Copyright Undertaking

This thesis is protected by copyright, with all rights reserved.

By reading and using the thesis, the reader understands and agrees to the following terms:

1. The reader will abide by the rules and legal ordinances governing copyright regarding the use of the thesis.
2. The reader will use the thesis for the purpose of research or private study only and not for distribution or further reproduction or any other purpose.
3. The reader agrees to indemnify and hold the University harmless from and against any loss, damage, cost, liability or expenses arising from copyright infringement or unauthorized usage.

IMPORTANT

If you have reasons to believe that any materials in this thesis are deemed not suitable to be distributed in this form, or a copyright owner having difficulty with the material being included in our database, please contact lbsys@polyu.edu.hk providing details. The Library will look into your claim and consider taking remedial action upon receipt of the written requests.

Pao Yue-kong Library, The Hong Kong Polytechnic University, Hung Hom, Kowloon, Hong Kong

http://www.lib.polyu.edu.hk
Advanced Bitstream Switching Techniques for H.264 Video Streaming and Multi-View Video

by

Ki-Kit LAI

A thesis submitted in partial fulfillment of the requirements for the degree of Doctor of Philosophy

April 2011
CERTIFICATE OF ORIGINALLITY

I hereby declare that this thesis is my own work and that, to the best of my knowledge and belief, it reproduces no material previously published or written, nor material that has been accepted for the award of any other degree or diploma, except where due acknowledgement has been made in the text.

___________________________ (Signed)

________________________________ (Name of student)
Abstract

The goal of video streaming is to transmit pre-encoded videos from a server to a client over the Internet. Owing to the large bandwidth fluctuation, efficient adaptation to channel bandwidth is critical when transmitting video over various networks. The most straightforward way of solving the fluctuation problem is to represent each pre-encoded video sequence by using multiple and independent bitstreams of different bitrates and quality. An efficient switching mechanism among different bitstreams is then used to select an appropriate bitstream. Therefore, in this thesis, some novel techniques are suggested for bitstream switching with the minimum storage requirement and the reasonable complexity of the server.

To put it in the nutshell, performing switching at P-frames and I-frames induces error propagation and sacrifices the coding efficiency significantly, respectively. In H.264, the new frame type - Synchronization-Predictive frames (SP-frames) - is developed for multiple bitrate streaming with the support of seamless switching. Notwithstanding the guarantee of drift-free switching, the trade-offs are the bulkiness of SP-frames and degradation of quality. Our analysis reveals that the conventional pixel-domain motion estimation is not appropriate for encoding secondary SP-frames such that a significant amount of additional space or bandwidth for storage or transmission is induced. For this reason, we propose a new motion estimation and compensation technique, which is operated in the quantized transform (QDCT) domain instead of pixel-domain, for
Abstract
coding secondary SP-frames. In this technique, the encoding structure of secondary SP-frames is re-designed. Our proposed work keeps the secondary SP-frames as small as possible without affecting the size of primary SP-frames by incorporating QDCT-domain motion estimation and compensation in the secondary SP-frame coding. Simulation results demonstrate that the size of secondary SP-frames can be reduced remarkably.

In the meantime, multi-view video service has been attracting more and more attention recently. Different from the traditional single-view video, multi-view video brings in a brand new viewing experience with a high degree of user interactivity. It allows users to select their favourite viewpoint. It switches the bitstream at a particular view when necessary instead of transmitting all the views. The SP-frames, with capability of drift-free switching, can be directly employed in the low-delay viewpoint switching of multi-view videos. It means that QDCT-domain motion estimation can then be proposed to apply on the multi-view video scenarios. Experiments show that the weak interframe correlation in multi-view videos, comparing to multiple bitrate videos, can get benefit from our proposed scheme due to the fact that QDCT-domain motion estimation aims at minimizing residues rather than finding true motion vectors.

Although the new QDCT-domain scheme exhibits promising results for reducing the storage requirement of secondary SP-frames in multiple bitrate video and multi-view video systems, there are still some cases that pixel-domain motion estimation outperforms QDCT-domain motion estimation. To avoid this, a straightforward way is to perform both pixel-domain and QDCT-domain motion
estimation separately. The sizes of secondary SP-frames generated by both estimation techniques are then compared and the set of final motion vectors with a smaller bit-count is chosen. This approach demands huge computational complexity. In our work, we propose two hybrid motion estimation algorithms (frame-based and macroblock-based) to select an appropriate domain of motion estimation in secondary SP-frame coding. The selection is based on inter-frame correlation, which is measured using the bit-counts of its corresponding primary SP-frame. Results show that the proposed algorithm significantly decreases the size of secondary SP-frames and consumes much less time.

By employing QDCT-domain motion estimation, the work in this thesis shows outstanding improvements in terms of size and computational complexity for both multiple bitrate video and multi-view video applications. There is no doubt that the results of our work will certainly be useful for the future development of bitstream switching systems.
Acknowledgments

I would like to take this opportunity to express my sincere gratitude to my supervisor, Dr. Y.L. Chan, and co-supervisor, Professor W.C. Siu, for their continuous encouragement, guidance and care during the period that I worked on this thesis. They gave me information, suggestions and valuable advice contributing to every success of my research. More importantly, I am deeply impressed with their hard working style and their willingness to devote to the advancement of science and research. This gives me a clear image of a great researcher should be and have inspired me to work hard on the thesis. It is beyond doubt that this will continuously influence my future research and career.

I would again like to express my sincere thank to Dr. C.H. Fu, Mr. Bill Ip, Mr. S.H. Tsang, Miss Z.P. Deng and Miss T.K Lee. The sharing of ideas and experience with them has greatly contributed to make every success of my work. It has been a wonderful time to me in these years to work with them.

Meanwhile, I am glad to express my gratitude to the Department of Electronic and Information Engineering and the Centre for Signal Processing for providing me a comfortable working environment, and The Hong Kong Polytechnic University for their generous financial support to carry to my research work.
Above all, I am deeply grateful to my family for their constant love, encouragement and support. Without their understanding and patience, it is impossible for me to complete this research study.
# Table of Contents

CERTIFICATE OF ORIGINALITY ................................................................................. I
ABSTRACT ................................................................................................................ II
ACKNOWLEDGMENTS ............................................................................................... V
TABLE OF CONTENTS ............................................................................................. VII
LIST OF FIGURES .................................................................................................. IX
LIST OF TABLES ...................................................................................................... XI
ABBREVIATIONS .................................................................................................... XII
LIST OF PUBLICATIONS ....................................................................................... XV

  INTERNATIONAL JOURNAL PAPERS ................................................................. XV
  INTERNATIONAL CONFERENCE PAPERS ......................................................... XVI

CHAPTER 1  INTRODUCTION ............................................................................... 1
  1.1 OVERVIEW .................................................................................................. 1
  1.2 SCALABLE VIDEO CODING ..................................................................... 2
  1.3 BITSTREAM SWITCHING USING SP-FRAMES ....................................... 3
  1.4 FREE VIEWPOINT / MULTI-VIEW / 3D VIDEO .......................................... 5
  1.5 MOTIVATION AND OBJECTIVES .............................................................. 6
  1.6 ORGANIZATION OF THIS THESIS ............................................................. 8

CHAPTER 2  LITERATURE REVIEW .................................................................... 10
  2.1 INTRODUCTION .......................................................................................... 10
  2.2 VIDEO COMPRESSION FUNDAMENTALS ............................................... 11
    2.2.1 Video compression techniques .............................................................. 11
    2.2.2 H.264 video coding ............................................................................. 18
  2.3 CONVENTIONAL VIDEO SWITCHING METHOD ....................................... 21
  2.4 PROBLEM OF BITSTREAM SWITCHING USING P-FRAMES ..................... 23
  2.5 VIEWPOINT SWITCHING IN MULTI-VIEW VIDEOS ...................................... 24
  2.6 CHAPTER SUMMARY .................................................................................. 27

CHAPTER 3 BITSTREAM SWITCHING IN MULTIPLE BITRATE VIDEOS .................. 29
  3.1 INTRODUCTION .......................................................................................... 29
  3.2 SP-FRAME ENCODING IN H.264 ............................................................... 30
  3.3 PROBLEM OF USING PIXEL-DOMAIN MOTION-COMPENSATED PREDICTION IN SECONDARY SP-FRAME ENCODING ...................................................... 35
  3.4 PROPOSED SCHEME FOR SECONDARY SP-FRAME ENCODING ............ 37
  3.5 EXPERIMENTAL RESULTS ...................................................................... 41
  3.6 CONCLUSION ............................................................................................... 44

CHAPTER 4 VIEWPOINT SWITCHING IN MULTI-VIEW VIDEOS ............................ 46
  4.1 INTRODUCTION .......................................................................................... 46
  4.2 MOTIVATION OF USING SP-FRAMES IN MULTI-VIEW VIDEOS .............. 47
  4.3 DEFICIENCY OF CONVENTIONAL SP-FRAMES IN MULTI-VIEW VIDEOS 48
  4.4 PROPOSED QDCT-ME FOR SECONDARY SP-FRAMES IN MULTI-VIEW VIDEOS 50
  4.5 EXPERIMENTAL RESULTS ...................................................................... 52
    4.5.1 Performance of QDCT-domain technique in secondary SP-frame ........ 54
    4.5.2 Storage requirement of secondary SP-frames in multi-view video coding system 65
  4.6 CHAPTER SUMMARY .................................................................................. 73
# Table of Contents

## CHAPTER 5 FRAME-BASED DOMAIN SELECTION IN MOTION ESTIMATION OF CODING SECONDARY SP-FRAMES

5.1 Introduction ................................................................. 75  
5.2 Impact of using QDCT-ME on secondary SP-frame coding .................................................... 76  
5.3 Proposed hybrid motion estimation techniques........................................................................... 78  
5.4 Experimental results of using hybrid motion estimation algorithm ........................................... 81  
5.5 Chapter summary .................................................................................. 84

## CHAPTER 6 MACROBLOCK-BASED MOTION ESTIMATION SELECTION ALGORITHM IN BITSTREAM SWITCHING USING FLEXIBLE MACROBLOCK ORDERING

6.1 Introduction .................................................................................................................. 85  
6.2 Proposed scheme using Flexible Macroblock Ordering .......................................................... 86  
   6.2.1 MB-based hybrid motion estimation scheme ...................................................................... 86  
   6.2.2 Utilization of FMO in the proposed hybrid Scheme ........................................................... 89  
6.3 Experimental results ......................................................................................................... 93  
6.4 Chapter summary .............................................................................................................. 102

## CHAPTER 7 CONCLUSIONS & SUGGESTIONS FOR FUTURE RESEARCH

7.1 Contributions of the thesis ............................................................................................. 104  
7.2 Future work ..................................................................................................................... 107  
   7.2.1 Further saving of computational complexity of QDCT-ME ............................................. 107  
   7.2.2 Entropy coding for secondary SP-frames ......................................................................... 109  
   7.2.3 MVC to AVC transcoding ................................................................................................. 109

## REFERENCES

............................................................................................................................................. 112
List of Figures

Figure 2.1. Block diagrams of the generic video codec: (a) encoder and (b) decoder...13
Figure 2.2. Zig-zag scanning.......................................................................................... 16
Figure 2.3. H.264 Baseline, Main and Extended profiles.............................................. 20
Figure 2.4. Conventional video streaming system........................................................... 22
Figure 2.5. Switching between two bitstreams B1 and B2............................................. 23
Figure 2.6. MVC prediction structure based on H.264 hierarchical B pictures............. 25

Figure 3.1. Switching bitstream from B1 to B2 using SP-frames.................................... 30
Figure 3.2. Simplified encoding block diagram of primary and secondary SP-frames [6][14] ..................................................................................................................... 31
Figure 3.3. Various modes and submodes in H.264. ...................................................... 32
Figure 3.4. Motion estimation and compensation of a secondary SP-frame encoder [77].... 34
Figure 3.5. Motion-compensated prediction using pixel-domain motion estimation in encoding a P-frame ........................................................................................................ 35
Figure 3.6. Motion-compensated prediction using pixel-domain estimation in encoding a secondary SP-frame............................................................................................. 36
Figure 3.7. QDCT-domain motion estimation on (a) a 4x4 sub-block, and (b) an 8x8 sub-block .................................................................................................................................. 39
Figure 3.8. The proposed secondary SP-frame encoder in the QDCT domain.............. 41
Figure 3.9. Size reduction of secondary SP-frames in percentage difference achieved by the proposed scheme over the scheme in [77], (a) Foreman, (b) Salesman, and (c) Table Tennis................................................................. 43

Figure 4.1. Switching bitstream from V1 to V2 using SP-frames..................................... 48
Figure 4.2. Energy distribution of quantized-transform coefficients in the residue for the “Ballroom” sequence in MBV: (a) the scheme in [77], (b) the proposed scheme, and (c) their difference........................................................................................................ 51
Figure 4.3. Energy distribution of quantized-transform coefficients in the residue for the “Ballroom” sequence in MVV: (a) the scheme in [77], (b) the proposed scheme, and (c) their difference........................................................................................................ 51
Figure 4.4. Frame-by-frame comparisons of size reduction of secondary SP-frames in percentage achieved by the proposed scheme over the scheme in [77] at QP = 28, (a) “Ballroom”, (b) “Exit”, (c) “Vassar”, (d) “Race1”, and (e) “Breakdancers”...55
Figure 4.5. (a) Reference frame in view 1 of “Ballroom”. Bit-count comparison in macroblock level between the proposed scheme and the scheme in [77] in (b) MVV, and (c) MBV. Macroblocs filled with white colour and black colour, which represent our proposed scheme have superior and inferior performances as compared with the scheme in [77], respectively.......................................................... 57
Figure 4.6. Size reduction of secondary SP-frames in percentage achieved by the proposed scheme over the scheme in [77] with different Qp, (a) “Ballroom”, (b) “Exit”, (c) “Vassar”, (d) “Race1”, and (e) “Breakdancers”................................. 59
Figure 4.7. Size reduction of secondary SP-frames in percentage achieved by the proposed scheme over the scheme in [77] for “Ballroom” in MBV and MVV..... 61
Figure 4.8. Frame-by-frame comparisons of size reduction of secondary SP-frames in percentage achieved by the proposed scheme over the scheme in [77] in (a) “Riverbed” and (b) “Crew”. .......................................................... 63
Figure 4.9. Energy distribution of quantized-transform coefficients in the residue for (a) first view of “Ballroom”, (b) “Riverbed”, and (c) “Crew”. .......................... 64
Figure 4.10. Multi-view Video Coding System. .......................................................... 65
Figure 4.11. Rate-distortion performance with periodic I-frame and SP-frame insertion for (a) “Ballroom”, (b) “Exit”, (c) “Vassar”, (d) “Race1”, and (e) “Breakdancers”. ................................................................................................................................. 68

Figure 5.1 Frame-by-frame comparisons of size reduction of secondary SP-frames in percentage achieved by QDCT-ME over pixel-ME in the "Crew" sequence. .... 77
Figure 5.2. Bitstream witching using primary and secondary SP-frames ....................... 78
Figure 5.3. Sizes of secondary SP-frames and the corresponding primary SP-frames of “Mobisode2”. .................................................................................. 79
Figure 5.4. Sizes of secondary SP-frames by applying the pixel-ME, QDCT-ME and proposed hybrid algorithms with Qp=24 to Qp=20 switching in (a) “Crew”, (b) “Mobisode1”, (c) “Mobisode2”. .............................................. 82

Figure 6.1. Switching bitstream from B1 to B2 using SP-frames...................................... 87
Figure 6.2. Bit-counts for all macroblocks in two rows of a secondary SP-frame and its corresponding primary SP-frame in the “Crew” sequence. ......................... 88
Figure 6.3. FMO map types. ........................................................................................ 91
Figure 6.4. Frame-by-frame size reduction of secondary SP-frames in percentage achieved by the proposed MB-based scheme and QDCT-ME over pixel-ME for the switching-down scenario. (a) Crew, (b) Riverbed, (c) Duckstageoff, and (d) Shuttlestart. ............................................................................................................. 94
Figure 6.5. Frame-by-frame size reduction of secondary SP-frames in percentage achieved by the proposed MB-based scheme and QDCT-ME over pixel-ME for the switching-up scenario. (a) Crew, (b) Riverbed, (c) Duckstageoff, and (d) Shuttlestart. ................................................................. 95
Figure 6.6. Size reduction of secondary SP-frames in percentage achieved by the proposed MB-based scheme and QDCT-ME over pixel-ME with different Qp for the switching-down scenario. (a) Crew, (b) Riverbed, (c) Duckstageoff, and (d) Shuttlestart. ............................................................................................................. 97
Figure 6.7. Size reduction of secondary SP-frames in percentage achieved by FMO and QDCT-ME over pixel-ME with different Qp for the switching-up scenario. (a) Crew, (b) Riverbed, (c) Duckstageoff, and (d) Shuttlestart. .......................... 98
Figure 6.8. Different areas using QDCT-ME and Pixel-ME. ........................................ 99

Figure 7.1. KS_IBP structure ...................................................................................... 110
Figure 7.2. Possible directions for extraction of target view using SI and SB frames. 111
List of Tables

Table 4.1. Average bit-counts of secondary SP-frames in MVV and MBV. ...............49
Table 4.2. The percentages of average size reduction of secondary SP-frames of our
proposed scheme over the scheme in [77] in MBV with various skipping factors.62
Table 4.3. Average bit-counts of different secondary SP-frames for “Ballroom” by
applying the scheme in [77] and the proposed scheme.................................71
Table 4.4. Average bit-counts of different secondary SP-frames for “Exit” by applying
the scheme in [77] and the proposed scheme............................................71
Table 4.5. Average bit-counts of different secondary SP-frames for “Vassar” by
applying the scheme in [77] and the proposed scheme..............................72
Table 4.6. Average bit-counts of different secondary SP-frames for “Race1” by
applying the scheme in [77] and the proposed scheme...............................72
Table 4.7. Average bit-counts of different secondary SP-frames for “Breakdancers” by
applying the scheme in [77] and the proposed scheme...............................73

Table 5.1. Performance comparison between the QDCT-ME and proposed hybrid
algorithms on different video sequences..................................................83

Table 6.1. Average time usage of secondary SP-frames with different $Qp$ for the
switching-down scenario.............................................................................101
Table 6.2. Average time usage of secondary SP-frames with different $Qp$ for the
switching-up scenario. ............................................................................101
Table 6.3. Comparison between frame-based and MB-based hybrid schemes on
switching-up scenario ($Qp = 24$ to $Qp = 20$)............................................102
<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>2D</td>
<td>2-dimensional</td>
</tr>
<tr>
<td>3D</td>
<td>3-dimensional</td>
</tr>
<tr>
<td>AC</td>
<td>Alternate Current</td>
</tr>
<tr>
<td>AV</td>
<td>Audio Visual</td>
</tr>
<tr>
<td>AVC</td>
<td>Advanced Video Coding</td>
</tr>
<tr>
<td>B-frame</td>
<td>Bi-directional Predicted Frame</td>
</tr>
<tr>
<td>CABAC</td>
<td>Context-based Arithmetic Coding</td>
</tr>
<tr>
<td>CAVLC</td>
<td>Context-based Variable Length Coding</td>
</tr>
<tr>
<td>dB</td>
<td>Decibel</td>
</tr>
<tr>
<td>DC</td>
<td>Direct Current</td>
</tr>
<tr>
<td>DCT</td>
<td>Discrete Cosine Transform</td>
</tr>
<tr>
<td>DVD</td>
<td>Digital Versatile Disc</td>
</tr>
<tr>
<td>FGS</td>
<td>Fine Granularity Scalable</td>
</tr>
<tr>
<td>FMO</td>
<td>Flexible Macroblock Ordering</td>
</tr>
<tr>
<td>FPS</td>
<td>Frames per Second</td>
</tr>
<tr>
<td>FVV</td>
<td>Free Viewpoint Video</td>
</tr>
<tr>
<td>GOP</td>
<td>Group of Pictures</td>
</tr>
<tr>
<td>GOV</td>
<td>Group of Views</td>
</tr>
<tr>
<td>HD</td>
<td>High Definition</td>
</tr>
<tr>
<td>IET</td>
<td>Institution of Engineering and Technology</td>
</tr>
<tr>
<td>I-frame</td>
<td>Intra-coded Frame</td>
</tr>
<tr>
<td>ITU-T</td>
<td>International Telecommunication Union</td>
</tr>
<tr>
<td>Abbreviation</td>
<td>Description</td>
</tr>
<tr>
<td>--------------</td>
<td>-------------</td>
</tr>
<tr>
<td>JM</td>
<td>Joint Model</td>
</tr>
<tr>
<td>J_{motion}</td>
<td>Joint R-D function for motion estimation</td>
</tr>
<tr>
<td>J_{mode}</td>
<td>Joint R-D function for mode decision</td>
</tr>
<tr>
<td>JMVM</td>
<td>Joint Multi-view Video Model</td>
</tr>
<tr>
<td>JVT</td>
<td>Joint Video Team</td>
</tr>
<tr>
<td>KDDI</td>
<td>Kokusai Denshin Denwa Inc</td>
</tr>
<tr>
<td>MB</td>
<td>Macroblock</td>
</tr>
<tr>
<td>MBAmap</td>
<td>Macroblock Allocation map</td>
</tr>
<tr>
<td>MBV</td>
<td>Multiple Bitrate Video</td>
</tr>
<tr>
<td>MC</td>
<td>Motion Compensation</td>
</tr>
<tr>
<td>ME</td>
<td>Motion Estimation</td>
</tr>
<tr>
<td>MERL</td>
<td>Mitsubishi Electric Research Laboratories</td>
</tr>
<tr>
<td>MPEG</td>
<td>Moving Picture Experts Group</td>
</tr>
<tr>
<td>MV</td>
<td>Motion Vector</td>
</tr>
<tr>
<td>MVC</td>
<td>Multi-view Video Coding</td>
</tr>
<tr>
<td>MVV</td>
<td>Multi-view Video</td>
</tr>
<tr>
<td>NAL</td>
<td>Network Abstraction Layer</td>
</tr>
<tr>
<td>P-frame</td>
<td>Forward Predicted Frame</td>
</tr>
<tr>
<td>PPS</td>
<td>Picture Parameter Set</td>
</tr>
<tr>
<td>PSP-frame</td>
<td>Primary SP-frame</td>
</tr>
<tr>
<td>QDCT</td>
<td>Quantized Discrete Cosine Transform</td>
</tr>
<tr>
<td>Qp</td>
<td>Quantization Parameter</td>
</tr>
<tr>
<td>Qs</td>
<td>Quantization Parameter for switching purpose</td>
</tr>
<tr>
<td>RBSP</td>
<td>Raw Byte Sequence Payload</td>
</tr>
<tr>
<td>Abbreviation</td>
<td>Description</td>
</tr>
<tr>
<td>--------------</td>
<td>---------------------------------------</td>
</tr>
<tr>
<td>RD</td>
<td>Rate Distortion</td>
</tr>
<tr>
<td>RTSP</td>
<td>Real Time Streaming Protocol</td>
</tr>
<tr>
<td>SAD</td>
<td>Sum of Absolute Difference</td>
</tr>
<tr>
<td>SAQTD</td>
<td>Sum of Absolute Quantized Transform Difference</td>
</tr>
<tr>
<td>SB-frame</td>
<td>Synchronization Bi-directional Frame</td>
</tr>
<tr>
<td>SI-frame</td>
<td>Synchronization Intra-coded Frame</td>
</tr>
<tr>
<td>SP-frame</td>
<td>Synchronization Predictive Frame</td>
</tr>
<tr>
<td>SSD</td>
<td>Sum of Square Difference</td>
</tr>
<tr>
<td>SSP-frame</td>
<td>Secondary SP-frame</td>
</tr>
<tr>
<td>TV</td>
<td>Television</td>
</tr>
<tr>
<td>UHD</td>
<td>Ultra High Definition</td>
</tr>
<tr>
<td>VCD</td>
<td>Video Compact Disc</td>
</tr>
<tr>
<td>VCL</td>
<td>Video Coding Layer</td>
</tr>
<tr>
<td>VCR</td>
<td>Video Cassette Recorder</td>
</tr>
<tr>
<td>VLC</td>
<td>Variable Length Coding</td>
</tr>
</tbody>
</table>
List of Publications

International Journal Papers


International Conference Papers


Chapter 1 Introduction

1.1 Overview

Recent advances in video streaming technologies and video compression standards such as MPEG-1, MPEG-2, MPEG-4 and H.264 are fuelling the emergence of a wide range of devices and applications that exploit the compressed-domain representation of digital videos. It facilitates visual data transmission through the Internet, contributes to the advent of digital broadcast systems, and makes possible the storage of digital video on DVD, Blu-ray disc, etc. Owing to the diversity of multimedia devices and applications, a video streaming server should have the capability of providing a wide variety of clients’ needs. In order to fulfill the requirements for various clients, an efficient organization of stored videos provides flexibility for a structure conversion under various conditions. The major challenge for a video streaming system is that the Internet is instinctively heterogeneous. Due to the variations of network conditions, the video server scales the bitrate of the compressed video and transmits it to the receiver. This service can be achieved by adjusting encoding parameters in real-time. However, this is not a typical way. A practical solution is to pre-encode videos by using multiple and independent bitstreams of different bitrates and quality. The server can then switches among bitstreams according to the network bandwidth. To comply the above scenario, some efficient methods for switching among different bitstreams are of great importance.
In this chapter, the fundamental principles of scalable video coding and the motives for bitstream switching are introduced. An overview of bitstream switching using SP-frames is then presented. Nowadays, multi-view videos or 3D videos are becoming very popular. We therefore discuss the problem of multi-view video coding standards. Finally, the motivation, objectives, and the organization of this thesis are presented.

1.2 Scalable video coding

Scalable coding is very successful for video streaming with channel bandwidth variation. It allows users to ubiquitously access and retrieve various video contents over heterogeneous networks with different display resolutions. For example, in the Internet streaming video applications, a server always has to serve a large amount of users with heterogeneous display resolutions and network bandwidth. When the users’ display resolution is small or the bandwidth between the server and some users is not enough to support higher resolution sequences, scalable coding [1-2] can provide different resolutions and bit rates to accommodate various users. In general, different scalable coding schemes adopted in H.263, MPEG-2, and MPEG-4 divide a video sequence into two layers – the base and enhancement layers. The video sequence is encoded at low quality or resolution to form the base layer, while the enhancement layer is encoded by computing the difference between the original image and the reconstructed image of the base layer. This arrangement of stored video is capable of gracefully coping with the users with
heterogeneous devices or the bandwidth fluctuations of the network. In the past
decade, many researchers are working on the problem of scalable coding to handle the heterogeneity of the receiver and network.

Among numerous scalable coding techniques, MPEG-4 fine granularity scalable (FGS) coding has become prominent due to its fine-grain scalability [3-4]. In MPEG-4 FGS, a video sequence is divided into the base and enhancement layers. The base layer is coded with non-scalable coding techniques, while the enhancement layer is coded according to the difference between the original image and the reconstructed image of base layer using bit-plane coding. By making this arrangement, it can be truncated flexibly to adapt to channel bandwidth variation. Since the enhancement layer can be truncated arbitrarily in any frame, MPEG-4 FGS provides a nice capability in readily and precisely adapting to channel bandwidth variation. However, its motion prediction is always based on the lowest quality base layer. As a result, low coding efficiency is the major disadvantage that prevents MPEG-4 FGS from being widely deployed in video streaming applications under a wide range of bandwidth variations.

1.3 Bitstream switching using SP-frames

The benefits of scalable video coding come at the cost of poorer coding efficiency, especially when the supported bit-rate range is very large. Therefore, for transmitting video over a network which has large bandwidth fluctuation, another approach is to independently encode video into several bitstreams with
different bitrates. By dynamically switching among these bitstreams, adaptation to channel bandwidth variation can be achieved. In the present video coding standards, drift-free switching between bitstreams is only possible at the position where the current and subsequent frames do not use any information previous to the current switching location, i.e. at I-frame. Seamless switching among multiple bitstreams at P-frames has also been investigated recently [5 - 7]. One simple way to solve such a problem is to losslessly compress the difference between two reconstructed references into an extra bitstream at the switching points [8 -9]. When switching is requested, the extra bitstream must be sent to the client and unfortunately, it also brings huge and often unacceptable additional overhead bits. In the state-of-art H.264 standard, a technique which supports drift-free switching at predictive frames is provided. At the switching point, a new picture type – Synchronization-Predictive frame (SP-frame) is adopted for seamless switching among bitstreams with different bitrates.

Like the normal P-frame, an SP-frame exploits temporal redundancy using predictive coding while allowing identical reconstruction of the frame even different reference frames are used. Nonetheless, there are some problems on the H.264 SP-encoding scheme which prevent it from further improved performance, especially on coding efficiency, and forbid it to be adopted in the existing commercial streaming server. In our investigation, the quality degradation of an SP-frame is over 1dB as compared with that of a P-frame. This fact implies the normal playback quality without bitstream switching will greatly be affected.
1.4 Free viewpoint / Multi-View / 3D video

3D video and free viewpoint video expand the user’s sensation far beyond what traditional media can offer. 3D video offers a 3D depth impression of the observed scenery while free viewpoint video (FVV) allows interactive selection of viewpoint and direction within a certain operating range. FVV targets at real world scenes as captured by real cameras with interactive free navigation. Some systems have already been using for sports and movies such as EyeVision.

To support this, some approaches have been designed to encode multiple video sources capturing from different viewpoints [10]. It requires tremendous amount of data for both storage and transmission. The direct way for encoding multi-view video is to encode all video signals independently using a H.264/MPEG4-AVC codec [11], which is usually called Simulcast. This method yields the highest bit rate for a 3D video compared to the other solutions since it does not consider the inter-view dependency and the temporal redundancy [10]. Currently, the Joint Video Team (JVT) has been developing a Multi-view Video coding (MVC) [12] which is based on the ITU-T H.264 video codec [13]. It differs from simulcast coding as it allows referencing between two views with a hierarchical B structure [11]. In most cases, MVC outperforms simulcast coding, yet requires more computation as expected. Nonetheless, random access, error robustness and synchronization are not really possible in the hierarchical B structure of MVC because all the frames in the Group of Pictures (GOP) and Group of Views (GOV) need to be decoded.
Further investigation is required to deal with the problems of random access and coding efficiency. In this research, the concept of SP-frames can be applied to facilitate the multiple viewpoint switching in FVV.

1.5 Motivation and objectives

Video streaming aims at transmitting stored videos from a server to a client over various networks. With the rapid developments in computing devices and networks, more and more users expect to enjoy video services through various devices over the Internet or wireless networks. Nevertheless, this kind of ubiquitous video services posts great challenges to the conventional video coding techniques. For instance, the Internet is instinctively heterogeneous and causes fluctuation of the effective bandwidth available to a user due to the changing network conditions. To gracefully cope with these variations, the server should provide different video bitrates according to the available channel bandwidth. In case of conversational services that are characterized by real-time encoding and point-to-point delivery, such situation can be achieved by adjusting, on the fly, source encoding parameters such as a quantization parameter or a frame rate based on the network feedback [14]. However, in typical streaming scenarios of sending an already encoded video bitstream to a client, the above solution cannot be applied. A straightforward solution of achieving bandwidth scalability is to compress the same video sequence into multiple and independent bitstreams of different bitrates and quality. Each source video is encoded into multiple non-scalable bitstreams with a preselected bitrate. An appropriate bitstream is selected at each time-slot.
according to the network feedback. The server then dynamically switches among the streams to accommodate the bandwidth variations. Furthermore, rebuffering due to congestion can be avoided to produce the best content quality and experience for the users. To conform with the aforementioned circumstance, some efficient methods for switching among different bitstreams are in great demand.

The rapid advancement of video coding techniques has enabled the expansion of video technology into new dimensions. Nowadays, we are not only looking into purely two-dimensional (2D) video technology, but also are expanding into three dimensional (3D) videos. In the past few years, the MPEG committee led the 3D audio-visual (3DAV) activity to explore the emergence of these new technologies. Potential applications include free-viewpoint video (FVV) or free-viewpoint television (FTV), 3D television (3DTV), immersive teleconference and surveillance. To support these applications, a video system requires capturing a scene from different viewpoints which results in generating several video sequences from different cameras. For multi-view video, it can give users the opportunity to choose their favourite viewpoints freely [15]. To accommodate seamless switching, we are also necessary to explore the ways for viewpoint switching in these new applications.

In order to switch among pre-encoded bitstreams with different bitrates and different viewpoints, the purpose of this research is to facilitate efficient bitstream switching algorithms with seamless reconstruction using SP-frames. In this thesis, we perform a detailed analysis for adopting SP-frames in
Chapter 1 Introduction

bitstream switching. Results of our investigation indicate that this arrangement is still primitive, and there is plenty of room for improvement. In our study, some novel algorithms are integrated into SP-frames, particularly to reduce the size requirement and minimize the coding complexity.

1.6 Organization of this thesis

This thesis is divided into seven chapters. Prior to our description of the main research in this thesis, a review of bitstream switching methods is given in Chapter 2. This review discusses the impact of implementing bitstream switching with different bitrates in the video streaming system. Afterwards, a review of multiple viewpoint switching is presented. Of course, this chapter begins with a brief description of some aspects of the video compression techniques that are relevant to this work.

In Chapter 3, some techniques of performing bitstream switching on H.264 video are presented. The drawback of using SP-frames into multiple bitrate switching is described. It includes the bulky size of secondary SP-frames. To reduce the size of SP-frames, a new motion estimation algorithm implemented in quantized-transform domain is proposed.

Chapter 4 mainly investigates a practical system of multiple viewpoint videos. The use of SP-frames illustrate that seamless switching can also be performed in multi-view videos. However, the size of SP-frames is much larger in this scenario. By adopting the proposed quantized-transform domain motion
estimation in SP-frame coding, significant reduction on the size of SP-frames in multi-view videos can be achieved. It also shows that the improvement can be found in multiple bitrate videos with complex motion.

The new motion estimation technique increases greatly the computational complexity. Besides, there are some conditions that the traditional pixel-domain motion estimation scheme outperforms the new scheme. Chapter 5 then discusses a frame-based hybrid motion estimation algorithm to select an appropriate domain of motion estimation in secondary SP-frame coding that could expedite the encoding process.

After the development of the frame-based hybrid algorithm, Chapter 6 proposes a more precise technique in which a specific domain for motion estimation can be selected on a macroblock basis rather than being the same for the whole frame. Flexible Macroblock Ordering is used to partition the SP-frames. The size of SP-frames can be further reduced and the computational complexity can be minimized.

Chapter 7 is devoted to a summary of the work herein and the conclusions reached as a result. Suggestions are also included for further research in this area.
Chapter 2  Literature Review

2.1  Introduction

Digital video has become extremely popular as a result of the digital revolution in the last two decades. Recent advances in network technology have further simulated the popularity of digital video applications such as video-on-demand, video streaming and video conferencing. Video standards such as H.264 [16] have been developed to provide standard video formats for convenient storage, process, and transmission. Due to the enormous amount of storage required by digital videos, numerous compression techniques such as transform coding and predictive coding are built and employed in the standards in order to reduce the video size while preserving the viewing quality. These techniques exploit all kinds of redundancy in the video data for achieving better compression ratio. In the meantime, dependencies are produced among the compressed video data together with compression effects. This is mainly due to the use of various predictive coding techniques in these video standards. While video streaming is becoming more and more important in various video applications, it typically imposes stringent bandwidth requirement on the underlying network since the bandwidth variation leads to high delay. To cope with the bandwidth variations, switching coded bitstreams with different bitrates is necessary. In additional, a new era of multi-view video increases the need of users' interaction for viewpoint switching. Recently, some research work related to bitstream
switching has been conducted in different ways and numerous algorithms have been proposed.

This chapter is organized as follows. In the first section, some fundamental concepts about the digital video representation are introduced. Then we illustrate various video compression techniques together with the generic encoder and decoder, followed by the H.264 video coding. Next, a conventional video switching method is presented. Bitstream switching is necessary on videos with different bitrates and different viewpoints. Wrong reference frames for prediction at switching point will lead to errors and the errors will propagate along the video. The problem is signified in multi-view video because the errors will be expanded within and across views. The final two sections review the problems in performing switching with different bitrates and different viewpoints.

2.2 Video compression fundamentals

2.2.1 Video compression techniques

Digital video is formed by a sequence of digital pictures which are created in the form of a two-dimensional matrix of individual picture elements called pixels. As compared with analog video, digital video includes no fading over time so that quality can be virtually preserved forever. It is more flexible as a storage form and more convenient for transmission through different types of networks. With the help of digital video processing tools, digital video can then be easily modified or edited.
Owing to the substantial amount of data in digital video, a number of video coding techniques have been developed over the decades. All of the advanced and efficient video coding techniques are built according to the characteristics of natural video and human visual perception [17-21]. They exploit some of the inherent redundancy in still images and moving sequences in order to provide significant data compression.

Real and natural images are usually smooth spatially. Although each image may consist of many objects and complex background, neighboring pixels in each of these areas have quite similar characteristics and relations. For instance, pixels in the dark background area have the values around zero with little variance. This spatial similarity is called spatial redundancy, and can be removed by using transform coding and quantization [22]. Besides, with the normal frame rate of 25-30 fps, neighboring frames in video are quite similar to each other. The similarity existing among neighboring frames is called temporal redundancy, and can be removed by motion-compensated predictive coding. A generic hybrid coding system that exploits all the redundancy mentioned above is illustrated in Figure 2.1. This hybrid motion-compensated coding model is widely used in nearly all video coding standards.
There are three major frame types adopted in various video standards. They are I-frames, P-frames and B-frames. I-frames, standing for intra-frames, are encoded independently from other frames and do not exploit temporal redundancy. The encoding process of I-frames is included in Figure 2.1(a). During I-frame encoding, the motion estimation and compensation processes are deactivated. The whole process is similar to the JPEG image compression standard [23], which was developed by the Joint Picture Expert Group. Each block of pixels is transformed into frequency domain to produce a set of transform coefficients. The output coefficients of the transformation are passed to the quantization process, which represents the sampled data at a finite...
number of levels. After the quantization, zig-zag scanning is used to arrange the quantized coefficients into a 1-D array for the entropy encoding.

Discrete cosine transform (DCT) [24-27] has been found to be more useful for image and video coding. The transformation packs the energy of the signal into a small number of coefficients, and this process paves the way for efficient compression. In real images, several DCT coefficients could be sufficient to represent the information of the whole block if its pixels vary smoothly without any edges. The DCT coefficients are ordered by frequency. Each DCT coefficient specifies the contribution of a sinusoidal pattern at a particular frequency to the actual signal. The most upper left corner coefficient is the DC coefficient, and represents the average value of the block. The other coefficients, known as AC coefficients, are arranged from low frequency to high frequency toward the lower right corner. Since the frequency response of the human eyes drops off with increasing spatial frequency, a small variation in intensity is more visible in slowly varying regions than in complex ones. Therefore, the DCT coefficients in the upper left corner, which represent low-frequency components, are more important than the high-frequency coefficients in the lower right corner.

In the quantization process [28-30], a 2-D quantization matrix is provided. Each DCT coefficient is divided by the corresponding quantization factor at the same position from the matrix and is then rounded to the nearest integer to obtain the quantized DCT value. This process aims at reducing the size of the DC and AC coefficients, and discards those unimportant coefficients so that fewer bits are
needed for storage or transmission. In the standards, the quantization matrix is
designed according to the visual sensitivity to different frequency components.
As discussed, the low-frequency coefficients in the upper left corner are more
visually important than the high-frequency coefficients in the lower right corner.
As a result, the low-frequency coefficients are assigned smaller quantization
factors than the high-frequency coefficients. In this way, the high-frequency
coefficients are quantized coarsely to achieve better compression efficiency
without greatly affecting visual quality. This also exploits the psycho-visual
redundancy phenomenon.

For generating an encoded bitstream, the quantized DCT coefficients are
scanned in a predetermined pattern. A common strategy is to zig-zag scan the
quantized DCT coefficients into a 1-D sequence as depicted in Figure 2.2. This
results in DCT coefficients in increasing order of frequency starting with the DC
coefficients and ending with the highest frequency AC coefficient. Reordering in
this zig-zag way tends to create long “runs” of zero-value coefficients and this is
beneficial to 2-D variable length coding (VLC) [31-33]. In the VLC process, the
philosophy is that those frequently occurring symbols are represented with short
codewords while less common symbols are represented with long codewords.
It is noted that the coding method of DC coefficients is different from that of AC
values. Owing to the small physical area covered by each block, the DC
coefficient varies only slowly from one block to the next. Hence, it is efficient to
use differential encoding in which the DC coefficient of the current block is
predicted from the DC value of the previous encoded block. The prediction
difference is encoded instead of the DC coefficient itself.
In addition to intra-frame coding, motion-compensated prediction [34-38] is employed to remove temporal redundancy between adjacent frames. In motion-compensated prediction, a pixel is not coded directly; rather, its value is predicted from those of adjacent pixels in a previous frame, which is usually called a reference frame. This is motivated by the fact that pixels in successive frames have similar color values, thus it is wasteful of bits to specify the current value independent of the past values. Both P- and B-frames take advantage of motion-compensated prediction to exploit the temporal redundancy. However, the coding of P-/B-frames is much more complicated than that of I-frames.

Figure 2.1(a) shows the key steps of motion-compensated prediction. In this case, both motion estimation and compensation are activated. In this encoder, each video frame is divided into blocks of a fixed size and each block is coded using a combination of motion-compensated prediction and transform coding. For the sake of simplicity, block-based motion estimation is employed in most video coding standards in which a block is predicted from a previously coded reference frame. In practice, the block size for motion estimation may not be
the same as that for transform coding. Generally, motion estimation is operated on a larger block known as a macroblock (MB). In most cases, the MB size is 16x16 pixels and the block size is 8x8 pixels. The motion estimation process uses the MB as a basic unit in which pixels of each MB in the current frame are compared on a pixel-by-pixel basis with pixels of the corresponding MB in the preceding I- or P-frame (reference frame). If a close match is located, the relative displacement between the current MB and the best-matched MB in the reference frame is encoded. This is known as a motion vector. The predicted MB is obtained from the reference frame based on the motion vector using motion compensation. The prediction error of the current MB is then coded by transforming it, quantizing the DCT coefficients, and converting them into variable length codewords using entropy coding. In principle, this procedure is similar to that described in encoding of I-frames.

Each motion vector consists of two motion vector components, the horizontal one first, followed by the vertical one, which are coded independently. Since the physical area of coverage of a MB is small, the motion vectors can have large values. Besides, most moving objects are normally much larger than a single MB. In this situation, adjacent MBs are moved in a similar way. Hence, motion vectors are coded differently with respect to previously decoded motion vectors so as to reduce the number of bits required to represent them. Each component of the differential motion vector is then coded by using a variable length codeword. To enhance the efficiency of motion compensation, half-pixel or quarter-pixel motion estimation is adopted in various video coding standards.
2.2.2 H.264 video coding

The most advanced video coding standard is H.264/AVC (ISO/IEC 13818) which is also known as MPEG-4 Part 10. It was jointly proposed by the ITU-T VCEG and the ISO/IEC MPEG in 2003. H.264 is targeted for providing similar functionality to the previous standards with significantly better compression performance. Besides, H.264 aims at having provision of a network-friendly video representation which addresses storage, broadcast and streaming applications.

There are some important terminologies and new concepts adopted in the H.264 standard. One of them is related to macroblocks. A coded picture consists of a number of macroblocks which are arranged in slices. A set of macroblocks of a slice is ordered in raster scan. An I-slice contains only an I macroblock type while a P-slice may contain P and I macroblock types. Besides, B-slice may contain B and I macroblock types. There are two more slice types which are SI and SP.

I macroblocks are predicted using intra prediction from decoded samples in the current slice. The prediction is formed in terms of a macroblock or a 4x4 block. P macroblocks are predicted using inter prediction from reference pictures. Variable block sizes are also supported. The inter-coded macroblock can be divided into 16x16, 8x16, 16x8 or 8x8 blocks. The 8x8 blocks can be further sub-divided into sizes of 8x8, 4x8, 8x4 and 4x4. B macroblocks are predicted using inter prediction from one or more reference pictures.
For the transform coding, H.264 applies integer transforms instead of DCT [39-43]. The size of these transform is mainly 4x4. Smaller block size enables better adaptation of prediction to the boundaries of moving objects. The advantage of integer transform is that only integer numbers are involved in the transform matrix. Integer operations including low complex shift, addition and subtraction can avoid mismatches of inverse transform.

Another new feature of H.264 is an adaptive deblocking filter [44-46]. The filter is applied to each decoded macroblock to reduce blocking distortion, resulting in smoothing block edges. The filtered image is used for motion-compensated prediction of future frames. The filter is highly adaptive, which means that the parameters, thresholds and local characteristics of picture itself control the strength of filtering process. Because of the fact that blocking artifacts become more severe when quantization gets coarse, the thresholds involved are all quantized dependent. Moreover, similar to integer transforms, the filter only contains additions and shifts. Computation is low and the improvement in subjective picture quality is remarkable.

As for error robustness and network friendliness, a coded video sequence in H.264 includes both Video Coding Layer (VCL) and Network Abstraction Layer (NAL). VLC specifies an efficient representation for coded video signal. On the other hand, each NAL unit contains a Raw Byte Sequence Payload (RBSP) which includes a set of coded video data or header information. Two concepts achieve essential conditions for real-time applications such as video streaming.
The Profiles of H.264 are also different from those defined in previous video standards. In H.264, three Profiles are specified. Each supports a particular set of coding functions.

Figure 2.3 shows the relationship between the three profiles and the coding tools supported by the standard. The Baseline Profile supports intra and inter-coding, and entropy coding with context-adaptive variable-length coding (CAVLC) [47-48]. The Main Profile includes interlaced video and inter-coding using B-slices. It enables inter-coding using weighted prediction and entropy coding using context-based arithmetic coding (CABAC) [49-52] as well. The Extended Profile does not support interlaced video or CABAC, but it enables
efficient switching between coded bitstreams using SP- and SI-slices. Error resilience with data partitioning is also supported in this profile. Furthermore, both Baseline Profile and Extended Profile support Flexible Macroblock Ordering (FMO) which allocates macroblocks into slice groups for error resilience. Potential applications of the Baseline Profile include videoconferencing and wireless communications. Main Profile is usually used for television broadcasting and video storage. Extended Profile is mainly employed in video streaming.

2.3 Conventional video switching method

Owing to the demand on video services, it is not suitable for sending an entire video bitstream to the user before playing them. Real-time streaming is an option to allow the video to play right after receiving a small portion of the whole video bitstream. This scenario requires real-time scaling and re-encoding of compressed video to cope with the bandwidth variations. However, this is not a practical solution.

The best way to accommodate the bandwidth scalability is to pre-encode video sequences independently with different bitrates and quality. Each source video is encoded into multiple non-scalable bitstreams with a preselected bitrate. An appropriate bitstream is chosen at each time-slot according to the network feedback. Then, the server can dynamically switch among streams. Such a scheme is extensively used in many commercial video streaming systems.
Figure 2.4. Conventional video streaming system.

Figure 2.4 depicts an ideal switching situation for the conventional video streaming system. There are two video bitstreams - B1 and B2. They are encoded in different bitrates using different quantization parameters. Assume that the network is inherently heterogeneous, and the effective channel bandwidth usually fluctuates in a wide range. When this happens at frame n, the switching is performed from high bitrate (B1) to low bitrate (B2). Figure 2.4 also shows that, in the conventional streaming system, the switching between B1 and B2 is only allowed at I-frame. It is because I-frame is intra-coded and it does not depend on information from any previous frames. It implies that all switching can only be performed at the beginning of each Group of Pictures (GOP).

Using an I-frame as a switching frame is simple. It can ensure that no drifting error is induced from bitstream switching. Nevertheless, if switching is triggered at the instant after the beginning of GOP, the remaining frames in the GOP will still be played. That means the switching can be only begun at the next GOP of another video bitstream which results in long delay.
To enable rapid and random switching between bitstreams, more I-frames can be inserted in the encoded bitstreams. However, it is well known that I-frames cost much more bits than the predictive frames to achieve the same decoded quality. Frequently inserting I-frames would inevitably decrease the coding efficiency of the bitstreams greatly.

### 2.4 Problem of bitstream switching using P-frames

From the above discussion, it is interesting to investigate a way to flexibly switch from one bitstream to another with high coding efficiency by using predictive frames, as depicted in Figure 2.5. In this figure, again, there is a video sequence encoded into two different bitstreams B1 and B2 with different bitrates. \( \{P_{1,t-2}, P_{1,t-1}, P_{1,t}, P_{1,t+1}, P_{1,t+2}\} \) and \( \{P_{2,t-2}, P_{2,t-1}, P_{2,t}, P_{2,t+1}, P_{2,t+2}\} \) show the sequence of the decoded frames from B1 and B2 respectively. Because of coming from the same sequence with different bitrates, the reconstructed frames at the same time instant are not identical. In the applications of bitstream switching, the server at the beginning sends B1 to the user. If the bandwidth of...
the network varies at time t, the server starts sending B2. Now, the user has received \{P_{1,t-2}, P_{1,t-1}, P_{2,t}, P_{2,t+1}, P_{2,t+2}\}. In this situation, \(P_{2,t}\) is obtained from the prediction of the reference frame \(P_{1,t-1}\). However, \(P_{2,t}\) needs to be predicted from \(P_{2,t-1}\) instead. Therefore, the wrong prediction or the mismatch between reference frames leads to visual artifacts. Furthermore, the visual artifacts are not only limited to \(P_{2,t}\). They will further propagate in time due to motion-compensated coding.

It is obvious that, without any special treatment, the switching between bitstreams at a P-frame would lead to drifting errors since it is always encoded using the prediction from the previous reconstructed reference frame. Furthermore, such errors could propagate and be accumulated in the subsequent P-frames until the next I-frame. It is noted that drifting errors would seriously deteriorate the decoded visual quality as the number of frames increases.

### 2.5 Viewpoint switching in multi-view videos

Research efforts on various multi-view video applications have been strengthened worldwide recently. Multi-view video coding (MVC) is an encoding framework for multiple video streams capturing from different viewpoints [10]. Since the amount of the raw data is very large, multi-view video needs much more bandwidth than traditional video. In the past few years, MPEG took up an interest in the area of coding efficiency on MVC and decided to call interested parties to bring evidence on this area in August 2004.
A common element of MVC systems is the use of multiple views of the same scene that have to be transmitted to users. These multiple videos enable different types of application scenarios depending on the adoption of coding techniques. On the one hand, the MPEG committee of the 3D audio-visual (3DAV) activity [53-56] group has worked towards the standardization for MVC with exploiting both temporal and inter-view [57-58] correlations by combining motion and inter-view prediction, as depicted in Figure 2.6. The inter-view prediction supported in the MVC standard can increase coding efficiency for all views and it is well suited for 3D video in which all views should be decoded in the 3D display [54]. In Figure 2.6, the MVC scheme uses a prediction structure of hierarchical B pictures for each view. Hierarchical B pictures can provide high Rate-Distortion performance. Besides, inter-view prediction is applied in every 2 views (S1, S3 and S5). A P-frame is started at each even number of views (including S7) and followed by hierarchical B structures. Typically GOP length is 8.

Figure 2.6. MVC prediction structure based on H.264 hierarchical B pictures.
On the other hand, some multi-view systems encode multiple videos independently (simulcast-coded) in order to facilitate a brand new viewing experience with a high degree of user interactivity. In the conventional single-view video, it only provides one view direction for an event at any time instance whereas users may want to watch the event from other directions. It means the users are very passive and can only watch the pre-selected video contents while there is no interactivity between the users and the content provider. With multiple videos, users can view a program with the freedom of view direction selection. Owing to its promising features, there are a large number of conceivable applications including interactive advertisement, educational program, sport games, etc. Although the inter-view prediction can provide better coding efficiency in the MVC standard, multiple views should be transmitted for decoding a single view due to its complex view-dependency structure. It is not a practical way to adopt the inter-view prediction in this scenario where viewpoint switching occurs frequently [15, 59-61]. In practical systems available in the market, it is desirable to develop a low-delay viewpoint switching scheme which can switch among different views upon requests instead of transmitting all the views. These systems provide interactive functions for the client users in multi-view video systems. For instance, an EyeVision system [62] was employed to shoot Super Bowl 2001 by using 30 cameras placed at different angles. Besides, in [60-61], a new streaming system for multi-view video was developed by Microsoft, which collects a number of video streams. These video streams are compressed independently (simulcast-coded) from control units. It provides interactive and reliable multi-
view video streaming based on server-client model. When views are simulcast-coded, this system can offer the interactive features such as viewpoint switching, frozen moment, and view sweeping. But, in these practical systems, switching between the bitstreams of different views is only allowed at I-frames, where the coding process does not rely on information from any previous frame, in order to avoid severe drifting problem. It is well-known that I-frames cost much more bits than P-frames. The interval between each I-frame in the bitstream of each view thus always keeps far apart. In this arrangement, switching cannot be taken place at any desired frame and this greatly affects the flexibility of the viewpoint switching capability. Alternatively, if I-frames are frequently inserted to support rapid switching between bitstreams of different viewpoints, the coding efficiency will be greatly reduced.

2.6 Chapter summary

In order to solve the difficulties of implementing video streaming, some problems related to bitstream switching have been reviewed in this chapter. We started this chapter by reviewing the compression techniques employed in current video standards. The redundancy exploited by these compression techniques provides good compression effects. The compression techniques also produce dependencies among the frames of the coded bitstream. These dependencies constrain the implementation of bitstream switching.

Therefore, the straightforward implementation of a conventional video switching system was also discussed. We showed that when users request to switch
between two bitstreams, switching can only be performed at I-frames. Next, we reviewed the problem of switching between different bitrates. Switching at P-frames will induce errors and the errors will propagate until the next I-frame. Furthermore, P-frames cannot be acted as switching points in multi-view videos as well. Similarly, if switching among viewpoints is not performed at I-frames, errors will propagate within and across views. Therefore, in the following chapters, we examine the possibility of bitstream switching in multiple bitrates and multiple viewpoints. Besides, SP-frames are used to achieve seamless switching on both application scenarios.
Chapter 3 Bitstream Switching in Multiple Bitrate Videos

3.1 Introduction

In the last chapter, we have shown the problem of a conventional bitstream switching mechanism. It is not possible to switch bitstreams at P-frames because it would lead to drifting errors due to the mismatch of the reconstructed references caused by different quantization steps. The switching is only allowed at I-frames. However, I-frames cost much more bits than P-frames and delay the switching time. In the meantime, H.264 includes a number of new features to achieve superior coding efficiency and provide more flexibility for applications to a wide variety of network environments. The SP-frame is one of these features. The motivation of introducing SP-frames is to facilitate error resilience, bitstream switching, splicing, random access, fast forward and fast backward [16, 63-65]. It can guarantee seamless switching between bitstreams. This property can accommodate the bandwidth variation by switching between compressed bitstreams with different bitrates. The tradeoffs of using SP-frames to perform switching are the coding performance and the storage cost of the frames. In this chapter, we thus design a novel motion estimation algorithm to reduce the size of SP-frames.

Part of the contents of this chapter have been published in references [66-67].
3.2 SP-frame encoding in H.264

H.264 SP-frame is a part of the Extended Profile. This special SP-frame is composed of primary and secondary SP-frames. They both exploit temporal redundancy with predictive coding, but use different reference frames. Although different reference frames are used, it still allows identical reconstruction. This property allows the use of SP-frames to replace I-frames for drift-free switching between compressed bitstreams of different quality and bitrates. Unlike I-frames, primary SP-frames are making use of motion-compensated predictive coding. For that reason, the coding efficiency of a primary SP-frame is much better than that of an I-frame and is slightly worse than that of a P-frame [68-73].

![Diagram](image)

Figure 3.1. Switching bitstream from B1 to B2 using SP-frames.

Figure 3.1 shows an example to apply SP-frames for drift-free switching between compressed bitstreams of different bit rates in order to accommodate the bandwidth variation. In this figure, a video sequence is encoded into two bitstreams (B1 and B2) with different bit rates. B1 is a sequence encoded in high bitrate while B2 is a low bitrate bitstream. Within each bitstream, two
primary SP-frames –SP₁,t and SP₂,t are placed at frame t (switching point). To allow seamless switching, a secondary SP-frame(SP₁₂,t) is produced, which has the same reconstructed values as SP₂,t even different reference frames are used. When switching from B₁ to B₂ is needed at frame t, SP₁₂,t instead of SP₂,t is transmitted. After decoding SP₁₂,t, the decoder can obtain exactly the same reconstructed values as normally SP₂,t decoded at frame t. Therefore it can continually decode B₂ at frame t+1 seamlessly.

Figure 3.2. Simplified encoding block diagram of primary and secondary SP-frames [6][14]

The way of encoding primary SP-frames is similar to that of encoding P-frames except additional quantization/dequantization steps with the quantization level Qs are applied to the transform coefficients of the primary SP-frame (SP₂,t in
Figure 3.1), as shown in Figure 3.2. Interested readers are encouraged to read the references [6, 14]. These extra steps ensure that the quantized-transform coefficients of $SP_{2,t}$ (denoted as $Qs[T(SP_{2,t})]$) are divisible by $Qs$, which is used in the encoding process of the secondary SP-frame, $SP_{12,t}$.

For coding $SP_{12,t}$, the reconstructed $P_{1,t-1}(\hat{P}_{i,j-1})$ acts as the reference and its target is to reconstruct $Qs[T(SP_{2,t})]$ perfectly. By using the reference frame $\hat{P}_{i,j-1}$, its prediction is firstly transformed and quantized using $Qs$ before generating the residue with $Qs[T(SP_{2,t})]$. Both the prediction and $Qs[T(SP_{2,t})]$ are thus synchronized to $Qs$ and there is no further quantization from this point, meaning that the decoder, with $\hat{P}_{i,j-1}$, $Qs$, and the residue available, can perfectly reconstruct $Qs[T(SP_{2,t})]$.

![Various modes and submodes in H.264.](image)

Producing secondary SP-frames also involves the processes of motion estimation and compensation. In H.264, it supports motion estimation using
different block sizes such as 16×16, 16×8, 8×16, 8×8, 8×4, 4×8, and 4×4 as depicted in Figure 3.3 [75-76]. To compute the coding modes and motion vectors for the secondary SP-frame, motion estimation is firstly performed for all modes and submodes independently by minimizing the Lagrangian cost function $J_{\text{motion}}$:

$$J_{\text{motion}}(mv_{12}, \lambda_{\text{motion}}) = SAD(s, r) + \lambda_{\text{motion}} \cdot R_{\text{motion}}(mv_{12} - pmv_{12})$$ (3.1)

where $mv_{12}$ is the motion vector used for prediction, $\lambda_{\text{motion}}$ is the Lagrangian multiplier for motion estimation, $R_{\text{motion}}(mv_{12} - pmv_{12})$ is the estimated number of bits for coding $mv_{12}$, and $SAD$ is the sum of absolute differences between the original block $s$ and its reference block $r$ [75-76].

After motion estimation for each mode, a rate-distortion (RD) optimization technique is used to obtain the best mode and its general equation is given by

$$J_{\text{mode}}(s,c,mode_{12}, \lambda_{\text{mode}}) = SSD(s,c,mode_{12}) + \lambda_{\text{mode}} \cdot R_{\text{mode}}(s,c,mode_{12})$$ (3.2)

where $\lambda_{\text{mode}}$ is the Lagrangian multiplier for mode decision, $mode_{12}$ is one of the candidate modes during motion estimation, $SSD$ is the sum of the squared differences between $s$ and its reconstruction block $c$, and $R_{\text{mode}}(s,c,mode_{12})$ represents the number of coding bits associated with the chosen mode. To compute $J_{\text{mode}}$, forward and inverse integer transforms, and variable length coding are performed. In the implementation of H.264 codec such as JM11.0 [13], motion estimation of the secondary SP-frame uses $\hat{P}_{1,t-1}$ and the original $SP_{1,t}$ as the reference and target frames respectively. This arrangement allows the reuse of coding modes ($mode_{1,t}$ in Figure 3.1) and motion vectors ($mv_{1,t}$ in Figure 3.1) during secondary SP-frame encoding. It means that
\[ mv_{12,t} = mv_{1,t} \]  
\[ (3.3) \]

and

\[ mode_{12,t} = mode_{1,t} \]  
\[ (3.4) \]

However, the reuse of coding modes and motion vectors reduces the coding efficiency of a secondary SP-frame since the purpose of the secondary SP-frame is to reconstruct \( SP_{2,t} \) instead of \( SP_{1,t} \). In [77], a secondary SP-frame is encoded to match the exact target frame (reconstructed \( SP_{2,t} \), \( \hat{SP}_{2,t} \)) based on the exact reference (\( \hat{P}_{1,t-1} \)), as depicted in Figure 3.4. By using the correct target and reference frames, better compression performance of the secondary SP-frames can be achieved. Note that the computational complexity evidently
increases without reusing coding modes and motion vectors. Besides, there is a trade-off between the coding performance of primary SP-frames and the storage cost for secondary SP-frames [74]. For example, a primary SP-frame with high quality results in a significantly high storage requirement for the secondary SP-frame. It is unfeasible to store such huge size of the secondary SP-frame.

3.3 Problem of using pixel-domain motion-compensated Prediction in secondary SP-frame encoding

Nevertheless, the improvement in [77] is not so significant. In this section, we explain the deficiency in using the conventional motion estimation and compensation processes, which are operated in the pixel domain, for secondary SP-frames. In the H.264 codec [13], the motion estimation process in P-frames and secondary SP-frames are the same.

![Figure 3.4](image.png)

Figure 3.4 illustrates the step of encoding a block in a P-frame using pixel-domain motion estimation. In this case, quantized coefficients of the P-frame
are computed as $Q_s[T[\hat{P}_{t,t} - MC(\hat{P}_{t-1,t})]]$, where $MC()$ is the motion-compensation operator. Since the transformation and quantization processes are performed on the residue, $P_{t,t} - MC(\hat{P}_{t-1,t})$, most of the coefficients become zero after transformation and quantization. This property paves the way for entropy coding. However, in Figure 3.4, the encoding of a secondary SP-frame involves firstly the transformation and quantization processes of the original $SP_{2,t}$ and $\hat{P}_{t-1,t}$. Then, the quantized coefficients of the secondary SP-frame at $t$, $Q_s[T(SP_{12,t})]$, can be obtained as,

$$Q_s[T[SP_{12,t}]] = Q_s[T[SP_{2,t}]] - Q_s[T[MС(\hat{P}_{t-1,t})]]$$  \hspace{1cm} (3.5)

<table>
<thead>
<tr>
<th>$SP_{2,t}$</th>
<th>$MC(P_{1,1,t})$</th>
</tr>
</thead>
<tbody>
<tr>
<td>247 247 249 210</td>
<td>224 248 255 192</td>
</tr>
<tr>
<td>254 254 248 200</td>
<td>255 193 228 255</td>
</tr>
<tr>
<td>254 254 210 195</td>
<td>193 255 213 251</td>
</tr>
<tr>
<td>254 254 222 184</td>
<td>212 193 208 204</td>
</tr>
</tbody>
</table>

$T+Q_s$ $T+Q_s$ $\text{Residue}$

| 15 1 -1 0 | 15 0 0 0 | 0 1 -1 0 |
| 0 0 0 0 | -1 0 0 1 | -1 0 0 -1 |
| 0 0 0 0 | -1 0 -1 1 | 1 0 1 -1 |
| 0 0 0 0 | 0 0 -2 -1 | 0 0 2 1 |

Figure 3.6. Motion-compensated prediction using pixel-domain estimation in encoding a secondary SP-frame.

Figure 3.6 uses the same example in Figure 3.5 again to show the residue of a secondary SP-frame in which a block is transformed and quantized prior to calculating the residue. In this case, their quantized-transform coefficients are
only near, but not equal, resulting in generating many non-zero residues, especially for a small $Q_s$. Since there is no further quantization from this point, these coefficients should be encoded completely. In entropy coding, the spread of non-zero coefficients exists, a significant amount of bits is required. Therefore, the size of secondary SP-frames becomes large, and this explains the situation in which the pixel-domain motion estimation is not sufficient enough for coding secondary SP-frames.

### 3.4 Proposed scheme for secondary SP-frame encoding

From Equation (3.5), to optimize the coding of $Q_s[T[SP_{12,t}]]$ for better entropy coding, quantized-transform domain ($Q_s[T[SP_{2,t}]] - Q_s[T[MC(\hat{P}_{1,t-1})]]$) instead of its pixel-domain counterpart should be minimized. In pixel-domain motion estimation, the Lagrangian cost function $J_{\text{motion}}$ in Equation (3.1) computes SAD, which is the sum of absolute differences between pixels of the original block $s$ and its reference block $r$. Figure 3.6 reveals that this pixel-domain distortion measure is not appropriate for coding secondary SP-frames. Consequently, a quantized-transform domain motion estimation (QDCT-ME) technique that minimizes $Q_s[T[SP_{2,t}]] - Q_s[T[MC(\hat{P}_{1,t-1})]]$ is adopted in coding secondary SP-frames. In the proposed QDCT-ME, the new Lagrangian cost function $J'_{\text{motion}}$ is based on the sum of absolute differences in quantized-transform coefficients and the estimated rate. This new cost function is expressed as

$$J'_{\text{motion}}(mv_{12}, \lambda_{\text{motion}}) = k \cdot SAQTD(s, r) + \lambda_{\text{motion}} \cdot R_{\text{motion}}(mv_{12} - pmv_{12})$$

where $SAQTD(s, r)$ denotes the sum of absolute differences between the...
quantized-transform coefficients of the original block \( s \) and the quantized-transform coefficients of its reference block \( r \), and it can be defined as,

\[
SAQTD(s, r) = \sum |Q_s[T(s)] - Q_s[T(r)]|
\]

(3.7)

and \( R_{motion}(mv_{12} - pv_{12}) \) is the estimated number of bits for motion vectors by using a simple table-lookup method. From Equations (3.6) and (3.7), \( SAQTD(s, r) \) requires a real quantization process to calculate the sum of absolute differences. But, the Lagrangian multiplier, \( \lambda_{motion} \), is specifically designed for SAD, which is not affected by quantization. It is important to note that the proposed \( SAQTD(s, r) \) is operated in the quantized DCT domain and the extra quantization causes the energy is no longer preserved although the integer transform itself is an orthonormal transform. After absorbing \( Q_s \) into the new cost function of the proposed QDCT-ME, \( k \) in Equation (3.6) is a weighting factor to compensate for the energy loss of \( SAQTD(s, r) \) due to the quantization, which can be predefined by empirical results. It is noted that a new \( \lambda_{motion} \) can be set and combined with \( k \).
Similar to pixel-domain motion estimation, QDCT-ME supports variable block sizes (from 4×4 to 16×16). Figure 3.7(a) illustrates how to perform QDCT-ME for a 4×4 block in which a macroblock is subdivided into sixteen 4×4 blocks.
Each current $4 \times 4$ block is transformed and quantized to $Q_s[T(s)]$, of course, in terms of $4 \times 4$ transform. A search window in the reference frame centred on the current $4 \times 4$ block position is searched. QDCT-ME starts with transforming and quantizing a $4 \times 4$ block, $Q_s[T(r)]$, in the top right-hand corner of the search window. After obtaining $Q_s[T(s)]$ and $Q_s[T(r)]$, $SAQTD(s,r)$ and $J_{motion}^{'}(m_{v12}, \lambda_{motion})$ in Equation(3.6) can be calculated for this candidate. Then, the next candidate is another $4 \times 4$ block shifted by 1 pixel in horizontal direction, as depicted in Figure 3.7(a), and this block is also $4 \times 4$ transformed and quantized for computing its $J_{motion}^{'}(m_{v12}, \lambda_{motion})$. These procedures continue for all possible candidates within the search window. Note that all possible candidates also need to be transformed and quantized in the QDCT domain.

As a consequence, the candidate that minimizes $J_{motion}^{'}(m_{v12}, \lambda_{motion})$ is chosen as the best match. For an $8 \times 8$ block, it contains four $4 \times 4$ blocks, and all these $4 \times 4$ blocks are $4 \times 4$ transformed and quantized to $Q_s[T(s)]$, as depicted in Figure 3.7(b). These four $4 \times 4$ quantized-transform coefficients are used as a whole to compute $SAQTD(s,r)$. For other block size, a similar $4 \times 4$ transform is also applied.

For the implementation, Figure 3.8 shows the block diagram of applying our new quantized-transform domain motion estimation technique in the secondary SP-frame encoder. The reference and target frames in the quantized-transform domain are the inputs of QDCT-ME. After the motion vectors for each block are obtained through Equation(3.6), a corresponding quantized-transform domain
motion compensation (QDCT-MC) is used to compute the $Q_s[T[MC(\hat{P}_{1,t-1})]]$.

With $Q_s[T[MC(\hat{P}_{1,t-1})]]$ and $Q_s[T[SP_{2,t}]]$, as depicted in Figure 3.8, the residue $Q_s[T[SP_{2,t}]]$ can then be calculated.

![Diagram](image)

Figure 3.8. The proposed secondary SP-frame encoder in the QDCT domain.

### 3.5 Experimental results

In order to evaluate the performances of the proposed scheme and the scheme in [77], three test sequences, “Foreman” (CIF), “Salesman” (CIF) and “Table Tennis” (SIF) were used in our experiments. The H.264 reference codec (JM11.0 [13]) was employed to encode primary SP-frames and secondary SP-
frames with a frame rate of 30 fps. All test sequences have a length of 200 frames. We used two different bitrate bitstreams encoded with two different sets of $Q_P$ and $Q_S$, and only the switching from a low bitrate bitstream to a high bitrate bitstream is shown. For the low bitrate bitstream, $Q_{P1}$ and $Q_{S1}$ were both fixed to 41, whereas $Q_{P2}$ and $Q_S$ were both set to 21 for the high bitrate bitstream. To have impartial comparison between both schemes, every frame was encoded in turn as an SP-frame while non-switching frames were encoded as P-frames.
The frame-by-frame comparisons of size reduction of secondary SP-frames are shown in Figure 3.9(a)-(c). In this figure, the positive values of the Y-axis mean the size reduction of a secondary SP-frame in percentage difference of our proposed scheme over the scheme in [77] whereas the negative values mean the proposed scheme generates more bit-count as compare to [77]. From Figure 3.9(a)-(c), it is observed that the proposed scheme can substantially reduce the size of secondary SP-frames, about 3%, 4.5% and 3% in average and up to 30%, 12% and 10% in “Foreman”, “Table Tennis” and “Salesman”, respectively. Note that there is no change of distortion since the switching is seamless. The distortion of a secondary SP-frame is exactly the same as that of the target primary SP-frame being switched to. The significant improvement of the proposed scheme is due to the benefit of performing motion estimation and compensation in the QDCT domain. In [77], even though a proper target frame
is selected for motion estimation, the performance is still not significant. It is due to the reason that only the conventional pixel-domain motion estimation technique is employed for coding secondary SP-frames. In this situation, most of transformed coefficients become non-zero after transformation and quantization, as shown in Figure 3.5, which unfavour the use of entropy coding. Consequently, more bits are required to encode secondary SP-frames. On the other hand, our proposed scheme produces secondary SP-frames using motion estimation in the QDCT domain. The quantized and transformed coefficients are used to calculate the distortion in the Lagrangian cost function. The new SAQTD really finds the motion vector with more coefficients to be zero that benefits the entropy coding of secondary SP-frames. This provides the remarkable size reduction of our proposed scheme.

3.6 Conclusion

In this chapter, an efficient scheme for coding H.264 secondary SP-frames has been proposed. We found that the use of conventional pixel-domain motion estimation is not appropriate for a secondary SP-frame encoder, which incurs considerable size of secondary SP-frames. To alleviate this, we have incorporated the QDCT-domain motion estimation technique in the encoding process of secondary SP-frames. The new technique utilizes an appropriate cost function, which is operated in the QDCT domain, for motion estimation in coding of secondary SP-frames. Based on the new metrics, experimental results showed that the proposed scheme can significantly reduce the size of
H.264 secondary SP-frames. Besides, the proposed technique does not affect the coding efficiency of primary SP-frames.
Chapter 4 Viewpoint Switching in Multi-view Videos

4.1 Introduction

We have seen from the previous chapter that the use of the proposed quantized-transform domain motion estimation algorithm is able to efficiently reduce the size of secondary SP-frames in multiple bitrate video (MBV). Meanwhile, multi-view video (MVV) service has been attracting more and more attention recently. MVV system aims at providing interactive and reliable multi-view video streaming based on the server-client model. Similar to MBV, switching between MVV bitstreams is only allowed at I-frames to avoid severe drifting problem. In this case, switching cannot be taken place at any desired frame. Therefore, in this chapter, H.264 SP-encoding is proposed to be directly adopted in MVV. Besides, the idea of quantized-transform domain motion estimation is also implemented to minimize the size of secondary SP-frames in MVV.

Details of the scheme are shown in the following sections, while results of this chapter have been published in references [67] and [78].
4.2 Motivation of using SP-frames in multi-view videos

With the rapid advancement of capture and display technologies, multi-view video/3D video is becoming a reality in consumer domain with different application opportunities. Many different envisioned 3D video applications are getting practical. These applications can be mainly divided into two categories: 3D TV and free-viewpoint video. The requirements of these applications are quite different and each category has its own challenges to be addressed. 3D TV is focused on the extension of traditional 2D TV displays to stereoscopic or 3D displays. In this application, more than one view is transmitted, decoded and displayed simultaneously. On the other hand, free-viewpoint video emphasizes on its functionality in free navigation. The distinguishing feature of free-viewpoint video lies on interactivity, which allows users to select their favourite viewpoint. It switches bitstreams at a particular view when necessary instead of transmitting all the views. At first, the new SP-frame in H.264 is developed for multiple bit-rate streaming with the support of seamless switching. By the same token, the SP-frame can then be directly adopted for low-delay viewpoint switching in multi-view streaming applications. Moreover, for MVV, an additional sequence of secondary SP-frames is required between each view pair. For example, the extra N-1 sequences of secondary SP-frames are stored in the MVV system with N views. There is a great need for reducing the size of secondary SP-frames, which can alleviate the storage requirement of the server.
4.3 Deficiency of conventional SP-frames in multi-view videos

Figure 4.1 illustrates the idea of viewpoint switching in a multi-view video using SP-frames. In this figure, two different views, which are captured by two cameras at the same time in the same scene, are encoded into two bitstreams (V1 and V2). V1 is an encoded bitstream in viewpoint 1 while V2 is in viewpoint 2. Within each view, two primary SP-frames, denoted by $SP_{1,t}$ and $SP_{2,t}$, are inserted at frame $t$ (switching point). To facilitate seamless switching, a secondary SP-frame ($SP_{12,t}$) is also required. $SP_{12,t}$ ensures the same reconstructed values as $SP_{2,t}$ in spite of using different reference frames. When switching from V1 to V2 is requested at frame $t$, $SP_{12,t}$ instead of $SP_{2,t}$ (or $SP_{1,t}$) is sent. After the decoder decodes $SP_{12,t}$, it can obtain exactly the same reconstructed values as normally $SP_{2,t}$ decoded at frame $t$. As a consequence, it can continually decode V2 at frame $t+1$ seamlessly.
Similar to MBV in the previous chapter, secondary SP-frames generated by pixel-domain motion estimation in [77], as shown in Figure 3.4, can be directly implemented in MVV. It is noted that by appropriately choosing reference and target frames for mode selection and motion estimation [77], the size of secondary SP-frames can be reduced for the representation of MBV.

Table 4.1. Average bit-counts of secondary SP-frames in MVV and MBV.

<table>
<thead>
<tr>
<th>Sequences</th>
<th>MVV (kbits)</th>
<th>MBV (kbits)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ballroom</td>
<td>353</td>
<td>221</td>
</tr>
<tr>
<td>Exit</td>
<td>176</td>
<td>124</td>
</tr>
<tr>
<td>Vassar</td>
<td>275</td>
<td>201</td>
</tr>
<tr>
<td>Race1</td>
<td>258</td>
<td>214</td>
</tr>
</tbody>
</table>

However, we found that the storage requirement for multi-view video becomes large when secondary SP-frames are acted as the bridging frames for viewpoint switching in MVV streaming. The average storage requirements of secondary SP-frames applied to MBV and MVV applications are shown in Table 4.1 where the testing multi-view video streams used for the simulation are “Ballroom”, “Exit”, “Vassar”, and “Race1” with a length of 100 frames. As in Figure 3.4, for each MBV, we only selected the first view for the simulation, and the quantization parameters of the high bitrate video ($Q_{P1}$) and the low bitrate video ($Q_{P2}$) were set to 24 and 28, respectively. On the other hand, for each MVV, the first and second views were used for the simulation. To have a fair comparison, the quantization parameters were also set to 24 and 28, respectively. For both cases, $\hat{I}_{t,t-1}$ and $SP_{2,t}$ are used as the reference frame and the target frame respectively. From Table 4.1, we can observe that the bits required in coding secondary SP-frames of MVV are much higher than those of MBV. The rise in
the bit-counts of MVV is due to the reason that the conventional motion estimation and compensation processes work ineffectively in secondary SP-frame coding. It is because the inter-viewpoint disparity in MVV causes an increase in the residue between $\hat{P}_{1,t-1}$ and $SP_{2,t}$ even though the presence of disparity between views can be reduced by correctly selecting reference and target frames. In other words, for coding the secondary SP-frame $SP_{12,t}$, the reference frame and the target frame are $\hat{P}_{1,t-1}$ and $SP_{2,t}$, respectively, and they come from different viewpoints of the same scene. It implies that both temporal motion and inter-view disparity exist between the reference and target frames in MVV. On the other hand, for MBV, both reference and target frames represent the same viewpoint of the scene and the only discrepancy is the quantization parameter used in the formation of the encoded bitstreams. Therefore, the correlation between the reference and target frames of MBV is always larger than that of MVV in secondary SP-frame coding.

4.4 Proposed QDCT-ME for secondary SP-frames in multi-view videos

As discussed in Section 3.3, the secondary SP-frame encoder natively generates many non-zero quantized-transform coefficients. In entropy coding, a significant amount of bits is required if the spread of non-zero coefficients exists. This case happens more often in multi-view video sequences since V1 and V2 are from different views and their correlation becomes lower, despite the fact
that the motion estimation scheme using the correct reference and target frames in V1 and V2 can exploit both motion and disparity in MVV.

![Energy distribution of quantized-transform coefficients in the residue for the “Ballroom” sequence in MBV: (a) the scheme in [77], (b) the proposed scheme, and (c) their difference.](image1)

Figure 4.2. Energy distribution of quantized-transform coefficients in the residue for the “Ballroom” sequence in MBV: (a) the scheme in [77], (b) the proposed scheme, and (c) their difference.

![Energy distribution of quantized-transform coefficients in the residue for the “Ballroom” sequence in MVV: (a) the scheme in [77], (b) the proposed scheme, and (c) their difference.](image2)

Figure 4.3. Energy distribution of quantized-transform coefficients in the residue for the “Ballroom” sequence in MVV: (a) the scheme in [77], (b) the proposed scheme, and (c) their difference.

Figure 4.2(a) and Figure 4.3(a) shows the energy distribution of quantized-transform coefficients for the “Ballroom” sequence in the MBV and MVV applications, respectively, when the scheme in [77] is adopted. Notice that the DC coefficients for these plots are omitted since the value of DC coefficients is relatively large and it affects the plotting scale. Again, for both cases, \(Q_{P1}\) and
were set to 24 and 28, respectively, in order to ensure the difference only comes from the presence of disparity between views. From these figures, it is clear to show that there are substantial values of quantized-transform coefficients in MVV as compared with those in MBV, which reduces the compression ratio of secondary SP-frames. It is confirmed that a smart scheme for secondary SP-frame coding is necessary to reduce these quantized-transform coefficients in bitstream switching in multi-view video systems.

Thus, QDCT-domain motion estimation can be embedded into secondary SP-frames in MVV. As in MBV, QDCT-ME involves Lagrangian cost function which calculate the sum of absolute difference between quantized-transform coefficients of reference frame $\hat{P}_{1,t-1}$ and the target frame $SP_{2,t}$. The cost function enhances the coding efficiency of secondary SP-frames which leads to a significant reduction of the storage requirement in the video server.

In summary, the pixel-domain motion estimation is to minimize the absolute differences in which the subtraction is performed in terms of pixels while the quantized-transform domain motion estimation is to minimize the absolute differences in which the subtraction is performed in terms of transformed and quantized coefficients.

4.5 Experimental results

To evaluate the performance of the proposed scheme, this section presents the simulation results in both MVV and MBV. For MVV, five test sequences with
eight views, “Ballroom” (640×480, from MERL[79]), “Exit” (640×480, from MERL[79]), “Vassar” (640×480, from MERL[79]), “Race1” (640×480, from KDDI [80]), and “Breakdancers” (1024×768, from Microsoft) were used for performance comparison in coding secondary SP-frames. “Ballroom”, “Exit”, and “Vassar” were captured by 1D parallel-aligned cameras while the camera arrangement of “Breakdancers” was 1D arc. In addition, “Race1” was produced by the moving set but fixed relative positions of cameras by KDDI [80]. Notice that the eight views in “Breakdancers” and “Race1” were non-rectified whereas those in other sequences were rectified. For MBV, two single-view sequences with complex motion, “Riverbed” (704×480) and “Crew” (1280×720), were used to demonstrate the performance of the proposed scheme. In each sequence, 100 frames were encoded by the H.264 reference codec (JM version 11.0) [13]. In our experiments, extended profile with CAVLC entropy encoding was used to configure the encoder. R-D optimization was enabled. For the motion estimation process, a search range of 32 was set for both P-frames and SP-frames. For the QDCT-ME scheme, the scaling factor k of Equation (3.6) was set to 3, which was found by experimental observations. There were two sets of experiments carried out. First, we focus on the performance of the proposed QDCT-domain motion-compensated prediction technique applying to secondary SP-frame coding in multi-view videos and single-view videos. Second, we demonstrate the storage reduction in a multi-view video coding system with the proposed technique. In both experiments, results are compared to the scheme in [77]. To make the comparison impartial, both schemes employed a full search algorithm for motion estimation.
4.5.1 Performance of QDCT-domain technique in secondary SP-frame

(i) Viewpoint switching in multi-view video

The first set of experiments aims at evaluating the coding efficiency of secondary SP-frames based on our proposed QDCT-domain motion-compensated prediction. For simplicity but without loss of generality, only two views (view 1 and view 2) were selected to perform switching using SP-frames. These two different bitstreams from two different viewpoints were encoded with the same sets of $Q_{P1}$, $Q_{P2}$, and $Q_S$. $Q_{P1}$ and $Q_{P2}$ were equal for encoding these two views, that is, $Q_{P1} = Q_{P2} = Q_P$. They were then set to 28 whereas $Q_S$ was set to 22 ($Q_P - 6$), which is the optimal setting according to [14]. To have a comprehensive and fair comparison between the proposed scheme and the scheme in [77], we did exhaustive simulation on the bit-counts of all possible secondary SP-frames switching from view 1 to view 2. That is, only the first frames of the two views were encoded as I-frames, and all the rest frames were encoded in turn as SP-frames while non-switching frames were encoded as P-frames.
Figure 4.4. Frame-by frame comparisons of size reduction of secondary SP-frames in percentage achieved by the proposed scheme over the scheme in [77] at QP = 28, (a) “Ballroom”, (b) “Exit”, (c) “Vassar”, (d) “Race1”, and (e) “Breakdancers”. 
Figure 4.4(a)-(e) show a frame-by-frame comparison of size reduction in secondary SP-frames for different multi-view sequences. In this figure, the values of the Y-axis mean the average size reduction of secondary SP-frames in percentage of our proposed scheme over the scheme in [77] whereas the negative values mean the proposed scheme generates more bit-count as compared with [77]. From Figure 4.4 (a)-(e), the proposed scheme can overwhelmingly reduce the size of secondary SP-frames of up to 11%, 25%, 7%, 10% and 13% in the SP-frames of “Ballroom”, “Exit”, “Vassar”, “Race1”, and “Breakdancers”, respectively. The remarkable improvement of the proposed scheme comes from the advantage of motion estimation and compensation in the QDCT domain. In [77], even though a proper target frame is selected for motion estimation, the performance is still not good enough. It is due to the fact that most of the quantized-transform coefficients become non-zero after transformation and quantization for pixel-domain motion estimation. As a result, more bits are required to encode secondary SP-frames. In contrast, our proposed scheme encodes secondary SP-frames using motion estimation in the QDCT domain in which the new SAQTD is capable of locating the motion vectors with more zero values of quantized-transform coefficients. This arrangement paves the way for the entropy coding of secondary SP-frames. It then achieves significant size reduction of our proposed scheme. Note that, as shown in Figure 4.4, some secondary SP-frames of the test sequences generated by using the QDCT-domain motion estimation technique introduces more bits as compared with the pixel-domain technique. It is due to the distribution of quantized-transform coefficients in some 4×4 blocks where all pixels are similar or with the same value. In these blocks, their quantized-
transform coefficients are mainly zeros except the DC coefficient. If this type of QDCT-domain blocks is used to search over a predetermined search area on the transformed and quantized reference frame, all the SAQTD values are very similar and the motion estimation process becomes very sensitive to noise. In this case, the smallest SAQTD value may not be presumed to be the best motion vector. Thus, the bit-counts of some macroblocks in the secondary SP-frames encoding by pixel-domain motion estimation are possible to be slightly smaller than those encoding by QDCT-domain motion estimation.

![Figure 4.5](image)

Figure 4.5. (a) Reference frame in view 1 of “Ballroom”. Bit-count comparison in macroblock level between the proposed scheme and the scheme in [77] in (b) MVV, and (c) MBV. Macroblocks filled with white colour and black colour, which represent our proposed scheme have superior and inferior performances as compared with the scheme in [77], respectively.

The evidence is shown in Figure 4.5 where the bit-count comparison in all macroblocks of one switching frame between the proposed scheme and the scheme in [77] in “Ballroom” is illustrated. In this example, the reference frame from view 1 is depicted in Figure 4.5(a). In Figure 4.5(b), the macroblocks filled with white colour are those macroblocks that the proposed scheme requires less bit-count as compared with the scheme in [77], and they are dominated in this frame. On the other hand, the macroblocks filled with black colour show the
region has inferior performance for our proposed scheme. They are mainly in the regions of the curtain and the floor that contain similar pixel values within the macroblock. However, it only happens in a few macroblocks in which the difference in bit-counts is very tiny. To avoid this, one straightforward way is to perform both pixel-domain and QDCT-domain motion estimation, and select the motion vector with the minimum bit-counts. Notice that the macroblocks without filling any colour in Figure 4.5(b) represent same bit-counts generated by both schemes.
Figure 4.6. Size reduction of secondary SP-frames in percentage achieved by the proposed scheme over the scheme in [77] with different Qp, (a) “Ballroom”, (b) “Exit”, (c) “Vassar”, (d) “Race1”, and (e) “Breakdancers”.
Figure 4.6(a)-(e) also show the average percentage reduction in size of secondary SP-frames with different $Q_P$ for various sequences. In this experiment, $Q_P$ was varied from 16 to 28 with a stepsize of 4. These quantization parameters lead to a bitrate variation of the sequences in different viewpoints from approximately 1.4Mbps to 160kbps. Again, $Q_S$ was set to $Q_P - 6$. It is obvious from Figure 4.6(a)-(e) that, by using the proposed QDCT-domain motion estimation, the size of secondary SP-frames can be remarkably reduced. This large reduction highlights the importance of using a proper domain for motion estimation in secondary SP-frame encoding. Figure 4.6(e) also shows that the overall gain of the “Breakdancers” sequence is higher than the other sequences. The reason is that, in “Breakdancers”, it is difficult to perform good disparity compensation using translational approximation (i.e. based on block-matching) due to the large baseline distance of camera arrangement. It results in low interview correlation. In this situation, directly minimizing the quantized-transform coefficients by the new SAQTD is more beneficial to the entropy coding of secondary SP-frames.

(ii) **Bitstream switching in multiple bitrate video with complex motion**
Figure 4.7. Size reduction of secondary SP-frames in percentage achieved by the proposed scheme over the scheme in [77] for “Ballroom” in MBV and MVV.

To demonstrate the impact of the proposed scheme on bitstream switching in MBV, Figure 4.7 gives the comparison of size reduction (in percentage) in secondary SP-frames for “Ballroom” in MBV and MVV applications. The high and low bitrate bitstreams of MBV were generated by encoding only the first view of “Ballroom” with $Q_{P1}$ and $Q_{P2}$ being equal to 24 and 28, respectively. For MVV, the first and second views were also encoded by setting $Q_{P1}$ and $Q_{P2}$ to 24 and 28, respectively, in order to have a fair comparison. As usual, $Q_S$ was set to 22. In Figure 4.7, only a very small size reduction of secondary SP-frames is obtained in MBV while the sizes of secondary SP-frames are considerably reduced in MVV. Notice that, in MBV, the bitstream switching is taken place from high to low bit rate video. As discussed in the previous chapter, for the scheme in [77], the energy in quantized-transform coefficients is already very small, as depicted in Figure 4.2(a). The room for improvement is very limited. Figure 4.2(b) also shows the energy distribution of quantized-transform coefficients for using our proposed QDCT-domain motion estimation.
It can be seen that the improvement is not noticeable, as illustrated in Figure 4.2(c) in which the distribution of the energy difference between the scheme in [77] and the proposed scheme is computed. The positive values represent that our proposed scheme produces smaller values or magnitudes of coefficients to be encoded. To have a closer look at the macroblock level in the case of MBV, the bit-count comparison of all macroblocks between the proposed scheme and the scheme in [77] is depicted in Figure 4.5(c). In the case of MBV shown in Figure 4.5(c), although the number of the macroblocks filled with white colour is more than that of the macroblocks filled with black colour, it is not remarkable as compared with Figure 4.5(b), since many macroblocks in the background region containing audiences produce same bit-counts. It is contrast to the case of MVV where the energy in the quantized-transform coefficients becomes larger, as shown in Figure 4.3(a), due to the presence of disparity. This situation favours to our proposed SAQTD, which is intended to minimize the residue in the QDCT domain. As a result, a considerable decrease in the energy of quantized-transform coefficients can be achieved, as illustrated in Figure 4.3 (b) and Figure 4.3 (c).

Table 4.2. The percentages of average size reduction of secondary SP-frames of our proposed scheme over the scheme in [77] in MBV with various skipping factors.

<table>
<thead>
<tr>
<th>no of skip frames</th>
<th>size reduction (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0.4831</td>
</tr>
<tr>
<td>1</td>
<td>0.5978</td>
</tr>
<tr>
<td>2</td>
<td>0.8374</td>
</tr>
<tr>
<td>3</td>
<td>0.9183</td>
</tr>
</tbody>
</table>

It is interesting to note that the proposed SAQTD can perform better in the case of MBV containing scenes with high motion activity. To verify this, for the sake
of simplicity, we could simulate high motion scenes in “Ballroom” by regularly dropping some frames. Table 4.2 shows the size reduction in secondary SP-frames for “Ballroom” in MBV with various skipping factors. It can be seen from this table that the percentages of average size reduction of the secondary SP-frames of our proposed scheme over the scheme in [77] are augmented as more frames are skipped in MBV since the interframe correlation decreases as skipping factor rises.

It is clear that the performance of the proposed SAQTD is determined by the degree of interframe correlation in MBV. The weaker the interframe correlation is, the better the SAQTD can provide. To further demonstrate this, Figure 4.8 shows a frame-by-frame comparison of size reduction in secondary SP-frames for the “Riverbed” and “Crew” sequences. In this figure, we used the same setting as the case of encoding the single-view video of “Ballroom” mentioned above. It is observed that the size reduction of secondary SP-frames is up to 8% and 33% in the SP-frames of “Riverbed” and “Crew” respectively. As
comparing with the MBV result of “Ballroom” from Figure 4.7, the proposed scheme applying on these two sequences can give more significant improvement. The merit of our proposed scheme is due to the inherent property of these video sequences. In the first view of “Ballroom”, owing to the frames of MBV being captured by the same camera at the same time in the same scene, the translational motion model could be good enough for practical purposes. Consequently, it generates smaller quantized-transform coefficients in secondary SP-frames. In contrast, “Riverbed” is the shot of a riverbed seen through the water while “Crew” contains NASA crew leaving a building with flashlights. In these sequences, they both include scenes with challenging motion in which translational motion is not able to find the true motion in these scenes. The scheme in [77] cannot find suitable motion vectors in pixel domain. The incorrect motion vectors cause an increase in residue of secondary SP-frames.

The evidence is shown in Figure 4.9 where the average energy of the residue in quantized-transform domain of “Ballroom”, “Riverbed”, and “Crew” are plotted. In this figure, it can be easily seen that the quantized-transform coefficients in
the residue of “Riverbed” and “Crew” are much larger than those of “Ballroom”. It implies that directly minimizing the quantized-transform coefficients using the proposed scheme is more crucial even though the obtained motion vectors are not reflecting the real motion. In other words, the QDCT-domain motion estimation aims at minimizing the quantized-transform coefficients and can provide remarkable improvement over the scheme in [77] when the complex motion exists.

4.5.2 Storage requirement of secondary SP-frames in multi-view video coding system

![Multi-view Video Coding System](image)

Figure 4.10. Multi-view Video Coding System.

To illustrate more concretely the benefits of adopting the proposed scheme in streaming multi-view video, we implemented a multi-view video system with the
support of low-delay viewpoint switching. For this practical system, a server-client architecture was used. The client is connected to the video server over an IP network and requests access to the specific view of the encoded multi-view video, which is stored in the server. In the encoded multi-view video, SP-frames are inserted periodically at intervals of 6 frames in each bitstreams, as depicted in Figure 4.10. In other words, we encoded a primary SP-frame every 6 frames while other frames were encoded as P-frames. Other settings are the same as those in the above experiments. Besides, all processes of viewpoint switching among 8 views are performed at SP-frames only. At each switching point, there are a total number of 7 secondary SP-frames that have to be prepared and stored in the server to support viewpoint switching among 8 views. This scenario is more applicable for low-delay viewpoint switching in a real multi-view video system. Note that only one-way switching direction is assumed in this experiment for simplicity. For viewpoint switching with both directions, another 7 secondary SP-frames in the opposite switching direction are added similarly. When a user subscribes to the server, the multi-view video service will be provided. Two logical channels are established between the server and the client: the data channel and the control channel. The data channel, implemented by Real-time Transport Protocol (RTP) in our system, is used to deliver the requested bitstream of multi-view video. This RTP standardizes the packet format for delivering video over the Internet. In addition, the server receives viewpoint switching command through the control channel, which was implemented by the Real Time Streaming Protocol (RTSP). If a viewpoint switching command is received, the server will continue sending video bitstream of the current view direction until reaching the next SP frame. After that, it will
send the video bitstream of the new viewpoint from that SP frame. For the example shown in Figure 4.10, a user is watching view 1 and can interact with the server. At frame 9, the user changes his/her viewpoint to 2, the switching command sends from the client to the server through RTSP. The server cannot perform switching immediately, and waits for $SP_{12}$ at frame 12. The server then sends this SP frame to the client which can now watch V2 from frame 12. $SP_{12}$ can perfectly reconstruct the primary SP-frame of V2 so that mismatch will not occur when switching is needed.
Figure 4.11. Rate-distortion performance with periodic I-frame and SP-frame insertion for (a) “Ballroom”, (b) “Exit”, (c) “Vassar”, (d) “Race1”, and (e) “Breakdancers”.

Before we analyze the storage requirement of secondary SP-frames encoded by our proposed scheme in the multi-view video coding system, Figure 4.11 gives the RD-performance when primary SP-frames are used instead of
periodic I-frame insertion for the first view of “Ballroom”, “Exit”, “Vassar”, “Race1”, and “Breakdancers”. This figure also shows the results of the bitstreams encoded by all P-frames with the exception of the first frame, which is always an I-frame, for the reference. It can be observed in Figure 4.11, that primary SP-frames have slightly lower coding efficiency than P-frames and significantly higher coding efficiency than I-frames, above 2 dB. Therefore, periodic insertion of I-frames is not appropriate for viewpoint switching in MVV.

In Figure 4.10, there are 7 secondary SP-frames to be stored in the server at each switching point in order to support switching among 8 views. The benefits of adopting the proposed scheme for coding secondary SP-frames is to reduce the bits required for secondary SP-frames. As a consequence, the storage requirement of the video server can be reduced. Table 4.3 to Table 4.7 summarize the average sizes of secondary SP-frames with different $Q_P$ for the test sequences “Ballroom”, “Exit”, “Vassar”, “Race1”, and “Breakdancers”, respectively. The sizes of different secondary SP-frames between two views such as view 1 to view 2 (1→2), view 2 to view 3(2→3), etc are also shown. In this example, switching can be only in one direction. Each row contains the average bit-counts of encoding SP-frames using the pixel-domain and QDCT-domain motion estimation techniques that need to be stored in the server, and the percentage reduction of our proposed scheme over the scheme in [77] at different switching viewpoints. The last row of each table shows the total storage requirement of secondary SP-frames in the server by using different schemes. From these tables, it can easily be observed that the required storage in the server of the proposed scheme is much fewer than that of the
scheme in [77], which provides up to 3%, 5%, 10%, 6% and 5% savings in “Ballroom”, “Exit”, “Vassar”, “Race1”, and “Breakdancers” respectively. Again, the merit of proposed scheme is mainly due to performing motion estimation in the QDCT domain, which obtains more desirable motion vectors for entropy coding of secondary SP-frames. Moreover, in a multi-view system, switching viewpoints may be necessary to come across all the in-between views. Implementing SP-frames can achieve drift-free and low-delay switching among views. With our proposed scheme, the total sizes of switching SP-frames can then be reduced and this is also favourable to the bandwidth requirements of multi-view streaming.
Table 4.3. Average bit-counts of different secondary SP-frames for “Ballroom” by applying the scheme in [77] and the proposed scheme.

<table>
<thead>
<tr>
<th>Qp</th>
<th>16</th>
<th>20</th>
<th>24</th>
<th>28</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Scheme in [77] [kbits]</td>
<td>Proposed scheme [kbits]</td>
<td>Δ [%]</td>
<td>Scheme in [77] [kbits]</td>
</tr>
<tr>
<td>0 -&gt; 1</td>
<td>958</td>
<td>947</td>
<td>1.1566</td>
<td>710</td>
</tr>
<tr>
<td>1 -&gt; 2</td>
<td>897</td>
<td>884</td>
<td>1.4656</td>
<td>648</td>
</tr>
<tr>
<td>2 -&gt; 3</td>
<td>895</td>
<td>886</td>
<td>1.0328</td>
<td>645</td>
</tr>
<tr>
<td>3 -&gt; 4</td>
<td>905</td>
<td>896</td>
<td>1.0246</td>
<td>661</td>
</tr>
<tr>
<td>4 -&gt; 5</td>
<td>882</td>
<td>868</td>
<td>1.5868</td>
<td>633</td>
</tr>
<tr>
<td>5 -&gt; 6</td>
<td>894</td>
<td>885</td>
<td>1.0013</td>
<td>652</td>
</tr>
<tr>
<td>6 -&gt; 7</td>
<td>929</td>
<td>919</td>
<td>1.1572</td>
<td>690</td>
</tr>
<tr>
<td>Total size</td>
<td>6360</td>
<td>6283</td>
<td>1.2019</td>
<td>4639</td>
</tr>
</tbody>
</table>

Table 4.4. Average bit-counts of different secondary SP-frames for “Exit” by applying the scheme in [77] and the proposed scheme.

<table>
<thead>
<tr>
<th>Qp</th>
<th>16</th>
<th>20</th>
<th>24</th>
<th>28</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Scheme in [77] [kbits]</td>
<td>Proposed scheme [kbits]</td>
<td>Δ [%]</td>
<td>Scheme in [77] [kbits]</td>
</tr>
<tr>
<td>0 -&gt; 1</td>
<td>821</td>
<td>804</td>
<td>2.0614</td>
<td>494</td>
</tr>
<tr>
<td>1 -&gt; 2</td>
<td>794</td>
<td>779</td>
<td>1.8081</td>
<td>482</td>
</tr>
<tr>
<td>2 -&gt; 3</td>
<td>789</td>
<td>775</td>
<td>1.7182</td>
<td>484</td>
</tr>
<tr>
<td>3 -&gt; 4</td>
<td>785</td>
<td>777</td>
<td>1.0732</td>
<td>501</td>
</tr>
<tr>
<td>4 -&gt; 5</td>
<td>797</td>
<td>777</td>
<td>2.4736</td>
<td>503</td>
</tr>
<tr>
<td>5 -&gt; 6</td>
<td>844</td>
<td>803</td>
<td>4.8385</td>
<td>974</td>
</tr>
<tr>
<td>6 -&gt; 7</td>
<td>836</td>
<td>816</td>
<td>2.3772</td>
<td>575</td>
</tr>
<tr>
<td>Total size</td>
<td>5665</td>
<td>5532</td>
<td>2.3593</td>
<td>4013</td>
</tr>
</tbody>
</table>
Table 4.5. Average bit-counts of different secondary SP-frames for “Vassar” by applying the scheme in [77] and the proposed scheme.

<table>
<thead>
<tr>
<th>Qp</th>
<th>Scheme in [77] [kbits]</th>
<th>Proposed scheme [kbits]</th>
<th>Δ [%]</th>
<th>Scheme in [77] [kbits]</th>
<th>Proposed scheme [kbits]</th>
<th>Δ [%]</th>
<th>Scheme in [77] [kbits]</th>
<th>Proposed scheme [kbits]</th>
<th>Δ [%]</th>
<th>Scheme in [77] [kbits]</th>
<th>Proposed scheme [kbits]</th>
<th>Δ [%]</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 -&gt; 1</td>
<td>934</td>
<td>930</td>
<td>0.4692</td>
<td>0.6166</td>
<td>402</td>
<td>392</td>
<td>2.5007</td>
<td>237</td>
<td>232</td>
<td>1.9908</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1 -&gt; 2</td>
<td>921</td>
<td>907</td>
<td>1.4547</td>
<td>1.5014</td>
<td>384</td>
<td>373</td>
<td>2.8764</td>
<td>233</td>
<td>228</td>
<td>2.1639</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2 -&gt; 3</td>
<td>933</td>
<td>925</td>
<td>0.7754</td>
<td>1.7555</td>
<td>395</td>
<td>383</td>
<td>2.8361</td>
<td>257</td>
<td>237</td>
<td>7.7057</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3 -&gt; 4</td>
<td>920</td>
<td>918</td>
<td>0.2328</td>
<td>1.0727</td>
<td>404</td>
<td>392</td>
<td>3.0209</td>
<td>243</td>
<td>240</td>
<td>1.0663</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4 -&gt; 5</td>
<td>902</td>
<td>897</td>
<td>0.6272</td>
<td>2.2440</td>
<td>388</td>
<td>372</td>
<td>4.2636</td>
<td>252</td>
<td>233</td>
<td>7.3849</td>
<td></td>
<td></td>
</tr>
<tr>
<td>6 -&gt; 7</td>
<td>990</td>
<td>975</td>
<td>1.5794</td>
<td>2.0984</td>
<td>464</td>
<td>447</td>
<td>3.6348</td>
<td>326329</td>
<td>294895</td>
<td>9.6325</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Total size</td>
<td>6954</td>
<td>6759</td>
<td>2.8003</td>
<td>3.9004</td>
<td>3000</td>
<td>2846</td>
<td>5.1333</td>
<td>327921</td>
<td>296386</td>
<td>9.6166</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 4.6. Average bit-counts of different secondary SP-frames for “Race1” by applying the scheme in [77] and the proposed scheme.

<table>
<thead>
<tr>
<th>Qp</th>
<th>Scheme in [77] [kbits]</th>
<th>Proposed scheme [kbits]</th>
<th>Δ [%]</th>
<th>Scheme in [77] [kbits]</th>
<th>Proposed scheme [kbits]</th>
<th>Δ [%]</th>
<th>Scheme in [77] [kbits]</th>
<th>Proposed scheme [kbits]</th>
<th>Δ [%]</th>
<th>Scheme in [77] [kbits]</th>
<th>Proposed scheme [kbits]</th>
<th>Δ [%]</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 -&gt; 1</td>
<td>759</td>
<td>748</td>
<td>1.4538</td>
<td>2.5655</td>
<td>359</td>
<td>349</td>
<td>2.7901</td>
<td>255</td>
<td>246</td>
<td>3.6990</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1 -&gt; 2</td>
<td>765</td>
<td>748</td>
<td>2.2425</td>
<td>3.2461</td>
<td>356</td>
<td>349</td>
<td>1.9133</td>
<td>248</td>
<td>244</td>
<td>1.6617</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2 -&gt; 3</td>
<td>823</td>
<td>799</td>
<td>2.9566</td>
<td>5.7215</td>
<td>405</td>
<td>391</td>
<td>3.3618</td>
<td>276</td>
<td>264</td>
<td>4.4027</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3 -&gt; 4</td>
<td>826</td>
<td>796</td>
<td>3.6595</td>
<td>4.7872</td>
<td>391</td>
<td>368</td>
<td>5.7625</td>
<td>269</td>
<td>256</td>
<td>4.8631</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4 -&gt; 5</td>
<td>741</td>
<td>717</td>
<td>3.2204</td>
<td>4.1243</td>
<td>338</td>
<td>325</td>
<td>3.8777</td>
<td>239</td>
<td>231</td>
<td>3.2857</td>
<td></td>
<td></td>
</tr>
<tr>
<td>6 -&gt; 7</td>
<td>818</td>
<td>802</td>
<td>1.9179</td>
<td>2.4605</td>
<td>376</td>
<td>367</td>
<td>2.4119</td>
<td>254</td>
<td>251</td>
<td>1.1549</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Total size</td>
<td>6235</td>
<td>5934</td>
<td>4.8400</td>
<td>6.1344</td>
<td>2891</td>
<td>2724</td>
<td>5.7697</td>
<td>1977</td>
<td>1864</td>
<td>5.7466</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Table 4.7. Average bit-counts of different secondary SP-frames for “Breakdancers” by applying the scheme in [77] and the proposed scheme.

<table>
<thead>
<tr>
<th>Qp</th>
<th>16</th>
<th>20</th>
<th>24</th>
<th>28</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>2034</td>
<td>1098</td>
<td>559</td>
<td>322</td>
</tr>
<tr>
<td>1</td>
<td>2004</td>
<td>1056</td>
<td>546</td>
<td>291</td>
</tr>
<tr>
<td>2</td>
<td>1980</td>
<td>1036</td>
<td>523</td>
<td>283</td>
</tr>
<tr>
<td>3</td>
<td>1938</td>
<td>1022</td>
<td>517</td>
<td>283</td>
</tr>
<tr>
<td>4</td>
<td>1938</td>
<td>1002</td>
<td>533</td>
<td>294</td>
</tr>
<tr>
<td>5</td>
<td>1990</td>
<td>1053</td>
<td>492</td>
<td>288</td>
</tr>
<tr>
<td>6</td>
<td>2034</td>
<td>1118</td>
<td>537</td>
<td>327</td>
</tr>
<tr>
<td>Total size</td>
<td>13968</td>
<td>7385</td>
<td>3707</td>
<td>2088</td>
</tr>
</tbody>
</table>

4.6 Chapter summary

A further analysis of viewpoint switching of multi-view video using SP-frames has been made in this chapter. From our investigation, the problem of the increase in the size of secondary SP-frames is more severe in multi-view video in which the correlation between the two views becomes smaller. For this reason, we also have incorporated the proposed QDCT-domain motion estimation technique in secondary SP-frame coding so as to provide the functionality in free navigation among various views. By employing the new technique which carries out motion estimation in the QDCT domain for coding of secondary SP-frames, experimental results demonstrated that our proposed scheme overwhelmingly outperforms the conventional pixel-domain motion estimation technique. As a consequence, the size of secondary SP-frames can be reduced remarkably, especially in multi-view video and single-view video with complex motion. It means that the proposed scheme is particularly useful...
to low-delay viewpoint switching in the multi-view video system in which users will be more interested in switching their views during an exciting event. An exciting event is always in the scene consisting of complex motion and inter-view disparity, and the correlation between the two views becomes lower. In this scenario, the proposed scheme is beneficial to the entropy coding of secondary SP-frames.
Chapter 5 Frame-Based Domain Selection in Motion Estimation of Coding Secondary SP-frames

5.1 Introduction

The key to reduce the size of secondary SP-frames in Chapters 3 and 4 lies in the quantized-transform domain motion estimation. However, there are still some exceptional cases that the coding performance of the quantized-transform domain technique is not as good as that of the pixel-domain technique. To avoid this, both pixel-domain and QDCT-domain motion estimation techniques should be considered. In this chapter, to further minimize the size of secondary SP-frames and reduce the computational complexity, a hybrid motion estimation algorithm combining two motion estimation techniques is proposed. The combination is based on inter-frame correlation, which is measured using the bit-count of its corresponding primary SP-frame.

Part of the contents of this chapter have been published in reference [81].
5.2 Impact of using QDCT-ME on secondary SP-frame coding

In Chapters 3 and 4, we have proposed the quantized-transform domain motion estimation technique for coding secondary SP-frames in multiple bitrate and multiple viewpoint videos. Simulation results showed that some secondary SP-frames generated by the QDCT-domain motion estimation (QDCT-ME) technique induce more bits than the pixel-domain motion estimation (pixel-ME) technique, as illustrated in Figure 5.1 where the test video stream used for simulation is the "Crew" sequence. The "Crew" sequence was encoded into two bitstreams with $Q_p$ being equal to 20 and 28, and $Q_s$ was set to $Q_p - 6$, i.e. 14 and 22 respectively. Switching was then taken place from the bitstream with $Q_p=20$ to the bitstream with $Q_p=28$. This phenomenon can be explained by the distribution of quantized-transform coefficients in some $4 \times 4$ blocks where all pixels are similar or with the same value. In these blocks, their quantized-transform coefficients are mainly zeros except the DC coefficient. If this type of QDCT-domain blocks is used to search over a predetermined search area on the quantized-transform coefficients of the reference frame, all the SAQTD values are very similar and the motion estimation process becomes very sensitive to noise. In this case, the smallest SAQTD value may not guarantee to secure the best motion vector, and then QDCT-ME introduces more bit-counts.
Figure 5.1 Frame-by-frame comparisons of size reduction of secondary SP-frames in percentage achieved by QDCT-ME over pixel-ME in the "Crew" sequence.

Another drawback of QDCT-ME is the surge in computational complexity. For QDCT-ME, each current block is transformed and quantized. A search window in the reference frame centred on the current block position is set. QDCT-ME starts with transforming and quantizing a block in the top right-hand corner of the search window. After transforming and quantizing current and reference blocks, $SAQTD$ can then be calculated for this candidate. Afterward, the next candidate is another block shifted by 1 pixel in horizontal direction, and this block is also transformed and quantized for computing its $SAQTD$. These procedures continue for all possible candidates within the search window. It means that all possible candidates within the search window also need to be transformed and quantized in QDCT domain. Consequently, $SAQTD$ is computationally very intensive though it can achieve higher coding efficiency as compared with SAD.
5.3 Proposed hybrid motion estimation techniques

![Diagram showing bitstream witching using primary and secondary SP-frames]

Figure 5.2. Bitstream witching using primary and secondary SP-frames

The aforementioned drawbacks motivate us to adopt a hybrid approach to reduce the computational complexity and further improve the coding efficiency of the QDCT-ME. In the proposed hybrid scheme, a selection mechanism based on inter-frame correlation between the current frame and the reference frame is adopted to choose between the use of pixel-domain and QDCT-domain techniques. One straightforward approach is to perform pixel-domain motion estimation and QDCT-domain motion estimation separately. This frame-based approach compares the sizes of secondary SP-frames generated by both estimation techniques and chooses the set of final motion vectors with a smaller bit-count. Nonetheless, it demands huge computational complexity. To provide higher coding efficiency and lower computational complexity, a selection algorithm is required to determine the use of an appropriate domain prior to encoding secondary SP-frames.
A hybrid motion estimation algorithm is proposed to make good use of high coding efficiency of the QDCT-domain algorithm and low computational complexity of the pixel-domain algorithm. The hybrid algorithm is based on inter-frame correlation between the current frame and the reference frame. In Chapter 4, we revealed that the improvement in coding efficiency of using QDCT-ME highly relies on the degree of inter-frame correlation between \( SP_{2,t} \) and \( P_{1,t-1} \) as shown in Figure 5.2, and it is denoted by \( corr_{SSP,t} \). The weaker the inter-frame correlation is, the better the coding efficiency of SAQTD can provide. Therefore, \( corr_{SSP,t} \) is a good measure to determine a proper domain for motion estimation in secondary SP-frame coding. In other words, a smaller value of \( corr_{SSP,t} \) tends to use QDCT-ME; otherwise, pixel-ME is employed to reduce the required computational complexity.

![Figure 5.3. Sizes of secondary SP-frames and the corresponding primary SP-frames of “Mobisode2”.](image-url)
The number of bits required for encoding $SP_{12,t}$ can be used directly as a measure of $corr_{SSP,t}$. However, this bit-count of $SP_{12,t}$ cannot be obtained prior to secondary SP-frame encoding. Figure 5.3 then shows the bit counts for $SP_{12,t}$ and $SP_{1,t}$ of “Mobisode2”. It is noted that the primary SP-frame is available when its corresponding secondary SP-frame is encoded. From this Figure, it can be easily seen that the general trends of the two curves are very similar. It is due to the fact that the current frames used for encoding $SP_{12,t}$ and $SP_{1,t}$ are the frames at time $t$ from $B2$ and $B1$, respectively. They share the same video content and the only discrepancy is the quantization parameters. It implies that $corr_{SSP,t}$ is reasonably approximated by the bit-count of $SP_{1,t}$, denoted by $BC_{PSP}$. $BC_{PSP}$ is then used to select a proper domain for motion estimation in secondary SP-frame coding with the proposed hybrid algorithm. When $BC_{PSP}$ is larger than a predefined threshold $Th$, $corr_{SSP,t}$ becomes lower and QDCT-ME is carried out for secondary SP-frame coding in order to offer higher coding efficiency. In contrast, pixel-ME is good enough when $BC_{PSP} \leq Th$, which can relieve the computational burden of secondary SP-frame coding. In general, more frames in a sequence with complex or fast motion activities will be encoded using QDCT-ME. In our implementation, $Th$ is set to $\alpha C$, where $C$ is a universal constant with a unit of bits and $\alpha$ is a scaling factor. The constant $C$ can be applied into all video sequences and it is equal to 50,000. Besides, $\alpha$ is sequence-dependent, and it is also considered as a parameter to control the trade-off between the computational complexity and the accuracy of domain selection of the proposed hybrid algorithm. Typically $\alpha$ is inversely proportional to the motion activity of a video sequence.
5.4 Experimental results of using hybrid motion estimation algorithm

The proposed hybrid algorithm has been implemented based on the JVT JM 11 encoder [13]. The luminance components of the first 100 frames of “Mobisode1” (832×480), “Mobisode2” (832×480) and “Crew” (1280×720) were used. For motion estimation, a search range of 32 was set for both P-frames and SP-frames. The sequences were encoded into two bitstreams with two different quantization levels (Qp=24 and 20). For the proposed algorithm, by taking into consideration of motion activity, $\alpha$ was set to 1 and 10 for Mobisode1/Mobisode2 and Crew, respectively.
Figure 5.4. Sizes of secondary SP-frames by applying the pixel-ME, QDCT-ME and proposed hybrid algorithms with Qp=24 to Qp=20 switching in (a) “Crew”, (b) “Mobisode1”, (c) “Mobisode2”.

82
Figure 5.4 depicts the switching scenarios of “Crew”, “Mobisode1” and “Mobisode2” performed from $Qp=24$ to $Qp=20$. Only the first frames were encoded as I-frames and all other frames were encoded in turn as SP-frames while non-switching frames were encoded as P-frames. The figure shows the number of required bits of all possible secondary SP-frames by employing the pixel-ME algorithm, the QDCT-ME algorithm and the proposed hybrid algorithm. The required bit-counts of the proposed algorithm are very similar to those of the QDCT-ME algorithm, but it requires less bit-counts as compared with the pixel-ME algorithm.

<table>
<thead>
<tr>
<th></th>
<th>Δsize</th>
<th>Δtime</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mobisode1</td>
<td>+0.25 %</td>
<td>-9.69%</td>
</tr>
<tr>
<td>Mobisode2</td>
<td>+0.85 %</td>
<td>-32.77%</td>
</tr>
<tr>
<td>Crew</td>
<td>+0.630 %</td>
<td>-41.92%</td>
</tr>
</tbody>
</table>

Besides, Table 5.1 further illustrates the performance of the proposed hybrid algorithm when compared to the QDCT-ME in the three testing sequences. In the table, $\Delta$bit-count and $\Delta$time represent the percentage changes of the bit-count and encoding time of secondary SP-frames respectively. The positive values mean increments whereas negative values decrements. Again, the table shows that the bit-count increase of the proposed algorithm is negligible. However, it is observed that the proposed algorithm can substantially reduce the computational complexity as compared with the QDCT-ME algorithm.
5.5 Chapter summary

In this chapter, a hybrid motion estimation algorithm for secondary SP-frame coding has been proposed, which adaptively selects QDCT or pixel domain for motion estimation according to the bit-count of its corresponding primary SP-frame. The hybrid algorithm has the advantages of utilizing both domains in encoding secondary SP-frames. All the points related to our proposed scheme have been verified experimentally. Experimental results showed that the proposed new measure for selecting an appropriate domain is effective. As compared with the QDCT-domain algorithm, the proposed algorithm can achieve remarkable computational savings while maintaining a similar size of a secondary SP-frame. As a concluding remark, the hybrid motion estimation algorithm in this chapter can strengthen the coding efficiency secondary SP-frames.
Chapter 6 Macroblock-Based Motion Estimation Selection Algorithm in Bitstream Switching using Flexible Macroblock Ordering

6.1 Introduction

By exploring the inter-frame correlation between the reference frame and the primary SP-frame being switched to, we have discussed the frame-based scheme in Chapter 5 that combining pixel-domain motion estimation and quantized-transform domain motion estimation techniques. This hybrid scheme can significantly reduce the computational complexity of coding secondary SP-frames. It can also maintain the size of secondary SP-frames. However, it suffers from one drawback. Since the hybrid scheme is in frame-based, correlation of macroblocks is not considered. As a result, the size of secondary SP-frames cannot be further reduced. In this chapter, the combination of pixel-ME and QDCT-ME is based on a new measurement of inter-frame correlation by using the bit-counts of the macroblocks in SP-frames, so that the hybrid scheme is dominated by employing QDCT-ME in the macroblocks with weaker inter-frame correlation; otherwise, it approaches pixel-ME.
However, mixing these two kinds of motion vectors might increase the size of a secondary SP-frame. It is because there is less correlation among motion vectors obtained by two different domains. With the further help of the explicit mode in Flexible Macroblock Ordering (FMO), the proposed hybrid scheme classifies macroblocks into two slice groups by examining the domain used in motion estimation prior to coding motion vectors in a secondary SP-frame. The slice structure of a secondary SP-frame using the explicit FMO mode is flexible and can be changed during the encoding of each new frame. The new scheme can further enhance the coding efficiency and computational complexity of secondary SP-frames.

6.2 Proposed scheme using Flexible Macroblock Ordering

6.2.1 MB-based hybrid motion estimation scheme

The disadvantage of the frame-based approach in Chapter 5 is that spatial characteristics within a frame are not considered. For instance, a 4×4 block in homogeneous area causes more quantized-transform coefficients to be zero in which QDCT-ME is no longer suitable. In contrast, choosing pixel-ME might give a lower-energy residual after motion compensation in this situation. In general, pixel-ME is appropriate for homogeneous areas of the frame and QDCT-ME is beneficial to detailed areas.

By taking this into consideration, the scheme proposed in this chapter are operated at macroblock (MB) level. Similar to the frame-based approach, two independent motion estimation techniques operated in pixel and QDCT
domains for each MB are carried out. Then two bit-counts for an MB can be
given and the minimum one, associated with a motion vector, is chosen. This
motion vector is considered as the best one. However, this brute-force
approach will increase the encoding time drastically.

The results obtained in Chapter 4 disclosed that the improvement in coding
efficiency of a secondary SP-frame from QDCT-ME greatly depends on the
degree of inter-frame correlation between the MB in $SP_{2,t}$ and its motion
compensation MB in $P_{1,t-1}$ as shown in Figure 6.1. In the proposed MB-based
approach, the correlation is measured at MB level and is represented by
$corr_{SSPMB,t}$. Similar to the frame-based approach, the weaker the correlation is,
the better the coding efficiency of QDCT-ME can achieve. As a consequence,
$corr_{SSPMB,t}$ can be used as a measure to select an appropriate domain for
motion estimation in coding an MB in a secondary SP-frame. Accordingly,
QDCT-ME is more suitable for a smaller value of $corr_{SSPMB,t}$, otherwise, pixel-
domain motion estimation is appropriate to reduce the required computational
complexity and maintain sufficient coding efficiency for MBs in homogeneous areas. The bit-count required for the MB in $SP_{12,t}$ can be employed as a measure of $corr_{SSPMB,t}$. Nevertheless, this bit-count cannot be obtained prior to secondary SP-frame coding.

Figure 6.2 then shows the bit-counts of all MBs in two rows of $SP_{12,t}$ and $SP_{1,t}$ in the 34th frame of the “Crew” sequence. It is noted that the MB of the primary SP-frame has already been available when its corresponding MB in the secondary SP-frame is encoded. From this Figure, it can be observed that the general trends of the two curves in MB level are very similar. It is because the current frames used for encoding $SP_{12,t}$ and $SP_{1,t}$ are the frames at time $t$ from B2 and B1, respectively. $corr_{SSPMB,t}$ can then be approximated by the bit-count.
Chapter 6 Macroblock-Based Motion Estimation Selection Algorithm in Bitstream Switching using Flexible Macroblock Ordering

of the MB in the primary SP-frame at time $t$, $SP_{1,t}$, denoted by $bit\text{-}count_{PSPMB}$. It is then used to choose the use of pixel-MEn or QDCT-ME in secondary SP-frame coding with the proposed MB-based scheme. When $bit\text{-}count_{PSPMB}$ is larger than a predefined threshold $TH$, $corr_{SSPMB,t}$ becomes lower and QDCT-ME is performed to provide higher coding efficiency. On the other hand, pixel-domain motion estimation is employed when $bit\text{-}count_{PSPMB} \leq TH$, which can offer less computational requirement of secondary SP-frame coding as well as providing better coding efficiency of MBs within the homogeneous area. In general, more MBs in a frame with complex spatial or temporal activities will be encoded using QDCT-ME. To determine $TH$, arithmetic mean for each frame with $M\times N$ MBs is used as a tool to derive the central tendency of bits in these MBs. $TH$ can then be formulated as

$$TH = \frac{1}{M \times N} \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} bit\text{-}count_{PSPMB(i,j)}$$

(6.1)

where $bit\text{-}count_{PSPMB(i,j)}$ is the bit-count needed to be encoded for the MB at the $i^{th}$ row and $j^{th}$ column of a primary SP-frame.

6.2.2 Utilization of FMO in the proposed hybrid Scheme

The proposed hybrid scheme estimates motion vectors obtained from two kinds of domains - pixel domain or QDCT domain. Unsystematically mixing these two kinds of motion vectors might increase the bit-count required by a secondary SP-frame. It is due to the adoption of motion vector prediction where motion vectors are coded differentially with respect to a predictor of the motion vectors.
from the previous coded blocks. This predictor is computed with consideration of the adjacent blocks and their motion tends to have very high spatial correlation. The thing is, the motion vector predictor is formed based on the median value of the motion vectors of the three adjacent blocks, on the left, top, and top-right (or top-left if top-right is not available). However, considering that the proposed hybrid scheme allows the use of different domains for motion estimation in secondary SP-frame coding, the motion vector of the MB with fewer bit-count obtained by ME in one domain will contribute to the motion vector predictor for the adjacent MB, no matter which domain is used in motion estimation [82-83]. Motion vectors obtained from different domains are spatially less correlated, resulting in increase of bit-counts of secondary SP-frames. It is noted that, in H.264 video, slice is a basic structure in a picture, and each picture can be subdivided into one or more slices. A sequence of MBs is defined in these slices. These slices segment a picture into different partitions and they are coded independently of each other. In the typical picture setting of an H.264 encoded video bitstream, only one slice is always used in a picture. In other words, even though two neighbouring MBs in a slice using different motion estimation techniques are still dependent on each other, mixing two kinds of motion vectors in a frame is not good for coding efficiency. It means that only one slice per frame is not very applicable to the proposed hybrid scheme. This problem can be solved by assigning only one MB per slice. Yet it is not a practical way due to the excessive bits required by the headers in these slices.
In the proposed hybrid scheme, an additional consideration is made based on the new MB classification policy by utilizing Flexible Macroblock Ordering (FMO) [84]. FMO is one of the most striking error resilience tools supported by H.264. It specifies a pattern that assigns MBs in a picture to one or several slice groups, and provides more flexible way of macroblock grouping. Each MB can be assigned into a slice group through a Macroblock-to-slice Allocation map (MBAmap) [84-86]. In this map, each MB is identified with a number that indicates which slice group the MB belongs to. The MBAmap can be updated each frame by using the Picture Parameter Set (PPS). By using this mechanism, FMO can divide a picture with different patterns of MBs. There are seven different types of slice group maps for FMO in the standard [16]. Figure 6.3 depicts six predefined slice group maps from type 0 to type 5: interleaved, dispersed, foreground and background, box-out, raster scan, and wipe. In
addition, there is an explicit mode (type 6), which is also the most flexible FMO type. This explicit mode allows users to define their own MBAmap such that MBs in a picture can be assigned to any slice group in any order.

In the proposed hybrid scheme, motion estimation is carried out either in pixel domain or QDCT domain. According to the domain selection in Section 6.2.1, there are two slice groups in each secondary SP-frame, and MBs are classified by examining the bit-counts of the primary SP-frame prior to the motion estimation and motion vector coding in a secondary SP-frame. The six predefined FMO slice groups in Figure 6.3 are not flexible enough to fulfill the classification of MBs since none of the patterns in the predefined slice groups can fit for all frames. Apart from the predefined patterns, fully flexible macroblock ordering in the explicit mode is used in the proposed scheme. By doing so, each secondary SP-frame changes dynamically the MB classification throughout the entire video sequence. The provision of dynamic formation of slice groups in every secondary SP-frame is exploited by the bit-counts of MBs in the primary SP-frame. Then each MB in the corresponding secondary SP-frame is classified either in pixel or QDCT domain. Specifically, an identification number is given to each MB. The MBs which are identified as using pixel-ME are assigned as slice group 0 while the MBs which are classified as using QDCT-ME are assigned as slice group 1. After that, an MBAmap is established. This map partitions the secondary SP-frame into two slice groups. During encoding, only MBs in the same slice depend on each other. It means that motion vector predictors, which are the motion vectors computed through motion estimation in the same domain, are from the same slice only. On the
whole, FMO partitions a secondary SP-frame into two slice groups referring to the MBAmap, which describes the use of the proper domain in motion estimation of each MB.

6.3 Experimental results

A large amount of experimental work has been conducted to evaluate the performances of the proposed MB-based scheme for bitstream switching using SP-frames. Results in terms of both coding efficiency and computational complexity were compared with those obtained using the pixel-domain and QDCT-domain schemes. Let us denote them as pixel-ME [13] and QDCT-ME [67], respectively. All schemes were implemented based on the H.264 reference codec (JM version 11.0) [13] for secondary SP-frame coding. Four test sequences with 1280x720 resolution, “Crew”, “Riverbed”, “Ducktakeoff” and “Shuttlestart”, were used for performance comparison. In each sequence, 100 frames were encoded with two QP to generate two bitstreams with different bit rates. Switching between two bitstreams in both directions was then performed. For the high bit rate bitstream, the quantization parameter QP was fixed at 20. On the other hand, QP varied from 24 to 32 with a step-size of 4 for coding low bit rate bitstream. According to the optimal setting in [14], QS was set to QP − 6, i.e. from 18 to 26. Only the first frames of the bitstreams were encoded as I-frames, and switching frames were encoded in turn as SP-frames while all the rest non-switching frames were encoded as P-frames. In our experiments, extended profile with CAVLC entropy encoding was used to configure the encoder. R-D optimization was enabled. For the motion
estimation process, a search range of 32 was set for both P-frames and SP-frames. For the quantized-transform motion estimation, the scaling factor $k$ of equation (3) was set to 3, which was found by experimental observations [67]. To make the comparison impartial, both schemes employed a full search algorithm for motion estimation.

Figure 6.4. Frame-by-frame size reduction of secondary SP-frames in percentage achieved by the proposed MB-based scheme and QDCT-ME over pixel-ME for the switching-down scenario.

(a) Crew, (b) Riverbed, (c) Duckstakeoff, and (d) Shuttlestart.
Chapter 6 Macroblock-Based Motion Estimation Selection Algorithm in Bitstream Switching using Flexible Macroblock Ordering

We aim at evaluating the coding efficiency of secondary SP-frames based on our proposed MB-based scheme. To have a comparison among the proposed hybrid scheme, QDCT-ME and pixel-ME, we did exhaustive simulation on the bit-counts of all possible secondary SP-frames. Figure 6.4(a)-(d) and Figure 6.5(a)-(d) show a frame-by-frame comparison of size reduction in secondary SP-frames for different sequences in both switching directions. Figure 6.4(a)-(d) demonstrate high-to-low bit rate switching (Qp = 20 to Qp = 28) while Figure 6.5(a)-(d) depict low-to-high bit rate switching (Qp = 28 to Qp = 20). In these figures, the values of the Y-axis mean the average size reduction of secondary SP-frames in percentage of our proposed MB-based scheme and the QDCT-ME.
ME scheme over the pixel-ME scheme. The positive value indicates the tested schemes generate less bit-count as compared with the pixel-ME whereas the negative value indicates the tested schemes require more bit-count as compared with the pixel-ME. In the high-to-low bit rate switching (switching down) scenario as shown in Figure 6.4(a)-(d), the MB-based scheme can substantially reduce the size of secondary SP-frames, up to 43%, 12%, 6% and 4.3% in “Crew”, “Riverbed”, “Ducktakeoff” and “Shuttlestart”, respectively, comparing to the traditional pixel-ME scheme. Similarly, in the low-to-high bit rate switching (switching up) case as illustrated in Figure 6.5, size reduction of secondary SP-frames using the MB-based scheme is also very significant, up to 24%, 7%, 3.5% and 13% in “Crew”, “Riverbed”, “Ducktakeoff” and “Shuttlestart”, respectively. From Figure 6.4 and Figure 6.5, it can be seen that our MB-based scheme can outperform the QDCT-ME in all secondary SP-frames for all sequences even though QDCT-ME can also have remarkable size reduction over the pixel-ME. The significant improvement of the proposed scheme is due to the flexibility of performing motion estimation and compensation in both pixel and QDCT domains. The process of selecting an appropriate domain for motion estimation can prevent the case when the MB has similar pixel values in which the motion estimation operated in pixel domain generates fewer bits than QDCT-ME, as explained in Section 6.3.1. The evidence of this phenomenon is also depicted in Figure 6.4 and Figure 6.5 where a number of secondary SP-frames of the test sequences generated by using the QDCT-ME introduce more bits in comparison with the pixel-ME. The proposed MB-based algorithm effectively combines two existing techniques at MB level during the encoding of each SP-frame. The combination is facilitated...
by the flexible explicit mode in FMO, and is then controlled by the bit-count from the corresponding MB of the primary SP-frame. By making use of this arrangement, the overwhelming reduction in Figure 6.4 and Figure 6.5 of the proposed MB-based scheme highlights the importance of using a proper domain at MB level for motion estimation in secondary SP-frame coding.

Figure 6.6. Size reduction of secondary SP-frames in percentage achieved by the proposed MB-based scheme and QDCT-ME over pixel-ME with different $Qp$ for the switching-down scenario.

(a) Crew, (b) Riverbed, (c) Duckstageoff, and (d) Shuttlestart.
Figure 6.7. Size reduction of secondary SP-frames in percentage achieved by FMO and QDCT-ME over pixel-ME with different Qp for the switching-up scenario. (a) Crew, (b) Riverbed, (c) Duckstakeoff, and (d) Shuttlestart.

By the same token, Figure 6.6(a)-(d) and Figure 6.7(a)-(d) show the average percentage reduction in size of secondary SP-frames with different Qp for switching-down and switching-up scenarios, respectively. Note that the quantization parameters were varied from 24 to 32 with a step-size of 4 and the switching processes were from and to Qp = 20. It is obvious from Figure 6.6(a)-(d) and Figure 6.7(a)-(d) that, by using the proposed MB-based scheme, the size of secondary SP-frames can be remarkably reduced for various quantization parameters. Moreover, it is interesting to note that the improvement of the MB-based scheme over QDCT-ME in “Crew” is more
outstanding than that in other sequences, as shown in Figure 6.6(a) and Figure 6.7(a).

Figure 6.8. Different areas using QDCT-ME and Pixel-ME.

Figure 6.8 shows one frame of the “Crew” sequence, which contains NASA crew leaving a building with flashlights. In this sequence, it includes scenes with challenging motion which is suitable for QDCT-ME. On the other hand, the MBs located at the wall, as highlighted in Figure 6.8, has inferior performance for QDCT-ME, as the pixels in this area have similar values. However, the results in Figure 6.6(a) and Figure 6.7(a) show that the MB-based scheme can make good use of pixel-ME and QDCT-ME, especially in this type of sequences.

To compare the computational complexity required by various schemes, the average encoding time of secondary SP-frames with different Qp were
measured and tabulated in Table 6.1 and Table 6.2 for switching-down and switching-up scenarios, respectively. All the simulations were carried out on a PC with an Intel Core™2 Quad Q9450 CPU at 2.66GHz and 12GB memory. We also demonstrate the savings of the proposed hybrid scheme over the QDCT-ME scheme in Table 6.1 and Table 6.2. \( \Delta Time \) in these tables represents the percentage change of the average encoding time of the hybrid scheme over the QDCT-ME and it is calculated as follows

\[
\Delta Time(\%) = \frac{Time_{MB-based} - Time_{QDCT-ME}}{Time_{QDCT-ME}} \times 100
\]  

(6)

where \( Time_{QDCT-ME} \) and \( Time_{MB-based} \) denote the encoding time used by the QDCT-ME scheme and the proposed MB-based scheme respectively. Owing to selecting an appropriate domain for motion estimation in secondary SP-frame coding, not all the blocks need to be encoded using QDCT-ME. As a result, it can be easily seen that the proposed MB-based scheme can substantially reduce the computational complexity of QDCT-ME up to 56%, 57%, 47% and 49% for the switching-down scenario and 48%, 49%, 43% and 55% for the switching-up scenario in “Crew”, “Riverbed”, “Ducktakeoff” and “Shuttlestart”, respectively, as shown in Table 6.1 and Table 6.2. Not only encoding time can be saved, but our proposed MB-based scheme can also reduce greatly the bit-counts of secondary SP-frames.
Chapter 6 Macroblock-Based Motion Estimation Selection Algorithm in Bitstream Switching using Flexible Macroblock Ordering

Table 6.1. Average time usage of secondary SP-frames with different $Qp$ for the switching-down scenario.

<table>
<thead>
<tr>
<th>$Qp$</th>
<th>pixel-ME (s)</th>
<th>QDCT-ME (s)</th>
<th>MB-based (s)</th>
<th>$\Delta$Time (%)</th>
<th>pixel-ME (s)</th>
<th>QDCT-ME (s)</th>
<th>MB-based (s)</th>
<th>$\Delta$Time (%)</th>
<th>pixel-ME (s)</th>
<th>QDCT-ME (s)</th>
<th>MB-based (s)</th>
<th>$\Delta$Time (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Crew</td>
<td>82</td>
<td>348</td>
<td>167</td>
<td>-52.07</td>
<td>69</td>
<td>211</td>
<td>93</td>
<td>-56.02</td>
<td>52</td>
<td>83</td>
<td>58</td>
<td>-29.54</td>
</tr>
<tr>
<td>Riverbed</td>
<td>103</td>
<td>556</td>
<td>275</td>
<td>-50.59</td>
<td>93</td>
<td>416</td>
<td>196</td>
<td>-52.76</td>
<td>77</td>
<td>243</td>
<td>105</td>
<td>-56.85</td>
</tr>
<tr>
<td>Duckstakeoff</td>
<td>97</td>
<td>535</td>
<td>296</td>
<td>-44.72</td>
<td>86</td>
<td>396</td>
<td>208</td>
<td>-47.58</td>
<td>71</td>
<td>231</td>
<td>149</td>
<td>-35.33</td>
</tr>
<tr>
<td>Shuttlestart</td>
<td>45</td>
<td>118</td>
<td>61</td>
<td>-48.52</td>
<td>31</td>
<td>41</td>
<td>31</td>
<td>-24.84</td>
<td>11</td>
<td>23</td>
<td>16</td>
<td>-27.84</td>
</tr>
</tbody>
</table>

Table 6.2. Average time usage of secondary SP-frames with different $Qp$ for the switching-up scenario.

<table>
<thead>
<tr>
<th>$Qp$</th>
<th>pixel-ME (s)</th>
<th>QDCT-ME (s)</th>
<th>MB-based (s)</th>
<th>$\Delta$Time (%)</th>
<th>pixel-ME (s)</th>
<th>QDCT-ME (s)</th>
<th>MB-based (s)</th>
<th>$\Delta$Time (%)</th>
<th>pixel-ME (s)</th>
<th>QDCT-ME (s)</th>
<th>MB-based (s)</th>
<th>$\Delta$Time (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Crew</td>
<td>93</td>
<td>498</td>
<td>262</td>
<td>-47.29</td>
<td>96</td>
<td>508</td>
<td>266</td>
<td>-47.50</td>
<td>99</td>
<td>526</td>
<td>306</td>
<td>-41.83</td>
</tr>
<tr>
<td>Riverbed</td>
<td>108</td>
<td>631</td>
<td>328</td>
<td>-48.00</td>
<td>107</td>
<td>630</td>
<td>327</td>
<td>-48.12</td>
<td>109</td>
<td>630</td>
<td>324</td>
<td>-48.60</td>
</tr>
<tr>
<td>Duckstakeoff</td>
<td>104</td>
<td>604</td>
<td>348</td>
<td>-42.39</td>
<td>105</td>
<td>608</td>
<td>349</td>
<td>-42.57</td>
<td>109</td>
<td>632</td>
<td>407</td>
<td>-35.63</td>
</tr>
<tr>
<td>Shuttlestart</td>
<td>60</td>
<td>264</td>
<td>125</td>
<td>-52.58</td>
<td>64</td>
<td>296</td>
<td>138</td>
<td>-53.34</td>
<td>68</td>
<td>329</td>
<td>147</td>
<td>-55.41</td>
</tr>
</tbody>
</table>

Table 6.3 shows the comparison between the frame-based hybrid scheme in Chapter 5 and the proposed MB-based hybrid scheme. It illustrates the size reduction and average time usage of the proposed scheme over QDCT-ME. The result obviously shows that the MB-based hybrid scheme significantly outperforms QDCT-ME on size reduction and time usage of secondary SP-frame coding. Moreover, the proposed scheme performs better than the frame-based hybrid scheme. The reason is that FMO is adopted in the MB-based hybrid scheme. Correlation of macroblocks has been considered. Mixing motion vectors in a frame can really be beneficial to the coding efficiency and computational complexity of encoding secondary SP-frames.
Table 6.3. Comparison between frame-based and MB-based hybrid schemes on switching-up scenario ($Qp = 24$ to $Qp = 20$).

<table>
<thead>
<tr>
<th></th>
<th>Frame-based scheme</th>
<th>MB-based Scheme</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$\Delta$size</td>
<td>$\Delta$time</td>
</tr>
<tr>
<td>Mobisode1</td>
<td>+0.25%</td>
<td>-9.69%</td>
</tr>
<tr>
<td>Mobisode2</td>
<td>+0.85%</td>
<td>-32.77%</td>
</tr>
<tr>
<td>Crew</td>
<td>+0.63%</td>
<td>-41.92%</td>
</tr>
</tbody>
</table>

6.4 Chapter summary

In this chapter, an adaptive MB-based approach for motion estimation in coding H.264 secondary SP-frames has been designed. The proposed hybrid scheme combines motion estimation techniques at MB level, which are operated at two different domains: QDCT domain and pixel domain. The QDCT-domain motion estimation in secondary SP-frame coding offers higher coding efficiency for macroblocks with complex spatial or temporal activities, but it increases encoding time. The pixel-domain motion estimation is more suitable for macroblocks in homogeneous area, and requires less encoding time. A new measurement of inter-frame correlation based on the bit-counts of the macroblocks in primary SP-frames has been proposed to determine the combination of the two techniques. The hybrid scheme adaptively selects QDCT or pixel domain motion estimation at macroblock level, which has the advantages of utilizing both domains in coding secondary SP-frames. Owing to the adoption of motion vector prediction in H.264, unsystematically mixing motion vectors obtained from the two different domains might increase the bit-
count required by a secondary SP-frame. By making use of an explicit mode in FMO, defining the patterns of the slice groups according to the bit-counts of primary SP-frames is another contribution. Experimental results showed that the proposed measure of determining the proper domain for motion estimation in secondary SP-frame coding is effective. The hybrid algorithm significantly reduces the bit-counts of secondary SP-frames. Additionally, the computational complexity can be saved tremendously comparing to the QDCT scheme.
Chapter 7 Conclusions & Suggestions for Future Research

In this thesis, we have investigated several algorithms for reducing the size of secondary SP-frames in an H.264 video streaming system. The coding mechanisms of SP-frames have been studied in details. We found that the use of secondary SP-frames in the streaming system incurs increase in the storage requirement of the server. To alleviate this, we have incorporated QDCT-domain motion estimation in the encoding process of the secondary SP-frame. These algorithms have been shown to be conducive to the implementation of SP-frames in multiple bitrate and multiple viewpoint videos with high coding efficiency and low computational complexity. In this chapter, we summarize the main contributions of this thesis and further discuss some possible directions that could be the focus for future research.

7.1 Contributions of the thesis

Our contributions mainly include a comprehensive study of SP-frame encoding in order to: (1) design a new QDCT-domain motion estimation algorithm for efficiently coding secondary SP-frames; and (2) contrive hybrid schemes for expediting the new SP-frame coding mechanism with the support of the QDCT-domain motion estimation. In particular, our conclusions are:
The influence of using SP-frames in bitstream switching of multiple bitrate videos is examined. SP-frames can exploit temporal redundancy with predictive coding, but use different reference frames. Even though different reference frames are used, it still allows identical reconstruction. The investigations in Chapter 3 indicate that the traditional pixel-domain motion estimation generates more bits of secondary SP-frames. The reason is that the pixel-domain motion estimation is not consistent with the encoding mechanism of secondary SP-frames. The reference and current frames are quantized and transformed and then subtracted in quantized-transform domain. The pixel-domain motion estimation (pixel-ME) does not aim at minimizing this residue. Consequently, a novel motion estimation technique which is operated in quantized-transform domain for secondary SP-frame coding is suggested. The QDCT-domain motion estimation (QDCT-ME) calculates the sum of absolute difference between quantized-transform coefficients of the original and reference blocks to minimize the residue. The bitcounts of secondary SP-frames can then be reduced substantially.

The adoption of the new QDCT-ME in viewpoint switching of multi-view videos is also investigated in Chapter 4. We reveal that QDCT-ME is more efficient in multi-view videos. It is because the existence of the inter-viewpoint disparity in multi-view videos. Frames in multiple bitrate switching have larger correlation, which causes a few residues. On the other hand, in multi-view videos, frames for encoding secondary SP-frames come from different viewpoints of the same scene. Correlation becomes lower and more bits are required for secondary SP-frames. The aim of QDCT-ME is
to minimize the residue in quantized-transform domain, which is more suitable in this scenario. Analysis also finds that QDCT-ME can also be beneficial in multiple bitrate videos with fast and complex motion. The weaker the interframe correlation is, the better the performance of QDCT-ME can offer.

- It is found that some SP-frames generated by QDCT-ME introduce more bits as compared with its counterpart because the quantized-transform coefficients of QDCT-ME in some blocks are mainly zeros. In this case, the blocks searching over the quantized-transform reference frame are inappropriate. To avoid this, a hybrid motion estimation algorithm combining two motion estimation techniques is proposed in Chapter 5. The new algorithm is based on the inter-frame correlation between the reference frame and the frame being switched to. As a result, bitcounts of primary SP-frames can be utilized to determine which motion estimation technique should be used in the corresponding secondary SP-frame. Again, simulation results confirm its usefulness. Furthermore, the computational complexity can be substantially reduced as compared with QDCT-ME.

- We further extend the hybrid solution to the MB-based approach, which considers the correlation at MB level. A novel measurement of inter-frame correlation based on the bit-counts of the macroblocks in primary SP-frames is designed to select QDCT-ME or pixel-ME at MB level. However, unsystematically mixing motion vectors obtained from the two different domains always increases the bit-count required by a secondary SP-frame.
To solve this, an explicit mode in FMO facilitates to divide MBs belonging to different motion estimation techniques into two slices based on the bit-counts of primary SP-frames. Experimental results prove that the size of secondary SP-frames can be further reduced and the computational time can be saved.

- In our present work, a number of techniques have been investigated that can reduce the size and the computational complexity of secondary SP-frames. We believe that the results obtained in this work contribute significantly to the efficient realization of modern video switching system in both multiple bitrate and multiple viewpoint videos.

### 7.2 Future work

Based on the successful techniques described in this thesis and proven by a wide range of experimental work, we propose here some directions for future research.

#### 7.2.1 Further saving of computational complexity of QDCT-ME

As shown in Chapter 3 and 4, quantized-transform domain motion estimation is computationally intensive. In Equation (3.6), SAQTD involves transforming and quantizing all the pixel blocks to the quantized-transform domain. Its computational complexity is now examined. One straightforward approach to implement the cost function in Equation (3.6) is to directly perform the transformation and quantization, on the fly, during SAQTD calculation. That is,
for each block, the additional transformation and quantization on $Q_s[T(s)]$ and all possible candidates of $Q_s[T(r)]$ within the search window are required. This straightforward implementation results in repetition of the same calculation of $Q_s[T(r)]$ because the search windows of neighbouring blocks are overlapped. This repetition increases linearly with the size of search window. To avoid this repetitive transformation and quantization, all possible locations of $Q_s[T(r)]$ can be pre-calculated and stored in the buffer. By exploiting the reusage of $Q_s[T(r)]$, the computation of SAQTD can be reduced drastically. It means that $4 \times 4$ blocks are transformed to the quantized-transform domain. Let $QT_{4 \times 4}$ be a notation of total number of operations involved in transforming and quantizing a $4 \times 4$ reference block, which is taken as a unit to measure and compare the extra computational complexity of the proposed SAQTD. The size of the video frame is supposed to be $W \times H$. To pre-calculate all possible locations of $Q_s[T(r)]$ in the buffer, the SAQTD requires additional $(W-3) \times (H-3)$ $QT_{4 \times 4}$ operations. It is also noted that the size of the additional buffer is $4(W-3) \times 4(H-3)$. For other block types in motion estimation, this additional buffer can also be re-used. Then, time for computation is expected to reduce drastically. Besides, some phase correlation techniques operated in the DCT domain were designed in [87-88] to estimate motion vectors in the MPEG standard. This phase correlation technique is also possible to expedite our proposed QDCT-ME. Further investigation is necessary since the QDCT-ME is operated in the QDCT domain while the technique in [87-88] is developed in the DCT-domain only.
7.2.2 Entropy coding for secondary SP-frames

It is noted that in Chapters 4 and 5, the bulky size of secondary SP-frames is due to the subtraction between the reference and target frames in quantized-transform domain. The residue has no further quantization before entropy coding. It implies that the existing entropy coding is not designed for encoding this kind of quantized-transform coefficients. It is a great need to propose a method that optimizes the quantized-transform coefficients in entropy coding. In [89-90], an adaptive coding technique of prediction error was proposed. It claimed that based on the hybrid coding using motion-compensated prediction and transform coding of the prediction error, the coding efficiency of the transform is high if the prediction error samples are correlated. For marginally correlated samples, the transform is inefficient. The scan order of residue is adjusted according to the magnitude of gradient of prediction image at the same spatial position. We believe that the technique being used in this scheme is also useful for entropy coding of secondary SP-frames. Future research could explore the way of adjusting scan order similar to this technique.

7.2.3 MVC to AVC transcoding

Nowadays, multi-view video coding framework has been working toward a standard. A multi-view content is coded into an MVC bitstream and sent to a user through the network. However, H.264 decoders can always decode one view of multi-view video bitstreams. In practice, a user can experience just base view. It is always favourable to the user if he/she can switch to another view of multi-view content for H.264 decoding and conventional 2D displaying. To
support this application, transcoding a non-base view from an MVC bitstream to an H.264 bitstream is very useful. Implementing MVC to AVC transcoder can switch multi-view video from one view to another with only H.264 decoders. In this way, a free-viewpoint video system can be supported with only an H.264 decoder in the client and a transcoder in the server.

As shown in Figure 7.1, KS_IBP is the most popular multi-view prediction structures in which only anchor pictures employ inter-view prediction. To transcode a specific view, the coded pictures belonging to the target view and the anchor pictures that are from other views and are dependent by the anchor pictures of the target view are to be processed. For instance, as depicted in Figure 7.1, to extract S2, S0/T0, S2/T0 and so on are needed to decode or transcode as an H.264 bitstream. Furthermore, if only S1 is required by the user, S0/T0, S2/T0, S1/T0 and so on are then decoded. However, not only pictures of the target view are necessary for producing a H.264 bitstream, but frames of other views are also required to include in the bitstream. This situation will
increase the size and the processing time of generating the desired H.264 bitstreams

Figure 7.2. Possible directions for extraction of target view using SI and SB frames.

To eliminate this problem, we can introduce SI and SB-frames as shown in Figure 7.2 in the KP_IBP structures. In order to extract target view S1, only SI, and so on are decoded or transcoded. SB-frame was proposed in [91]. Similar to the SP-frame, this frame type, together with SI-frame, can provide an immediate access point on B-frames. This approach could be applied for solving the switching problem in the new MVC prediction structure.
Chapter 7 Conclusions & Suggestions for Future Research

References


Chapter 7 Conclusions & Suggestions for Future Research


