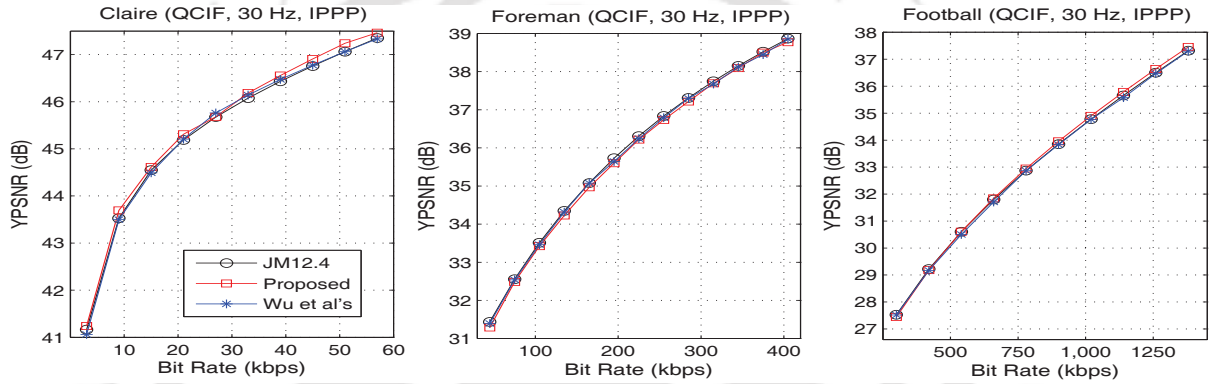
6.8 Conclusions

The transform coefficients for a MB are required for the calculation of the RD cost during the mode decision process. Thus the transform coefficients are available at the encoder during the encoding process. This chapter proposed two algorithms in the transform domain: the Transform Domain based Fast Mode Decision (TDFMD) and the Selective Coefficient Fast Mode Decision (SCFMD). Depending upon the number of zero valued transform coefficients, the TDFMD algorithm yields fast mode decision for both the intra and the inter prediction process. Simulations have shown that TDFMD algorithm gives significant encoding time saving.

It is well known that the energy of the transformed coefficients is concentrated in the lower frequency coefficients. The SCFMD algorithm uses this property for the mode



(a) RD curve for CIF sequences of Class A,B and C



(b) RD curve for QCIF sequences of Class A,B and C

Figure 6.10: RD curves of sequences from different classes

decision process. The decision on the encoding mode is taken based on six low frequency transform coefficients. Experimental results show the SCFMD algorithm gives considerable time saving as compared to the JM reference software.

7

Subband/DCT Based Fast Mode Decision Algorithms

Contents

7.1 Subband/DCT based Fast Mode Decision for H.264/AVC (SBFMD) 132

7.2 Experimental Observations 136

7.3 Subband/DCT based Fast Encoding Algorithm For Scalable Video Coding 138

7.4 Experimental Observations 143

7.5 Conclusions 144

Wavelet analysis can be used to divide the information of an image into approximation and detail subsignals. These subsignals can be obtained by passing the signal through a filter bank. If a signal is put through a high-pass filter, high frequency information is kept, low frequency information is lost. When put through a low pass filter, low frequency information is kept, high frequency information is lost. The approximation subsignal (low-pass) shows the general trend of pixel values and the three detail subsignals (high-pass) show the vertical, horizontal and diagonal details. The approximation subsignal which retains most of the information of the signal can be used for analysis of the image.

In this chapter, we make use of wavelet analysis of the input image to obtain the four subbands. These subbands are utilized for the proposed fast mode decision algorithm. In the first part of the chapter, a fast encoding method for the H.264/AVC standard is described. The subband decomposition of the input signal is computed using the traditional wavelet decomposition. Four subbands namely the LL, LH, HL and HH are obtained. Most of the information is present in the LL subband and hence exhaustive encoding is done for this subband. The information about encoding modes obtained from the LL subband is used for the encoding the other three subbands. Motivated by the encouraging results from the proposed algorithm applied to H.264/AVC, a new algorithm based on subband decomposition is proposed for fast encoding of the SVC video coding standard. These algorithms are described in the following sections.

7.1 Subband/DCT based Fast Mode Decision for H.264/AVC (SBFMD)

The proposed fast mode decision algorithm is based on the subband decomposition of the input image. Figure 7.1 shows the first frame of the Mobile sequence that has been decomposed into four subbands.

Among all the subbands, the LL subband has a good similarity with the original image and this subband retains most of the information. Although the resolution of the resulting LL subband has reduced, the image is still recognizable. This is an useful

(a) Original image of Frame 1 of Mobile(CIF) Sequence

(b) Decomposed image into four subbands

Figure 7.1: Example to demonstrate the Wavelet Decomposition of an image

property of the subband decomposition. The other three subbands contain the high frequency components. These subbands preserve the vertical, horizontal and the diagonal details of the image. Since the LL subband contains the gross information of the pixel values, information from the LL subband can be used for the encoding of the higher subbands.

7.1.1 Proposed SBFMD Algorithm

The proposed SBFMD algorithm consists of two parts: the fast intra prediction and the fast inter prediction methods. The fast intra prediction is described first followed by fast inter prediction process.

- *Step1: Decomposition of the input video into four subbands*

  This step is common for both intra and inter prediction process. The input video (at any resolution) is first decomposed using Daubechies 9/7 filter to obtain LL, LH, HL and HH subbands. The LL subband which is at half the resolution of the input video is directly used for both the intra and the inter prediction. The LH, HL and HH subbands use the information from the LL subband for the intra and the inter prediction.

A. Intra Prediction

- *Step2: Exhaustive LL band intra prediction*

  The LL subband contains the maximum detail of the input video. The resolution of the LL subband is half the resolution of the original image. Hence, number of MBs become one fourth of that of the original number of MBs. Exhaustive search is performed only for the LL subband so that minimum detail is lost. The best mode is decided based on the RDO performance.

- *Step3: Intra Mode selection for High Frequency Subbands*

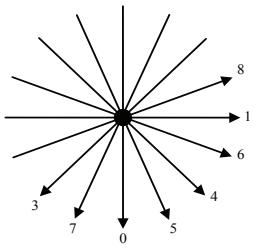
  The subband decomposition can be thought as signal decomposition in a set of spatially oriented frequency subbands. The LH, HL and HH subbands respectively represent the horizontal, vertical and the diagonal high frequencies. Thus the high frequency subbands can be viewed as directional bands. Hence only directional prediction pertaining to the respective directions of the subbands are considered for the intra prediction. The candidate modes for intra prediction in these subbands will be those modes which are inclined in these directions. The choice of I4MB or I16MB will depend upon the MB partitioning of the collocated MB in the LL subband.

  – If the collocated MB in the LL subband is encoded in the I16MB mode, then the MB in high frequency subbands are also encoded in the I16MB mode. There are only four directional modes for the I16MB prediction. The LH subband is encoded in mode 1 (horizontal), the HL subband is encoded in mode 0 (vertical) and the HH subband is encoded in mode 3 (plane).

  – When the collocated MB in the LL subband is encoded in the I4MB mode, then for all the three subbands the modes to be used for the RD calculation are chosen from a look-up table. For the LH subband having horizontal information, only four modes (out of the nine modes) having horizontal orientations

Table 7.1: Look-up Table for High Frequency Subbands

Bands	High Frequency Subband Modes								
	0	1	2	3	4	5	6	7	8
LH		√	√				√		√
HL	√		√			√		√	
HH			√	√	√		√		



are used for the mode decision. Similarly vertical modes for the HL and diagonal modes for HH subbands are used for the prediction. For the HH subband, mode 6 which is in the horizontal down direction is also chosen since it is observed that motion in videos in general is more in the horizontal direction than in the vertical direction. The best mode is selected based on the best R-D performance. The overall search region for the intra mode prediction is greatly reduced by this approach. The look-up table for the corresponding modes to be chosen for the three subbands is given in Table 7.1

B. Inter Prediction

- *Step2: Exhaustive LL band inter prediction*

  For inter prediction, full search VBS-ME is performed only for the LL subband and the best mode is decided based on the RDO performance. The motion vectors (MV) of the LL subband are stored for their use in the high frequency bands.

- *Step3: Inter Prediction for LH, HL and HH subbands*

  No motion search is performed for these subbands. The MVs obtained for the collocated MB in the LL subband (Step 2) is used for the motion compensation in the higher frequency bands. Thus, there is no additional motion search performed for the high frequency subbands. The complexity is thus greatly reduced.

7.2 Experimental Observations

Simulation results of the proposed SBFMD algorithm on some test sequences is discussed here. The reduction in the number of modes to be searched using the SBFMD algorithm as compared to the JM reference software is given. The encoding time saving for different sequences are also tabulated.

7.2.1 Complexity Analysis

Table 7.2 summarizes the number of modes selected for both the intra mode and the inter mode RDO calculation in the proposed SBFMD algorithm. For the intra prediction, RDO calculation for the LL band is 148 searches per MB for luma component. For the high frequency bands the total searches are 4 per MB. For the inter prediction, the RDO calculation for the LL subband is 19 searches per MB for luma component. For the high frequency subbands, there will be no RDO calculations for inter prediction. There will be only the motion compensation with the MVs inherited from the LL subband. This results in considerable reduction in complexity in inter prediction process.

Table 7.2: Reduction In Intra and Inter RDO Calculations

Type of Sequence	No. of RDO calculation per I frame for JM12.4	No. of RDO calculation per I frame for SBFMD method	No. of RDO calculation per P frame for JM12.4	No. of RDO calculation per P frame for Proposed method
CIF	58608	33660	7524	1881
QCIF	14652	8500	1881	475

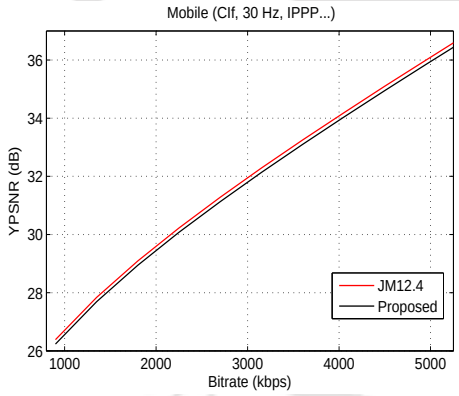
7.2.2 *Distortion and Compression Ratio Comparisons*

Table 7.3 gives the percentage time saving, differences in PSNR and increments in bitrate. It is observed that the SBFMD algorithm gives an encoding time saving of around 60% for all the sequences. However there is a PSNR loss of 0.53 dB and bitrate increase of 1.80%. Figure 7.2 shows the RD curves for the QCIF and CIF sequences. Its performance however is inferior to the proposed methods in the transform domain.

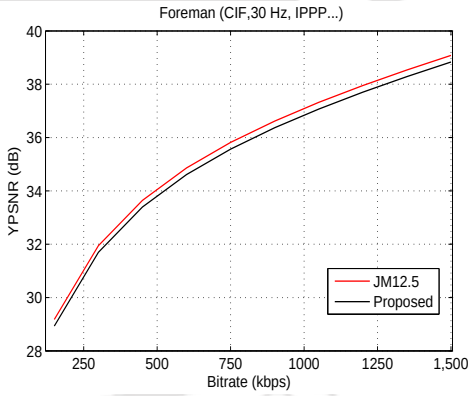
Table 7.3: Results For Different Sequences

Format	Sequence	ΔPSNR(dB)	ΔRate(%)	ΔT(%)
QCIF	Foreman	0.52	1.95	62.80
	City	0.51	2.12	62.15
	Mobile	0.51	2.56	60.12
	Bus	0.35	1.59	63.51
	Football	0.49	1.47	64.60
	Soccer	0.70	2.24	64.12
	Crew	0.84	2.16	60.62
CIF	Foreman	0.60	2.09	62.07
	City	0.54	2.46	60.15
	Mobile	0.42	1.65	58.77
	Bus	0.64	1.83	60.12
	News	0.32	1.07	66.15
	Silent	0.34	0.96	65.80
	Tempete	0.67	2.27	61.11
	Mother-daughter	0.36	1.24	67.34
	Crew	0.80	2.59	65.23
	Average	**0.53**	**1.89**	**62.79**

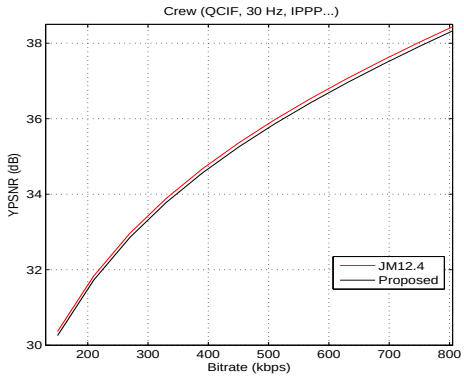
ΔPSNR(+/-): picture quality loss/gain measured in dB
ΔT(+/-): encoding time saving/loss measured as a %
ΔRate(+/-): bitrate increase/decrease measured as a %



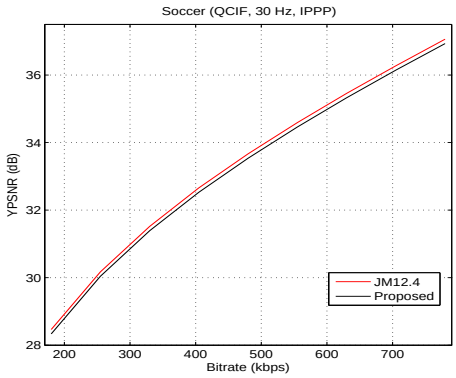
(a) RD Performance for Mobile(CIF)



(b) RD Performance for Foreman(CIF)



(c) RD Performance for Crew(QCIF)



(d) RD Performance for Soccer(QCIF)

Figure 7.2: RD Performance for various sequences in the CIF and the QCIF resolution

7.3 Subband/DCT based Fast Encoding Algorithm For Scalable Video Coding

Applications of video communication range from low bandwidth application like multimedia messaging, video telephony etc. to high bandwidth applications like HDTV broadcasting, Blu-ray disc, optical storage media etc. These diverse applications require varying video transmission rates and storage capabilities. The Scalable Video Coding (SVC) standard [103] is an attractive solution to the problems posed by the modern transmission characteristics. The SVC has been standardized as an extension of the H.264 /AVC video coding standard [104]. The objective of the SVC is to enable the creation of a video bitstream that is structured in layers.

The coding complexity of the SVC standard is very high. For the intra prediction in the SVC standard, there are three modes namely the 4×4 mode (I4MB), 8×8 mode (I8MB) and the 16×16 mode (I16MB) [105]. The prediction modes I4MB and the I16MB is similar to that in H.264/AVC standard. The I8MB mode is similar to that of the I4MB prediction with nine prediction directions to choose from. The time required for the intra mode prediction and decision is large and is a practical problem in real-time applications. The encoding process is computationally complex. An attempt has been made here to reduce the coding complexity of the intra coding process of the SVC video coding standard.

Several approaches have been proposed to reduce the complexity and the time in the mode decision process in the SVC standard. In [106], [107] the authors suggest that the 16×16 prediction mode be skipped and only the vertical, horizontal and DC modes of the I4MB be used for prediction. These approaches speed up the encoding process. However, skipping modes or using some pre-defined modes result in reduction of the overall coding efficiency. To ensure that the encoding process is efficient, we propose a method for fast mode decision for intra prediction in the SVC standard in which all the encoding modes are retained. Mode selection is based on the QP and the directional feature of the MBs.

The SVC video is structured in layers. The base layer is encoded in H.264/AVC compatible bitstream. In the proposed algorithm, only dyadic scalability is considered. The inherent scalability offered by traditional wavelet decomposition as described in the earlier part of the chapter is used for the different layers. Similar to the SBFMD algorithm, the input video is decomposed into four subbands using wavelet decomposition. The LL subband is directly used for the base layer video. The high frequency bands are used for the enhancement layer video.

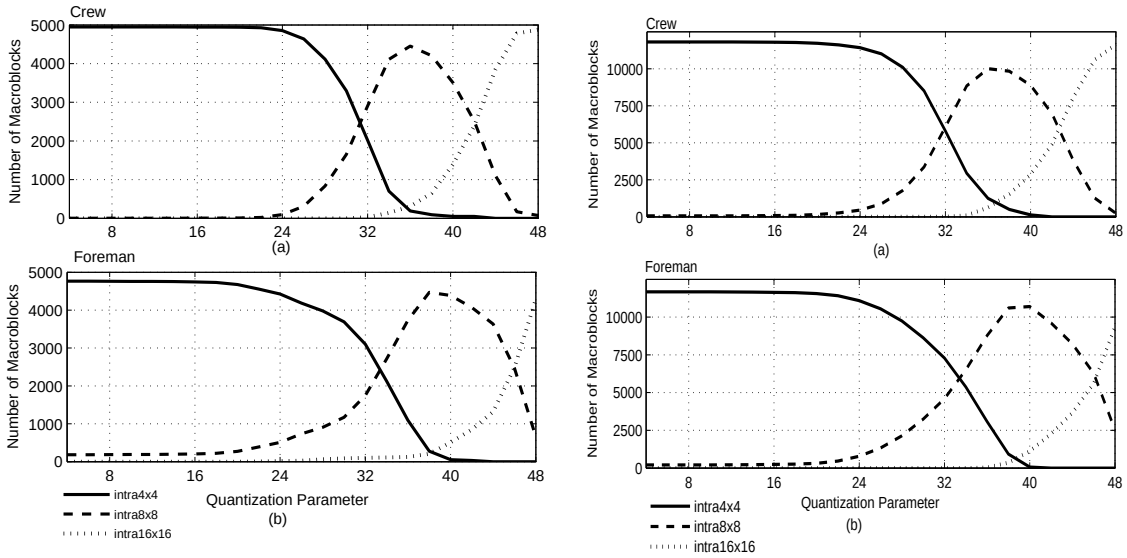
The mode distribution in the base layer and enhancement layer of the SVC is studied in detail and certain observations made. These observations on the encoding modes for the SVC standard is described in the next section. The proposed mode decision for the intra prediction for the SVC is based on these observations.

7.3.1 Analysis of Intra prediction modes

We performed exhaustive intra prediction search for both the base layer and the enhancement layer. The experimental setup consisted of a base layer at QCIF resolution while the enhancement layer is at dyadic upsampling resolution that is the CIF resolution. The study of the correlation between the layers at different resolutions gives us the following observations.

A. Mode Dependency of Base Layer at Different QP

Figure 7.3 (a) gives the distribution of the prediction modes for the first 50 frames at different QP for two sequences Foreman and Crew. Similar results have also been observed for other sequences namely City, Mobile, Bus and Football. It is observed that for QP up to 24, most of the MBs are encoded in the I4MB mode. For values of QP above 24, more MBs are encoded in the I8MB mode. For QP above 45, almost all the MBs are encoded in the I16MB mode.



(a) Base Layer (QCIF) (b) Enhancement Layer (CIF)

Figure 7.3: Prediction Mode Distribution (Base Layer QCIF and Enhancement Layer CIF)

B. Mode Dependency of Enhancement Layer at Different QP

We perform exhaustive intra prediction for the enhancement layer. Figure 7.3 (b) gives the distribution of the prediction modes for the first 50 frames at different QP for sequences for which the base layer analysis was done. For the enhancement layer also, the mode dependency is similar to that of the base layer. Here for QP up to 24, most of the MBs are encoded in the I4MB mode. The distributions are seen to be similar to the base layer prediction indicating high correlation between the layers.

7.3.2 Proposed Intra Fast Mode Decision Algorithm for SVC

The wavelet decomposition of the input video is done and the LL subband is used as the base layer. The observations made in the Section 7.3.1 are used for choosing the modes based on the QP for the base layer (QP_{BL}) and QP for the enhancement layer (QP_{EL}). The steps are described next.

Steps for Fast Mode Decision Algorithm

- *Step1: Decomposition of the input video into four subbands*

  We first decompose the input video which is at a CIF resolution using analysis filter (AF). We employ the Daubechies 9/7 filter set for the subband synthesis of the individual frames. We get the LL, LH, HL and HH subbands. The LL subband which is at the QCIF resolution contains maximum information of the original image. This subband is utilized directly for the base layer. The LH, HL and HH subbands are used for the enhancement layer.

- *Step2: Exhaustive base layer prediction based on QP*

  We define two thresholds T_1 and T_2. Referring to Figure 7.3, we see that below the QP value of 24, most of the MBs are encoded in the I4MB mode and above 40, more MBs are encoded in the I16 MB mode. Hence T_1 and T_2 are chosen at 24 and 40 respectively. If the $QP_{BL} < T_1$, prediction is performed with only I4MB using all the nine modes. For $QP_{BL} \geq T_2$ only I16MB search are performed. For $T_1 \leq QP_{BL} < T_2$ both the I4MB and the I16MB prediction is done. The mode information is stored for future reference.

- *Step3: Enhancement Layer Intra Prediction*

  The enhancement layer prediction modes have a high correlation with the modes from the base layer prediction. The difference between the reconstructed base layer frame and the LL subband is used for the prediction of the LL subband for the enhancement layer. The LH, HL and the HH band are directly used for the prediction. These are directional bands and the different modes for the prediction are chosen from the directions of the bands as already discussed in Section 7.1.1. The look up table given in Table 7.1 is used for the mode selection of the high frequency subbands. The enhancement layer video is formed by passing the four subbands through a synthesis filter.

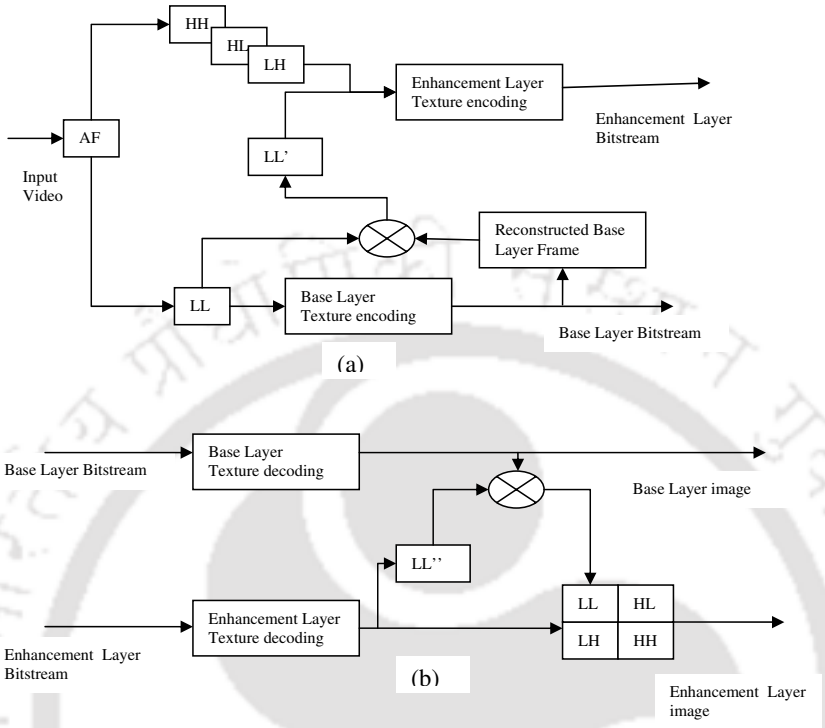


Figure 7.4: Block Diagram for encoding and decoding of the video sequence

- *Step4: Prediction based on Quantization Parameter*

  For QP of base layer above threshold T_2, the base layer reconstructed MB for the residual prediction is too coarse for enhancement layer reference. The enhancement layer intra prediction is performed independent of the base layer prediction. The base layer prediction is done using the I16MB prediction mode. For the enhancement layer, for $QP_{EL} < T_1$, I4MB prediction is performed. For $T_1 \leq QP_{EL} < T_2$ both I4MB and I8MB prediction is done and for$QP_{EL} \geq T_2$, only I16MB prediction is performed.

Figure 7.4 shows the block diagram for the encoding and the decoding process for the base and the enhancement layer.

Table 7.4: Simulation Results for Base Layer and Enhancement Layer at different QP

Sequence	Quantization Parameter (Base)						Quantization Parameter (Enhancement)					
Foreman		8	16	24	32	40	QP(Base)=16	8	16	24	32	40
	Δ PSNR (dB)	0.02	0.01	0.07	0.01	0.03	Δ PSNR(dB)	0.02	0.01	0.08	0.07	0.05
	Δ T (%)	43	42	44	28	26	Δ T (%)	55	56	60	59	57
	CompRed(%)	22	22	22	5	5	Comp Red(%)	70	70	24	24	85
Crew	Δ PSNR(dB)	0.05	0.01	0.01	0.02	0.05	Δ PSNR(dB)	0.05	0.01	0.01	0.02	1.58
	Δ T (%)	40	41	44	26	27	Δ T (%)	54	52	56	56	57
	CompRed(%)	22	22	22	5	5	Comp Red(%)	70	70	24	24	85
City	Δ PSNR(dB)	0.02	0.01	0.02	0.01	0.01	Δ PSNR(dB)	0.02	0.01	0.01	0.01	0.01
	Δ T (%)	38	42	42	24	25	Δ T (%)	52	52	57	56	60
	CompRed(%)	22	22	22	5	5	Comp Red(%)	70	70	24	24	85
Bus	Δ PSNR(dB)	0.06	0.01	0.01	0.38	0.11	Δ PSNR(dB)	0.06	0.01	0.06	0.03	0.01
	Δ T (%)	40	42	42	26	25	Δ T (%)	53	53	49	56	53
	CompRed(%)	22	22	22	5	5	Comp Red(%)	70	70	24	24	85
Mobile	Δ PSNR(dB)	0.01	0.02	0.74	0.01	0.01	Δ PSNR(dB)	0.62	0.55	0.74	0.04	0.09
	Δ T (%)	42	44	41	23	23	Δ T (%)	52	52	48	54	56
	CompRed(%)	22	22	22	5	5	Comp Red(%)	70	70	24	24	85
Football	Δ PSNR(dB)	0.01	0.01	0.01	0.01	0.04	Δ PSNR(dB)	0.01	0.02	0.01	0.01	0.04
	Δ T (%)	42	44	45	26	26	Δ T (%)	55	53	49	50	51
	CompRed(%)	22	22	22	5	5	Comp Red(%)	70	70	24	24	85

ΔPSNR(+/-): picture quality loss/gain measured in dB
Δ T(+/-): encoding time saving/loss measured as a %
CompRed(+/-): complexity decrease/increase measured as a %

7.4 Experimental Observations

The proposed algorithm is implemented as per the SVC Reference Software Joint Scalable Video Model (JSVM8). The CIF sequences Foreman, Crew, City, Bus, Mobile and Football have been used for the implementation. We employ the Daubechies 9/7 filter set for the subband decomposition of the individual frame. The LL subband at QCIF resolution is directly used for the base layer encoding. The LL subband along with the high frequency subbands are used for the enhancement layer predicton. Table 7.4 lists the performance of the proposed scheme with the JSVM8 in terms of PSNR loss, overall saving in the encoding time and reduction in the complexity (CompRed) for the base layer and enhancement layer in terms of the reduction in the number of searches performed at each layer.

The results shown here for the enhancement layer are with base layer QP_{BL}=16. For both the base layer and the enhancement layers , there is substantial saving in the encoding time and the complexity. The PSNR loss in also negligible.

Figure 7.5: Enhancement Layer Reconstructed Frame at different QP(base)

For high QP in base layer, the reconstructed picture for use in the enhancement layer is too coarse. Due to this, even for low values of QP in the enhancement layer the reconstructed enhancement layer picture gives relatively poor picture quality. This is shown in Figure 7.5 where frame 1 of Football sequence is shown encoded at a low enhancement layer QP of 8 and different base layer QP of 8 and 40 respectively. The degradation in quality at high QP is clearly visible.

7.5 Conclusions

In this chapter, we have proposed a Subband/DCT based Fast Mode Decision (SBFMD) algorithm for H.264/AVC standard. The method is based on the wavelet decomposition of the input video into four subbands. The LL subband was used for the exhaustive intra and inter prediction process. Information of the encoding modes and the motion vectors of the LL subband were used for the encoding of the high frequency subbands. The algorithm when applied to different test sequences have shown good complexity reduction but the distortion of the pictures encoded is higher than those obtained form the fast encoding methods described so far in the previous chapters.

This chapter also proposed Subband/DCT based Fast Mode Decision algorithm for

the SVC standard. The LL subband was used for the base layer and encoded in the H.264/AVC standard. For the enhancement layer at dyadic scalability, the prediction was done using the base layer reconstructed image and the three high frequency subbands. Information from the base layer was used for the prediction in the enhancement layer. Results presented show the effectiveness of the subband decomposition applied to the SVC standard.

8

Fast Mode Decision for Intra only Spatial SVC

Contents

8.1 Analysis of Intra prediction modes 148
8.2 Proposed Fast Mode Decision Algorithm 149
8.3 Experimental Observations 153
8.4 Conclusions 156

In the previous chapter, it is seen that the objective of the SVC is to enable the creation of a video bitstream that is structured in layers. Scalable video coding is attractive due to the capability of reconstructing lower resolution or lower quality signals from partial bit streams. The SVC standard [103] gives enhanced coding efficiency but at the cost of higher complexity. Three types of scalability modes are provided that include temporal scalability (increase/decrease of frame rate) , spatial scalability(increase /decrease of picture resolution) and fidelity scalability (increase/decrease of quantization accuracy). Among the scalability provided, the spatial scalability is an important feature required for transmission over variable bandwidth and devices with varying display resolutions over a heterogeneous network [108]. One of the key feature in the SVC standard is the inter layer prediction between different layers [44], [109].

For intra prediction in the SVC standard, there are three modes namely the INTRA 4×4 (I4MB), INTRA 8×8 (I8MB) and the INTRA 16×16 (I16MB) [105] and the I_{BL} mode. There is a strong correlation between the different layers in the scalable video. This relation between the layers can be exploited for the fast mode prediction process. In this chapter, a fast intra mode decision algorithm for the spatial SVC is proposed. In this algorithm, the choice of I4MB and the I8MB modes depends upon the QP in the base layer and the enhancement layer. Making use of the high directional similarity between the different resolution layers in scalable video, the candidate modes in the enhancement layer can be predicted depending upon the base layer prediction. Base layer is predicted using rigorous prediction technique. Results from the base layer are used for the enhancement layer prediction.

8.1 Analysis of Intra prediction modes

A detailed analysis of the intra prediction modes for the SVC was described in the previous chapter. The dependance of the intra modes on the QP for the base layer and the enhancement layer has already been discussed in Section 7.3.1. Also, from the similarity of the distribution of encoding modes between the two layers it may be concluded that the

base layer and enhancement layer are highly correlated. From a study of the correlation between the different modes in the base layer and the enhancement layer, the following observation is made.

8.1.1 Correlation of Modes Across Layers

After carrying out an exhaustive base layer and enhancement layer prediction, the directional prediction between layers is analyzed. The MBs may be coded in the I4MB, or I8MB, or I16MB prediction modes. The prediction direction depends upon which one of the nine modes of the I4MB or I8MB, or four modes of I16MB gives the best prediction. To ascertain whether the directional features are preserved in both the layers, we compare the prediction modes of the enhancement layer MBs with that of the corresponding base layer MBs. For the comparison, we consider a region of $\pm 45^{\circ}$ of the direction of the base layer prediction mode. If the encoding mode of the enhancement layer is inclined within this region of $\pm 45^{\circ}$ with respect to the base layer direction, then we consider them to have similar directional features. For example, if the base layer MB has the best prediction in the 'horizontal direction' (mode 1) in the I4MB mode, then the enhancement layer is said to have similar directional features if the corresponding MB in the enhancement layer has the modes 'horizontal' (mode 1) or 'horizontal down' (mode 6) or 'horizontal up'(mode 8) of I4MB or I8MB prediction. It is observed that the probability of similar directional features in both the layers is above 75% in most of the cases and this correlation increases when the base layer and enhancement layer have same QP. If the base layer QP (QP_{BL})is high, then the reconstructed picture becomes too coarse for use in the enhancement layer prediction. This is true even if the enhancement layer QP (QP_{EL}) is low.

8.2 Proposed Fast Mode Decision Algorithm

From the above discussions, it is found that the encoding modes for both the base and enhancement layer are dependent on the QP. The encoding mode of the enhancement layer is related to the base layer encoding mode of the collocated MB. These observations

are utilized for fast mode decision for intra coding in SVC. The different steps for the algorithm are as follows.

8.2.1 Steps in Fast Mode Decision in Spatial SVC

- **Step1: Rigorous Base Layer Prediction**

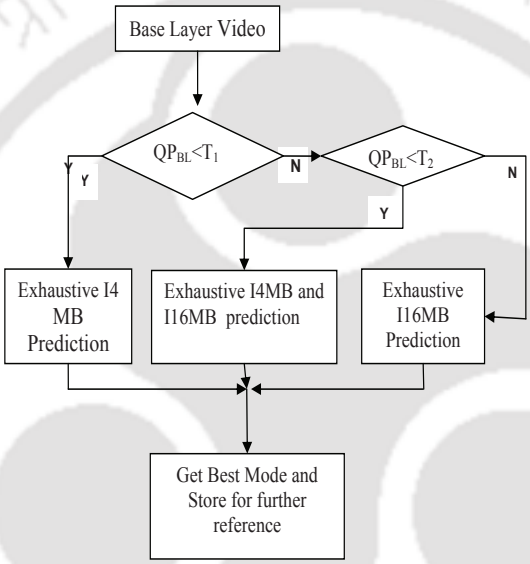


Figure 8.1: Flow Chart for proposed Base Layer Prediction

The flowchart for the base layer prediction is given in Figure 8.1. The intra mode decision is QP dependent as observed in Section 7.3.1. From Figure 7.3, it is seen that for QP_{BL} upto 24, most of the MBs are encoded in I4MB modes and therefore T_1 is taken as 24. For QP_{BL} values above 45, most of the MBs are encoded in I16MB mode. Thus T_2 is taken as 45. For the base layer prediction, if $QP_{BL} < T_1$, exhaustive intra prediction is performed with only I4MB prediction modes. For $T_1 \leq QP_{BL} < T_2$, both I4MB and I16MB intra predictions are performed to identify the best mode. For $QP_{BL} \geq T_2$, only I16MB exhaustive intra prediction is performed. The mode information is stored for future reference. The mode is selected based on the best R-D performance and is stored for further reference for the upper layers.

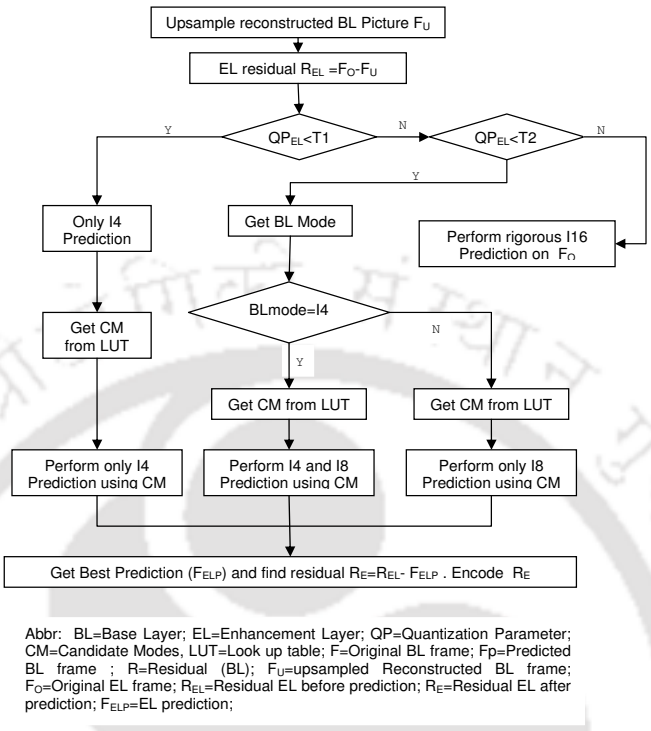


Figure 8.2: Flow Chart for proposed Enhancement Layer Prediction

- **Step2: Enhancement Layer Intra Prediction ($QP_{EL} < T_2$)**

  The enhancement layer prediction modes have a high correlation with the modes of the base layer prediction. The flowchart given in Figure 8.2 describes the proposed intra prediction method for the enhancement layer. If $QP_{EL} < T_1$, only I4MB prediction is done. For $QP_{EL} \geq T_2$, only I16MB exhaustive intra prediction is performed. If $T_1 \leq QP_{EL} < T_2$, the information for the prediction mode is taken from the stored base layer prediction information. If the base layer MB takes the I4MB mode then the corresponding enhancement layer MB prediction includes both the I4MB and I8MB modes. If the base layer MB takes the I8MB mode then for the corresponding enhancement layer MB the prediction mode is I8MB and I16MB. The candidate modes for all the cases are taken from the look up table formulated for the prediction. The candidate modes of the enhancement layer are narrowed down by retaining only those modes that have similar directions as those in base layer.

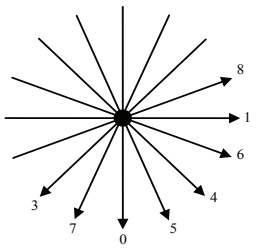
The details of the modes to be retained by the enhancement layer are discussed next.

Candidate Mode selection for enhancement layer prediction

It is observed that the enhancement layer prediction modes are highly correlated to the base layer prediction modes 8.1. As the enhancement layers have higher resolutions, more blocks are predicted in the I8MB and I16MB modes. Hence, if the base layer prediction mode is I8MB or I16MB the enhancement layer prediction modes are also I8MB or I16MB thereby excluding smaller block sizes. If the base layer prediction mode is I4MB then the enhancement layer modes are I4MB and I8MB. To have the best trade-off between the number of searches and the quality of the image, only those modes are retained for the enhancement layer which are directed along the predicted base layer direction. A look-up table is developed for a reduced set of candidate modes. All the modes are retained in the base layer prediction but only a reduced number of candidate modes are retained in the enhancement layer depending upon the base layer prediction directions. Modes selected are those that lie within ±45° of the base layer prediction direction. The look-up table for the enhancement layer is given in Table 8.1.

Table 8.1: Look-up Table for I4MB and I8MB Modes for Enhancement Layer.

Base Layer Modes	Enhancement Layer Modes								
	0	1	2	3	4	5	6	7	8
0	√		√			√		√	
1		√	√				√		√
2	√	√	√						
3			√	√				√	
4			√		√	√	√		
5	√		√		√	√			
6		√	√		√		√		
7	√		√	√				√	
8		√	√						√



- **Step2: Enhancement Layer Intra Prediction ($QP_{BL} \geq T_2$)** For $QP_{BL} \geq T_2$, the base layer prediction is not taken into consideration and enhancement layer is independently predicted wherein all the modes are again retained for the enhancement layer prediction. For QP_{BL} values above threshold T_2, the base layer prediction is too coarse for enhancement layer reference. The enhancement layer intra prediction is done independently without referring to the base layer prediction. The base layer prediction is done using the I16MB prediction mode. For the enhancement layer also, prediction is done using the I16MB mode.

8.3 Experimental Observations

The proposed algorithm is implemented as per the SVC Reference Software Joint Scalable Video Model, JSVM8. Several test video sequences have been used for the implementation. The base layer resolution is taken as the QCIF resolution. The enhancement layer is at the dyadic resolution (CIF). The results of simulations carried out on some common test sequences are given in Table 8.2. It lists the performance of the proposed scheme with the JSVM in terms of YPSNR loss (Δ PSNR) in dB, overall saving in the encoding time (ΔT) and reduction in the complexity (CompRed) for the base layer and the enhancement layer in terms of the reduction in the number of searches performed at each layer. The enhancement layer results shown in Table 8.2 are with QP_{BL} =16.

For QP_{BL} is 32, the results are shown in Table 8.3. The quality of the enhancement layer decreases with the increase in the QP_{BL}. This is clearly seen in Table 8.3 for QP_{BL}=32. When the QP_{BL} is high, the reconstructed picture in the enhancement layer is coarse even for low values of QP_{EL}. The degradation in the visual quality is shown in Figure 8.4 where the QP(EL)=8 and QP_{BL} =16, 32. The degradation in quality for QP_{BL}=32 is clearly visible. The RD curves for the two sequences Crew and Bus are given in Figure 8.3.

Table 8.2: Simulation Results at different QP

Sequence	Base Layer					Enhancement Layer (QP_{BL}=16)				
	QP_{BL}	Δ PSNR (dB)	ΔT (%)	Δ Rate (%)	Comp Red (%)	QP_{EL}	Δ PSNR (dB)	ΔT (%)	Δ Rate (%)	Comp Red (%)
Foreman	8	0.01	25	0.35	30	8	0.19	27	0.91	69
	16	0.01	27	0.30	30	16	0.38	29	0.73	69
	24	0.05	35	0.06	30	24	0.28	33	0.64	69
	32	0.02	25	0.04	15	32	0.52	36	1.10	69
	40	0.03	26	0.31	15	40	0.65	41	1.92	69
	avg	**0.02**	**27.6**	**0.21**	**24**	**avg**	**0.40**	**33.2**	**1.06**	**69**
Crew	8	0.00	25	0.02	30	8	0.24	29	1.92	68
	16	0.02	32	0.60	30	16	0.24	31	0.77	68
	24	0.03	35	0.20	30	24	0.36	34	0.72	68
	32	0.01	20	0.10	15	32	0.36	35	1.35	68
	40	0.40	20	0.80	15	40	0.35	37	2.72	68
	avg	**0.09**	**26.4**	**0.34**	**24**	**avg**	**0.32**	**33.4**	**1.39**	**68**
City	8	0.02	27	0.56	30	8	0.17	27	0.89	68
	16	0.01	29	0.28	30	16	0.37	29	1.18	68
	24	0.36	38	1.03	30	24	0.36	34	0.72	68
	32	0.01	23	0.90	15	32	0.39	35	0.29	68
	40	0.01	22	0.05	15	40	0.35	37	2.72	68
	avg	**0.08**	**27.8**	**0.56**	**24**	**avg**	**0.33**	**31.8**	**1.14**	**68**
Bus	8	0.01	25	0.53	30	8	0.15	27	1.73	68
	16	0.01	26	0.03	30	16	0.29	29	0.45	69
	24	0.02	30	0.42	30	24	0.30	34	1.09	69
	32	0.02	20	0.90	15	32	0.24	39	1.91	69
	40	-0.15	22	0.31	15	40	0.18	43	1.80	68
	avg	**0.04**	**24.6**	**0.43**	**24**	**avg**	**0.23**	**34**	**1.39**	**68**
Mobile	8	0.02	29	0.14	30	8	0.29	27	1.84	68
	16	0.01	32	0.16	30	16	0.24	31	0.77	68
	24	0.01	30	0.20	30	24	0.36	34	0.72	68
	32	0.01	20	0.17	15	32	0.36	35	1.35	68
	40	0.02	20	0.19	15	40	0.17	44	0.86	68
	avg	**0.01**	**26.2**	**0.17**	**24**	**avg**	**0.28**	**34.6**	**1.10**	**69**
Football	8	0.03	24	0.82	30	8	0.28	28	0.25	69
	16	0.01	27	0.07	30	16	0.36	32	1.67	69
	24	0.02	35	0.09	30	24	0.33	35	0.64	69
	32	0.02	22	0.90	15	32	0.11	40	0.95	69
	40	0.02	25	0.80	15	40	0.30	45	0.77	69
	avg	**0.02**	**26.6**	**0.53**	**24**	**avg**	**0.28**	**36**	**0.86**	**69**
	Average	**0.03**	**26.6**	**0.37**	**24**	**Average**	**0.36**	**33.83**	**1.38**	**68.5**

ΔPSNR(+/-): picture quality loss/gain measured in dB
Δ T(+/-): encoding time saving/loss measured as a %
CompRed(+/-): complexity increase/decrease measured as a %

Table 8.3: Simulation Results of Enhancement Layer

Sequence	Enhancement Layer at QP_{BL}=32				
	QP_{EL}	Δ PSNR	ΔT %	Δ Rate%	Comp Red(%)
Foreman	8	0.75	19	2.91	65
	16	0.88	24	1.62	65
	24	1.07	32	0.88	65
	32	0.47	41	0.60	65
	40	0.76	45	2.93	65
	avg	**0.79**	**32.2**	**1.78**	**65**
Crew	8	0.82	20	0.86	62
	16	0.96	24	0.87	62
	24	0.85	29	0.74	62
	32	0.87	33	1.83	62
	40	0.76	39	1.35	62
	avg	**0.85**	**29**	**1.13**	**62**
City	8	0.35	19	1.56	65
	16	0.54	25	0.79	65
	24	0.49	29	1.82	65
	32	0.48	34	0.73	65
	40	0.79	44	3.06	65
	avg	**0.53**	**30.4**	**1.59**	**65**
Bus	8	0.36	18	1.66	66
	16	0.72	20	1.41	66
	24	0.77	29	2.18	66
	32	0.74	37	1.03	66
	40	0.50	43	3.20	66
	avg	**0.62**	**29.4**	**1.89**	**66**
Mobile	8	0.32	19	1.98	67
	16	0.32	24	0.45	67
	24	0.57	29	1.32	67
	32	0.56	33	1.63	67
	40	0.42	40	0.80	67
	avg	**0.44**	**29**	**1.23**	**67**
Football	8	0.97	19	1.20	67
	16	0.88	23	0.30	67
	24	0.91	28	1.80	67
	32	0.89	35	2.00	67
	40	0.77	43	2.30	67
	avg	**0.88**	**29.6**	**1.52**	**67**
	Average	**0.68**	**30**	**1.53**	**65.33**

ΔPSNR(+/-): picture quality loss/gain measured in dB
Δ T(+/-): encoding time saving/loss measured as a %
CompRed(+/-): complexity increase/decrease measured as a %

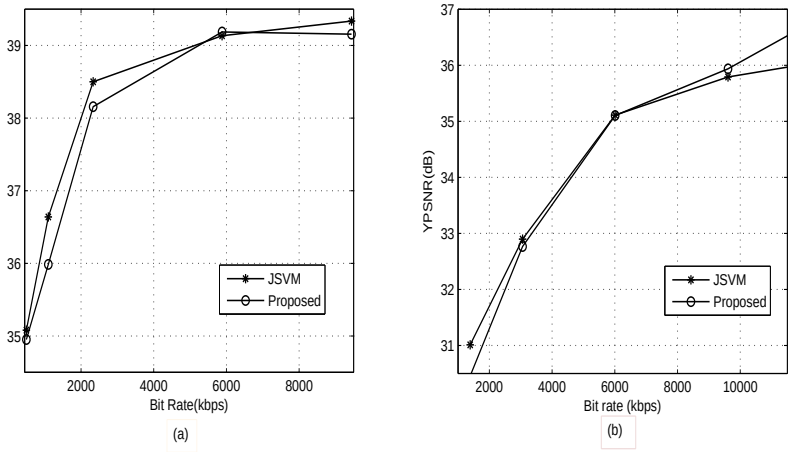


Figure 8.3: RD curve for (a) Crew and (b) Bus for Enhancement Layer

(a) Frame 1 of Crew at QP_{BL}=16 (b) Frame 1 of Crew at QP_{BL}=32

Figure 8.4: Frame 1 of CREW at different QP for BL and EL

8.4 Conclusions

In this chapter a scheme to reduce the complexity of intra prediction for the SVC standard was proposed. The correlation between the different layers was exploited for the mode decision in the fast encoding process. The base layer intra prediction is QP dependant. For the enhancement layer, the prediction mode for a MB is dependent on the final encoded mode in the collocated base layer MB. The results have shown that the reduction in complexity of the enhancement layer (68%) is more than that of the base layer (24%). The computational reduction for the base layer is modest.

9

Summary and Conclusions

Contents

9.1 Summary of the Work 158
9.2 Scope for the Future Work 163

H.264/AVC offers a significant improvement in the rate-distortion efficiency when compared with the existing standards such as MPEG-2 Video. This improved performance however comes at the expense of increased encoding complexity. The H.264/AVC standard gives a significant gain in the compression efficiency over previous standards due to different coding tools used in the standard. The use of the advanced coding tools and mode selection process eventually increases the computational complexity of the encoder. One such tool which increases the complexity is the variable block size prediction process. There are practical constraints in the implementation of the full rate distortion optimized mode selection process. The high complexity of the encoder becomes a bottleneck in real-time applications. This thesis investigates the overall complexity that arises from the encoding and decoding process in the H.264/AVC encoder. In this work, an attempt has been made to reduce the complexity and to speed up the encoding process. The reduction in the complexity has been investigated both in the spatial domain and in the transform domain.

This chapter summarizes the main contributions of this work. The algorithms and the experimental results are critically reviewed. The thesis explored various fast encoding methods for the H.264/AVC video coding standard. The main focus in the fast encoding techniques is the reduction in the number of searches performed to arrive at the best encoding mode. Some directions for future research in relation to the main findings are also indicated.

9.1 Summary of the Work

The aim of this work has been to develop novel algorithms for reducing the computational complexity of a H.264/AVC encoder. The main contributions of this research has been presented in five chapters.

We performed extensive intra prediction process for each MB for different test sequences. The resulting statistics were studied. It was observed that the probability of occurrence of different modes depend upon the value of the quantization parameter.

Homogeneous regions have more MBs encoded with larger blocks whereas those having complex textures have MBs encoded with smaller block sizes. From the full search ME process, we observed that the sequences with limited motion have more MBs encoded with larger block sizes and those with complex motion have more MBs encoded with smaller block sizes. A detailed description of the findings from the mode decision process has been described in this thesis.

Edge Histogram based Fast Mode Decision

Chapter 4 described a method for the fast intra and the fast inter prediction process based on the edge histogram amplitudes of the input image. The main contributions of this chapter are as follows:

(i) A detailed study of the exhaustive intra and inter prediction was first performed. The mode dependency of the MBs on the QP and on the directional features of the intra prediction process was observed. The relation between the edge histogram information in a MB and the corresponding mode selected for encoding was studied. The mode distribution of the MBs in a frame encoded as a P frame were studied in detail. Overall, a detailed study of the prediction process was carried out in this chapter.

(ii) For reducing the computational complexity of the intra prediction process, a new algorithm was proposed where the decision of the encoding mode was taken depending upon the pixel edge directions in the MB and the QP for the MB. Homogeneity of the MB was determined by finding the residual complexity in the MB after the prediction process.

(iii) A new algorithm for fast inter prediction was proposed. The decision on the selected mode was taken from the information of the frame difference image between consecutive frames. The SKIP mode decision was taken based on the pixel difference residues prior to performing the ME process. The homogeneity of the MB was

decided from the edge histogram amplitude for the MB. Use of smaller block size ME was considered based on the edge direction of the residue image.

(iv) Experimental results showed that the algorithms proposed in this chapter reduced the encoding time considerably with negligible reduction in the quality and marginal increase in the bitrate.

Early SKIP Mode Detection and Weighted Prediction for Fast Inter Prediction

Chapter 5 presented two algorithms for speeding up the inter mode decision process in the H.264/AVC encoder. Both the algorithms were based on the characteristics of the input video. The main contributions of the algorithms are given below:

(i) A detailed analysis of the SKIP mode decision process was first presented. The relation between the pixel differences between consecutive frames and the SKIP mode was studied. The conditions for the early detection of zero blocks in a MB were discussed. The first algorithm presented was a fast encoding algorithm which identifies the SKIP mode early from the values of the residuals of the difference image. Further check for the SKIP mode was done from early detection of the ZBs in a MB. The mode decision for the MBs that do not qualify for the SKIP mode was based on the number of ZBs in a MB. Experimental results have shown that the algorithm gives a good encoding time saving while maintaining similar PSNR and bitrate as that obtained from the JM reference software.

(ii) Each frame in a video has certain attributes that can be exploited for narrowing down the space of coding options. The second algorithm described a fast mode decision process where each MB in a frame is examined for its motion content and homogeneity. The predicted MV for the MB was obtained from the MVs of the neighboring MBs. The most probable mode in which the MB could be encoded depending on the modes of the neighboring MBs was determined. Each MB will have

different degrees of these attributes. Weights are assigned for these parameters and the final mode is selected based upon these weights. Good time saving is achieved by this method. It is interesting to note that the modes obtained from the proposed method and those obtained from the JM software were same or similar to a large extent.

Fast Mode Decisions in Transform Domain

Estimation of RD cost for each mode requires that the encoding bits should be available at the encoder. Entropy coding is performed for every mode so as to arrive at the best mode which gives the least RD cost. Thus transform coefficients are available at the encoder at no additional computational cost. Chapter 6 describes a method for the fast intra and inter prediction process based on a detailed study of these coefficients. The main contributions of this chapter are as follows:

(i) A detailed analysis of the transform coefficients after the transformation and quantization process on the residuals of the prediction process was first performed. It was noted that many transform coefficients for a MB are zero valued. As the QP increases, more coefficients become zero valued. The number of zero valued coefficients in a transformed block gives an indication of the possible mode for encoding a MB. A new algorithm was proposed for the fast mode selection process and reducing the computational complexity of the intra and the inter prediction process. From the observations on the number of zero-valued transform coefficients in a MB, a decision on the final encoding mode was taken. Results have shown the effectiveness of the algorithm.

(ii) The DCT exhibits excellent energy compaction for highly correlated images. The low frequency transform coefficients obtained after performing DCT contain most of the spatial detail. This property of the DCT was exploited and a novel algorithm for the inter prediction process was proposed. H.264/AVC uses 4×4 (subblock)

integer DCT transform. The first few low frequency coefficients for a subblock were calculated and a decision based on the values of these coefficients was taken on whether or not the block will be a ZB. Depending on whether or not the block is a ZB, the final decision on the encoding mode was taken. Both algorithms have shown substantial savings in the encoding time as compared to JM reference software.

Subband/DCT based Fast Mode Decision

The subband decomposition of images by passing it through a filter bank is used for image compression algorithms. The subband coding decomposes the image into a number of subbands. The low frequency subband retains the maximum information regarding the image. This idea was explored and a new encoding algorithm for the H.264/AVC was proposed in Chapter 7. This idea of subband coding is extended to the SVC standard and a fast encoding algorithm was proposed for the SVC standard. The main contributions of this chapter are as follows:

(i) Subband decomposition of the input image was performed and exhaustive encoding was done for the LL subband. Full search intra and inter prediction was performed on the LL subband. Information from the LL subband was used for the encoding of the higher subbands namely the LH, HL and HH subbands

(ii) For the SVC standard, the LL subband was used for the base layer encoding and was encoded in the H.264/AVC compliant bitstream. For the enhancement layers, the LH, HL and HH subbands were used. Information from the base layer was used for reducing the number of modes for encoding the enhancement layer

(iii) Experimental results showed that the proposed algorithms gives encoding time saving while maintaining coding performance similar to the JM12.4 reference software. However, the enhancement layer image when QP of base layer was low gave degradation in the quality.

Mode Decision for Intra only Spatial Scalable Video Coding

The different layers in the scalable video are highly correlated. In chapter 8, this correlation between the layers is exploited in the proposed intra mode decision algorithm for the SVC standard. The intra prediction in the base layer depends on the QP of the base layer. The prediction modes for the enhancement layer is based on the encoding mode of the collocated MB in the base layer. The I4MB, I8MB or the I16MB mode selection is based on the QP of the base layer. Mode information from the base layer is utilized for the mode selection in the enhancement layer. For enhancement layer, the direction pertaining to the mode in the collocated MB in base layer is used for reducing the number of modes. Experimental results have shown the effectiveness of the proposed algorithm.

9.2 Scope for the Future Work

There are a number of issues that require further investigation in order to enhance the performance of the existing low complexity H.264/AVC encoders. The focus of the thesis has been to reduce the complexity of the encoder. A few possible research directions are listed:

- The algorithms in these thesis have been tested with test video sequences that are divided into three classes depending upon the motion activity. Investigating the amount of motion contained in a video sequence would be an interesting aspect that can be looked into. The activity detection prior to the ME and then classifying the video would be a future direction of research.

- The thresholds that have been used in the implementations are based on various observations made from an exhaustive prediction process. Though different video sequences have different levels of complexity, the same thresholds have been used for all classes of sequences. It may be noted that the thresholds selected for the

experiments are on the conservative side. Improvements can be made on the existing algorithms by proper statistical modeling of the input video based on the different characteristics like texture information, motion content, resolution, QP, etc. for the development of adaptive thresholds. It is expected that use of thresholds adapted to the level of complexity of the input sequence should further enhance the performance.

- The bit rate control of the encoder has not been taken into consideration for the fast encoding modes. Multimedia devices like mobile devices have low processing power and low bandwidth, video applications should be available at a very low rate. Usually the quality degrades at low bitrate. One of the future direction of work would be to consider fast encoding algorithm with bitrate reduction while maintaining quality.

- High Efficiency Video Coding (HEVC) also called H.265 is a proposed video compression standard and a successor to H.264/MPEG-4 AVC (Advanced Video Coding), currently under joint development by the ISO/IEC Moving Picture Experts Group (MPEG) and ITU-T Video Coding Experts Group (VCEG). It is expected to improve the compression by a factor of two but at the expense of a large increase in the decoder and encoder complexity. The draft of the standard is expected to be finalized by mid 2012. It would be interesting to study the complexity of this encoder and ways to improve upon it.

Bibliography

[1] Y. W. Huang, B. Y. Hsieh, S. Y. Chien, S. Y. Ma, and L. G. Chen, "Analysis and Complexity Reduction of Multiple Reference Frames Motion Estimation in H.264/AVC," *IEEE Transaction on Circuits and System for Video Technology*, vol. 16, no. 4, pp. 507–522, April 2006.

[2] D. Marpe, T. Wiegand, and G. J. Sullivan, "The H.264/MPEG4 Advanced Video Coding Standard and its Applications," *IEEE Communications Magazine*, vol. 44, no. 8, pp. 134 – 143, August 2006.

[3] I. E. G. Richardson, "H.264 and MPEG-4 Video Compression: Video Coding for Next-generation Multimedia," *John Wiley and Sons, 2003.*

[4] "http://www.itu.int/ITU-T."

[5] "Mpeg website: http://www.mpeg.org."

[6] "ITU-T: H.261: Codec For Audiovisual Services At n 384 kbit/s (1988)."

[7] ISO/IEC JTC1/SC29, "Coding of moving pictures and associated audio for digital storage media at up to about 1.5 Mbit/s, ISO/IEC 11172-2, International Standard," November 1992.

[8] MPEG 2: ISO/IEC JTC1/SC29/WG11 and ITU-T, "ISO/IEC 13818-2, Generic Coding of Moving Pictures and Associated Audio Information: Video, ISO/IEC and ITU-T," 1994.

[9] MPEG 4: ISO/IEC JTC1/SC29/WG1, "ISO/IEC 14 496:2000-2: Information on Technology-Coding of Audio-Visual Objects-Part 2: Visual, ISO/IEC," 2000.

[10] H.264/AVC:, "Joint Video Team of ITU-T and ISO/IEC JTC 1, Draft ITU-T Recommendation and Final Draft International Standard of Joint Video Specification (ITU-T Rec. H.264 — ISO/IEC 14496-10 AVC), document JVT-G050r1." 2003.

[11] I. Richardson, "White Paper: An Overview of H.264 Advanced Video Coding," Vcodex, Tech. Rep., 2007. [Online]. Available: http://www.vcodex.com

[12] J.-B. Lee and H. Kalva, *The VC-1 and H.264 Video Compression Standards for Broadband Video Services*, B. Furht, Ed. Springer, 2008.

[13] T. Wiegand, G. J. Sullivan, G. Bjontegaard, and A. Luthra, "Overview of the H.264/AVC Video Coding Standard," *IEEE Transaction on Circuits and System for Video Technology*, vol. 13, no. 7, pp. 560–576, July 2003.

[14] J. Ostermann, J. Bormans, P. List, D. Marpe, M. Narroschke, F. Pereira, T. Stockhammer, and T. Wedi, "Video Coding with H.264/AVC: Tools, Performance and Complexity," *IEEE Circuits and Systems Magazine*, vol. 4, no. 1, pp. 7 – 28, 2004.

[15] G. J. Sullivan, P. Topiwala, and A. Luthra, "The H.264/AVC Advanced Video Coding Standard: Overview and Introduction To The Fidelity Range Extensions," SPIE Conference on Applications of Digital Image Processing XXVII Special Session on Advances in the New Emerging Standard: H.264/AVC, August, 2004.

[16] R. Schafer, T. Wiegand, and H. Schwarz, "The Emerging H.264/AVC Standard," *EBU Technical Review*, pp. 1–12, January 2003.

[17] H. Kalva, "The H.264 Video Coding Standard," *IEEE Transactions on Multimedia*, vol. 13, no. 4, pp. 86–90, 2006.

[18] D. Marpe, H. Schwarz, and T. Wiegand, "Context-Based Adaptive Binary Arithmetic Coding in the H.264/AVC Video Compression Standard," *IEEE Transaction on Circuits and System for Video Technology*, vol. 13, no. 7, pp. 620–636, July 2003.

[19] T. Wiegand, H. Schwarz, A. Joch, F. Kossentini, and G. J. Sullivan, "Rate-Constrained Coder Control and Comparison of Video Coding Standards," *IEEE Transaction on Circuits and System for Video Technology*, vol. 13, no. 7, pp. 688–703, July 2003.

[20] G. J. Sullivan and T. Wiegand, "Rate-Distortion Optimization for Video Compression," *IEEE Signal Processing Magazine*, pp. 74–90, November 1998.

[21] F. Pan, X. Lin, R. Susanto, K. P. Lim, Z. G. Li, G. N. Feng, D. J. Wu, and S. Wu, "Fast Mode Decision for Intra Prediction," *Joint Video Team (JVT) of ISO/IEC MPEG & ITU-T VCEG (ISO/IEC JTC1/SC29/WG11 and ITU-T SG16 Q.6)*, March 2003.

[22] F. Pan, X. Lin, S. Rahardja, K. P. Lim, Z. G. Li, D. Wu, and S. Wu, "Fast Mode Decision Algorithm for Intraprediction in H.264/AVC Video Coding," *IEEE Transaction on Circuits and System for Video Technology*, vol. 15, no. 7, pp. 813–823, July 2005.

[23] Z. Y. dong, D. Feng, and L. S. xun, "Fast 4x4 Intra Prediction Mode Selection for H.264," in *ICME*, vol. 2, June 2004, pp. 1151–1154.

[24] L. Qiong, H. Rui-min, Z. Li, Z. Xin-chen, and H. Zhen, "Improved Fast Intra Prediction Algorithm of H.264/AVC," *Journal of Zhejiang University SCIENCE A*, pp. 101–105, 2006.

[25] J.-F. Wang, J.-C. Wang, J.-T. Chen, A.-C. Tsai, and A. Paul, "A Novel Fast Algorithm for Intra Mode Decision in H.264/AVC Encoders," in *ISCAS*, 2006, pp. 3498–3501.

[26] Wei-Guang, L. An-Chao, T. Jhing-Fa, and W. J.-F. Yang, "A Simple Direction Detection Algorithm for Fast H.264 Intra Prediction," in *TENCON 2007, IEEE Region 10 Conference*, 2007.

[27] J. Kim, D. Kim, and J. Jeong, "Complexity Reduction Algorithm for Intra Mode Selection in H.264/AVC Video Coding," in *ACIVS, LNCS*, vol. 4179, 2006, pp. 454–465.

[28] J. Kim, K. Jeon, and J. Jeong, "H.264 Intra Mode Decision for Reducing Complexity Using Directional Masks and Neighboring Modes," *PSIVT*, vol. LNCS 4319, pp. 959–968, 2006.

[29] L.-J. Pan and Y.-S. Ho, "A Fast Mode Decision Algorithm For H.264/AVC Intra Prediction," in *SiPS*, 2007, pp. 704–709.

[30] G. Lee, H. Lin, and M. Wang, "Rate Control Algorithm Based On Intra-picture Complexity for H.264/AVC," *IET Image Processing*, vol. 3, pp. 26–39, 2008.

[31] A. Elyousfi, A. Tamtaoui, and E. Bouyakhf, "A New Fast Intra Prediction Mode Decision Algorithm for H.264/AVC Encoders," *World Academy of Science, Engineering and Technology*, vol. 27, pp. 1–7, 2007.

[32] C.-H. Tseng, H.-M. Wang, and J.-F. Yang, "Enhanced Intra 4x4 Mode Decision for H.264/AVC Coders," *IEEE Transaction on Circuits and System for Video Technology*, vol. 16, no. 8, pp. 1027–1032, August 2006.

[33] Y. N. Sairam, N. Ma, and N. Sinha, "A Novel Partial Prediction Algorithm for Fast 4x4 Intra Prediction Mode Decision in H.264/AVC," in *Data Compression Conference*, 2008, pp. 232–241.

[34] J. Liu and Z. Guo, "Efficient Intra 4 x 4 Mode Decision Based on Bit-rate Estimation in H.264/AVC," in *ISCAS*, 2008, pp. 492 – 495.

[35] H.-C. Lin, W.-H. Peng, and H.-M. Hang, "A Fast Mode Decision Algorithm With Macroblock-Adaptive Rate-Distortion Estimation For Intra-Only Scalable Video Coding," in *ICME*, 2008, pp. 765–768.

[36] T. K. Tan, C. S. Boon, and Y. Suzuki, "Intra Prediction By Template Matching," in *ICIP*, 2006, pp. 1693–1696.

[37] I.-H. Cho, J.-H. Lee, W.-H. Lee, and D.-S. Jeong, "New Intra Luma Prediction Mode in H.264/AVC Using Collocated Weighted Chroma Pixel Value," in *ACIVS*, ser. LNCS, vol. 4179, 2006, pp. 344–353.

[38] C. Kim, H.-H. Shih, and C.-C. J. Kuo, "Fast H.264 Intra-prediction Mode Selection Using Joint Spatial and Transform Domain Features," *Journal of Visual Communication and Image Representation.*, vol. 17, p. 291310, 2005.

[39] J.-W. Chen, C.-H. Chang, C.-C. Lin, Y.-H. O. Yang, J.-I. Guo, and J.-S. Wang, "A Condition-based Intra Prediction Algorithm for H.264/AVC," in *ICME*, 2006, pp. 1077–1080.

[40] A. Robert, I. Amonou, and B. Pesquet-Popescu, "Improving Intra Mode Coding in H.264/AVC Through Block Oriented Transforms," in *IEEE 8th Workshop on Multimedia Signal Processing*, 2006, pp. 382 – 386.

[41] T. Tsukuba, I. Nagayoshi, T. Hanamura, and H. Tominaga, "H.264 Fast Intra-Prediction Mode Decision Based On Frequency Characteristic," in *Proc. of European Signal Processing Conf. EUSIPCO2005*, 2005.

[42] Z. Wang, J. Liu, and J. Tian, "H.264-Based Scalable Intra Coding Scheme," in *ISCP*, 2006.

[43] H. Li, Z. G. Li, C. Wenl, and L.-P. Chaul, "Fast Mode Decision for Spatial Scalable Video Coding," in *ISCAS*, 2006, pp. 3005–3008.

[44] C. Park, C. Park, and S. Ko, "Generalisation of Inter-layer Intra Prediction for Scalable Video Coding," *Electronics Letters*, vol. 44, no. 5, February 2008.

[45] W. Yang, G. Rath, and C. Guillemot, "Improved Interlayer Prediction for Scalable Video Coding," in *ICASSP*, ser. I, 2007, pp. 649–652.

[46] S.-T. Hsiang, "Intra-frame Dyadic Spatial Scalable Coding Based On A Subband/Wavelet Framework For MPEG-4 AVC/H.264 Scalable Video Coding," in *ICIP*, 2007, pp. 73–76.

[47] Z. Chen, P. Zhou, Y. He, and Y. Chen, "Fast Integer Pel and Fractional Pel Motion Estimation for JVT," *Joint Video Team (JVT) of ISO/IEC MPEG & ITU-T VCEG Document: JVT-F017*, 2002.

[48] B. Jeon and J. Lee, "Fast Mode Decision for H.264," *Joint Video Team (JVT) of ISO/IEC MPEG & ITU-T VCEG Document: JVT-J033*, 2003.

[49] T. G. We, Z. Z. Yang, Z. Y. Jun, and Z. W. Jun, "Fast Mode Decision for Inter Macroblocks in H.264/AVC," *Journal of Shanghai University*, vol. 10, no. 2, pp. 233–237, 2006.

[50] A. Ahmad, N. Khan, S. Masud, and M. A. Maud, "Selection Of Variable Block Sizes in H.264," in *ICASSP*, vol. 3, 2004, pp. 173–176.

[51] B.-D. Choi, J. H. Nam, M. C. Hwang, and S. J. Ko, "Fast Motion Estimation and Intermode Selection for H.264," *EURASIP Journal on Applied Signal Processing*, vol. 2006, no. 71643, pp. 1–8, 2006.

[52] I. Choi, J. Lee, and B. Jeon, "Fast Coding Mode Selection With Rate-Distortion Optimization for MPEG-4 Part-10 AVC/H.264," *IEEE Transaction on Circuits and System for Video Technology*, vol. 16, no. 12, pp. 1557–1561, December 2006.

[53] Y. Ding, Y. Si, and C. Yao, "A Novel Inter Mode Decision Algorithm For H.264/AVC," in *Congress on Image and Signal Processing*, 2008, pp. 334 – 338.

[54] T. D. H. Du, "Macroblock Mode Decision for H.264," in *MIR05*, ser. ACM, 2005, pp. 167–172.

[55] C. Gang, J. Zhen-hong, and C. He, "A Fast Quarter-pixel Motion Estimation Algorithm for H.264/AVC," *Optoelectronics Letters*, vol. 4, no. 1, pp. 65–67, January 2008.

[56] B. Girod, "Motion-Compensating Prediction with Fractional-Pel Accuracy," *IEEE Transactions On Communications*, vol. 41, no. 4, pp. 604–612, 1993.

[57] X. Jing and L. P. Chau, "Fast Approach for H.264 Inter Mode Decision," *Electronic Letters*, vol. 40, no. 17, pp. 1050–1052, 2004.

[58] Y.-H. Kam and W.-C. Siu, "A Fast Full Search Scheme for Rate-Distortion Optimization of Variable Block Size and Multi-frame Motion Estimation," in *49th IEEE International Midwest Symposium on Circuits and Systems.*, vol. 1, 2006, pp. 183 – 186.

[59] N. Khan, S. Masud, and A. Ahmad, "A Variable Block Size Motion Estimation Algorithm For Real-time H.264 Video Encoding," *Signal Processing: Image Communication*, vol. 21, pp. 306–315, 2006.

[60] D. Kim, J. Kim, and J. Jeong, "Adaptive Macroblock Mode Selection for Reducing the Encoder Complexity in H.264," in *ACIVS*, ser. LNCS, no. 4179, 2006, pp. 396 – 405.

[61] G. Y. Kim, Y. H. Moon, and J. H. Kim, "An Early Detection of All-Zero DCT Block in H.264," in *Int. Conf. Image Process (ICIP)*, vol. 1, October 2004, pp. 453–456.

[62] Y. H. Kim, J. W. Yoo, S. W. Lee, J. Shin, J. Paik, and H. K. Jung, "Adaptive Mode Decision for H.264 Encoder," *Electronic Letters*, vol. 40, no. 19, pp. 1172 – 1173, 2004.

[63] U. V. Koc and K. J. R. Liu, "DCT-Based Subpixel Motion Compensation And Fully DCT-Based Video Coder," in *ICIP*, vol. 3, 1997, pp. 598–601.

[64] C. H. Kuo, M. Shen, and C. C. J. Kuo, "Fast Inter-Prediction Mode Decision And Motion Search For H.264," in *ICME*, vol. 1, 2004, pp. 663 – 666.

[65] Y.-L. Lee, Y.-K. Lee, and H. Park, "A Fast Motion Vector Search Algorithm for Variable Blocks," in *ACIVS, LNCS*, vol. 4179, 2006, pp. 311–322.

[66] Y.-M. Lee and Y. Lin, "Zero-Block Mode Decision Algorithm for H.264/AVC," *IEEE Transaction on Circuits and System for Video Technology*, vol. 18, no. 3, pp. 524–533, March 2009.

[67] X. Li, "Least-Square Prediction for Backward Adaptive Video Coding," *EURASIP Journal on Applied Signal Processing*, vol. 2006, no. 90542, p. 113, 2006.

[68] Y.-C. Liaw, J. Z. CLai, and Z.-C. Hong, "Fast Block Matching Using Prediction and Rejection Criteria," *Elsevier Journal of Signal Processing*, vol. 89, no. 6, pp. 1115–1120, June 2009.

[69] D. K. Lim and Y. S. Ho, "A Fast Block-Matching Motion Estimation Algorithm with Motion Modeling and Motion Analysis," in *PCM, LNCS*, vol. 2532, 2002, pp. 135–142.

[70] H. Y. Lim and A. A. Kassim, "Fast Sub-pixel Motion Estimation for H.264," in *ACIVS*, ser. LNCS, vol. 4179, 2006, pp. 242–252.

[71] C.-F. Lin and J.-J. Leou, "An Adaptive Fast Full Search Motion Estimation Algorithm For H.264," in *ISCAS*, vol. 2, 2005, pp. 1493 – 1496.

[72] Z. Liu, L. Shen, and Z. Zhang, "An Efficient Intermode Decision Algorithm Based on Motion Homogeneity for H.264/AVC," *IEEE Transaction on Circuits and System for Video Technology*, vol. 19, no. 1, pp. 128–132, January 2009.

[73] K. K. Ma and G. Qiu, "An Improved Adaptive Rood Pattern Search For Fast Block-matching Motion Estimation in JVT/H.26L," in *IEEE International Symposium on Circuits and Systems (ISCAS)*, May 2003, pp. II–708–II–711.

[74] L. J. Pan and Y. S. Ho, "Fast Mode Decision Algorithm for H.264 Inter-prediction," *Electronic Letters*, vol. 43, no. 24, pp. 1351 – 1353, November 2007.

[75] I. Park and D. W. Capson, "Improved Inter Mode Decision Based on Residue in H.264/AVC," in *International Conference on Multimedia & Expo*, 2008, pp. 709–712.

[76] M. Paul, M. R. Frater, and J. F. Arnold, "An Efficient Mode Selection Prior to the Actual Encoding for H.264/AVC Encoder," *IEEE Transactions on Multimedia*, vol. 11, no. 4, pp. 581–588, June 2009.

[77] C. A. Rahman and W. Badawy, "A Quarter Pel Full Search Block Motion Estimation Architecture For H.264/AVC," in *ICME*, 2005.

[78] S.-H. Ri and J. Ostermann, "Fast Mode Decision for H.264/AVC Using Mode Prediction," in *ACIVS , LNCS*, vol. 4179, 2006, pp. 354–363.

[79] M. Sayed, W. Badawy, and G. Jullien, "Towards an H.264/AVC HW/SW Integrated Solution: An Efficient VBSME Architecture," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 55, no. 9, pp. 912 – 916, 2008.

[80] A. Chia, W. Yu, G. R. Martin, and H. Park, "Fast Inter-Mode Selection in the H.264/AVC Standard Using a Hierarchical Decision Process," *IEEE Transaction on Circuits and System for Video Technology*, vol. 18, no. 2, pp. 186–195, February 2008.

[81] L. Shen, Z. Liu, Z. Zhang, and X. Shi, "Fast Inter Mode Decisions Using Spatial Property of Motion Field," *IEEE Transaction on Multimedia*, vol. 10, no. 6, pp. 1208–1214, October 2008.

[82] L. A. Sousa, "General Method For Eliminating Redundant Computations in Video Coding," *Electronic Letters*, vol. 36, pp. 306–307, February 2000.

[83] Y. H. Moon, G. Y. Kim, and J. H. Kim, "An Improved Early Detection Algorithm for All-Zero Blocks in H.264 Video Encoding," *IEEE Transaction on Circuits and System for Video Technology*, vol. 15, no. 8, pp. 1053–1057, August 2005.

[84] T. Toivonen and J. Heikkila, "Efficient Method for Half-Pixel Block Motion Estimation Using Block Differentials," in *International Workshop VLBV*, ser. LNCS, vol. 2849, 2003, pp. 225–232.

[85] H. Wang, S. Kwong, and C.-W. Kok, "An Efficient Mode Decision Algorithm for H.264/AVC Encoding Optimization," *IEEE Transaction on Multimedia*, vol. 9, no. 4, pp. 882–888, June 2007.

[86] C.-C. Wu and J.-J. Leou, "A New Fast Full Search Motion Estimation Algorithm for H.264/AVC," in *IEEE Conference on Wireless Networks, Communications and Mobile Computing*, vol. 2, 2005, pp. 1249 – 1254.

[87] D. Wu, F. Pan, K. P. Lim, S. Wu, Z. G. Li, X. Lin, S. Rahardja, and C. C. Ko, "Fast Intermode Decision in H.264/AVC Video Coding," *IEEE Transaction on Circuits and System for Video Technology*, vol. 15, no. 6, pp. 953–958, July 2005.

[88] J. You, W. Kim, and J. Jeong, "16x16 Macroblock Partition Size Prediction for H.264 P Slices," *IEEE Transactions on Consumer Electronics*, vol. 52, no. 4, pp. 1377–1383, November 2006.

[89] H. Zeng, C. Cai, and K.-K. Ma, "Fast Mode Decision for H.264/AVC Based on Macroblock Motion Activity," *IEEE Transaction on Circuits and System for Video Technology*, vol. 19, no. 1, pp. 1–11, April 2009.

[90] M. Zhang, T. Zhou, and W. Wang, "Adaptive Method for Early Detecting Zero Quantized DCT Coefficients in H.264/AVC Video Encoding," *IEEE Transaction on Circuits and System for Video Technology*, vol. 19, no. 1, pp. 103–107, January 2009.

[91] S.-H.Kim and Y.S.Ho, "Fast Mode Decision Algorithm for H.264 Using Statistics of Rate-Distortion Cost," *Electronic Letters*, vol. 44, no. 14, July 2008.

[92] C.-M. Mak, C.-K. Fong, and W.-K. Cham, "Fast Motion Estimation for H.264/AVC in Walsh-Hadamard Domain," *IEEE Transaction on Circuits and System for Video Technology*, vol. 18, no. 6, pp. 735–745, June 2008.

[93] X. Xu and Y. He, "Improvements on Fast Motion Estimation Strategy for H.264/AVC," *IEEE Transaction on Circuits and System for Video Technology*, vol. 18, no. 3, pp. 285–293, March 2008.

[94] P.-J. Lee and Y.-J. Shih, "Fast Inter-Frame Coding with Intra Skip Strategy in H.264 Video Coding," *IEEE Transaction on Consumer Electronics*, vol. 55, no. 1, pp. 158–164, February 2009.

[95] C.C.Lin, Y.Lin, and H.J.Hsieh, "Multi-direction Search Algorithm for Block Motion Estimation in H.264/AVC," *IET Image Processing*, vol. 3, no. 2, pp. 88–99, 2009.

[96] Z. Liu, J. Z. S. Goto, and T. Ikenaga, "Motion Estimation Optimization for H.264/AVC Using Source Image Edge Features," *IEEE Transaction on Circuits and System for Video Technology*, vol. 18, no. 8, pp. 1095 – 1107, August 2009.

[97] Z. Xuan, Y. Zhenghua, and Y. Songyu, "Method For Detecting All-Zero DCT Coefficients Ahead Of Discrete Cosine Transformation and Quantisation," *Electronic Letters*, vol. 34, no. 19, pp. 1839–1840, September 1998.

[98] "http://iphome.hhi.de/suehring/tml/download/."

[99] G. Sullivan and G. Bjontegaard, "Recommended Simulation Common Conditions for H.26L Coding Efficiency Experiments on Low-Resolution Progressive-Scan Source Material," *ITU-T VCEG, Doc. VCEG-N81*, pp. 1–3, 2001.

[100] G. Bjontegaard, "Calculation of Average PSNR Difference Between RD-Curves," *ITU-T VCEG, Doc. VCEG-M33*, pp. 1–2, March 2001.

[101] A. Ahmad, N. Khan, S. Masud, and M. A. Maud, "Efficient Block Size Selection in H.264 Video Coding Standard," *Electronics Letters*, vol. 40, no. 1, pp. 19–21, January 2004.

[102] Y.-M. Lee, Y.-T. Sun, and Y. Lin, "SATD-Based Intra Mode Decision for H.264/AVC Video Coding," *IEEE Transaction on Circuits and System for Video Technology*, vol. 20, no. 3, pp. 463–469, March 2010.

[103] T. Wiegand, G. J. Sullivan, G. Bjontegaard, and A. Luthra, "Joint Draft 11 of SVC Amendment," *Joint Video Team (JVT)*, vol. Doc. JVT-X201, 2007.

[104] "JVT of ITU-T and ISO/IEC JTC 1, Draft ITU-T Recommendation and Final Draft International Standard of Joint Video Specification (ITU-T Rec. H.264 — ISO/IEC 14496-10 AVC)," *Document JVT-G050r1*, May 2003.

[105] H. Schwarz, D. Marpe, and T. Wiegand, "Overview of the Scalable Video Coding Extension of the H.264/AVC Standard," *IEEE Transaction on Circuits and System for Video Technology*, vol. 17, no. 9, pp. 1103–1120, September 2007.

[106] L. Xiong, "Reducing Enh. Layer Directional Intra Prediction Modes," *ISO/IEC JTC1/SC29/WG11 and ITU-T SG16 Q.6, JVT-P041*, 2005.

[107] L. Yang, Y. Chen, J. Zhai, and F. Zhang, "Low Complexity Intra Prediction for Enhancement Layer," *ISO/IEC JTC1/SC29/WG11 and ITU-T SG16 Q.6, JVT-Q084*, 2005.

[108] C. Segall and Gary.J.Sullivan, "Spatial Scalability within H.264 AVC Scalable Video Coding Extension," *IEEE Transaction on Circuits and System for Video Technology*, vol. 17, no. 9, pp. 1121–1135, September 2007.

[109] K. D. Wolf, D. D. Schrijver, S. D. Zatter, and R. V. de Walle, "Scalable Video Coding : Analysis and Coding Performance of Inter-Layer Prediction," in *ISSPA*, February 2007, pp. 1 – 4.

[110] G. Lu and L. Chen, "Fast Mode Decision for H.264 Based on DC Coefficient," in *Seventh International Conference on Information Technology*, 2010, pp. 382–386.

[111] M. Kim, S. Jung, C.-S. Kim, and S. Sull, "An Efficient Inter-frame Coding with Intra Skip Decision in H.264/AVC," *IEEE Transaction on Consumer Electronics*, vol. 56, no. 2, pp. 856 – 862, May 2010.

List of Publications

(i) Amrita Ganguly, Anil Mahanta; "Improved Mode Decision Algorithm Based on Residues and Early Zero Block Detection in H.264/AVC," *Journal of Signal, Image and Video Processing* Online in February, 2012, Doi:10.1007/S11760-012-021-7

(ii) Amrita Ganguly, Anil Mahanta; "New Fast Mode Decision Algorithm for Intra Only Spatial Scalable Video Coding," *Int. J. of Recent Trends in Engineering and Technology,* Vol. 4, No. 3, Nov 2010

(iii) Amrita Ganguly, Anil Mahanta; "Weight Based Fast Mode Decision For H.264/AVC Video Coding Standard," *selected for publication in "Lecture Notes" (CCIS) published by Springer-Verlag,*, February, 2012.

(iv) Amrita Ganguly, Anil Mahanta; "Fast Mode Decision Algorithm for H.264/AVC Intra Prediction" *IEEE, TENCON 2009,* Singapore, Nov. 2009.

(v) Amrita Ganguly, Anil Mahanta; "A Subband/DCT based Fast Mode Decision Algorithm for H.264" *IEEE, TENCON 2009,* Singapore, Nov. 2009.

(vi) Amrita Ganguly, Anil Mahanta; "Fast Mode Decision Algorithm for Intra only Scalable Video Coding using combined Subband/DCT Coding" *Proceedings of 2009 International Conference on Software Technology and Engineering*, pp. 141-145, July 2009.

(vii) Amrita Ganguly, Anil Mahanta," Fast Intra Prediction Algorithm in the Transform Domain for H.264/AVC", *IEEE International Conference on Advances on Communication, Network and Computing*, Oct. 2010.

(viii) Amrita Ganguly, Anil Mahanta," Fast Mode Decision Algorithm for H.264/AVC using Edge Characteristics of Residue Images", *Seventh Indian Conference on Computer Vision, Graphics and Image Processing*, Chennai, Dec. 2010.

(ix) Amrita Ganguly, Anil Mahanta,“ Improved Inter Mode Decision for H.264/AVC Using Weighted Prediction”, *International Conference on Signal Processing and Multimedia Applications, SIGMAP,* Spain, July 2011.

Under review

(i) Amrita Ganguly, Anil Mahanta; “Fast Mode Decision Algorithm for H.264/AVC Using Transform Coefficient Statistics,” Submitted to *Signal ,Image and Video Processing, Springer, Manuscript No.1717. Has undergone one favorable review.*

(ii) Amrita Ganguly, Anil Mahanta; “IMPROVED MODE DECISION BASED ON ZERO TRANSFORM COEFFICIENTS FOR H.264/AVC,” Submitted to *SADHANA (Springer), 2011, Manuscript No.SADH-D-10-00175.*