เทคนิคการเข้ารหัสที่คืนสภาพความผิดพลาดแบบปรับตัวได้สำหรับการส่งวีดิทัศน์ H.265/HEVC

นายฮทู เมือง เมือง

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรดุษฎีบัณฑิต
สาขาวิชาวิศวกรรมไฟฟ้า ภาควิชาวิศวกรรมไฟฟ้า
คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย
ปีการศึกษา 2558

ADAPTIVE ERROR-RESILIENT TECHNIQUES FOR H.265/HEVC VIDEO TRANSMISSION

Mr. Htoo Maung Maung

A Dissertation Submitted in Partial Fulfillment of the Requirements
for the Degree of Doctor of Philosophy Program in Electrical Engineering
Department of Electrical Engineering
Faculty of Engineering
Chulalongkorn University
Academic Year 2015

Thesis Title	ADAPTIVE ERROR-RESILIENT TECHNIQUES FOR H.265/HEVC VIDEO TRANSMISSION
By	Mr. Htoo Maung Maung
Field of Study	Electrical Engineering
Thesis Advisor	Assistant Professor Supavadee Aramvith, Ph.D.
Thesis Co-Advisor	ProfessorYoshikazu Miyanaga, Ph.D.

Accepted by the Faculty of Engineering, Chulalongkorn University in Partial Fulfillment of the Requirements for the Doctoral Degree

....Dean of the Faculty of Engineering
(Professor Bundhit Eua-arporn, Ph.D.)

THESIS COMMITTEE
....Chairman
(Professor Prasit Prapingmongkolkarn, Ph.D.)
....Thesis Advisor
(Assistant Professor Supavadee Aramvith, Ph.D.)
....Thesis Co-Advisor
(ProfessorYoshikazu Miyanaga, Ph.D.)
....Examiner
(Assistant Professor Suree Pumrin, Ph.D.)
....Examiner
(Assistant Professor Widhyakorn Asdornwised, Ph.D.)
....External Examiner
(Associate ProfessorWuttipong Kumwilaisak, Ph.D.)

ฮทู เมือง เมือง : เทคนิคการเข้ารหัสที่คืนสภาพความผิดพลาดแบบปรับตัวได้สำหรับการส่งวีดิทัศน์ H.265/HEVC (ADAPTIVE ERROR-RESILIENT TECHNIQUES FOR H.265/HEVC VIDEO TRANSMISSION) อ.ที่ปรึกษาวิทยานิพนธ์หลัก: สุภาวดี อร่ามวิทย์, อ.ที่ปรึกษาวิทยานิพนธ์ร่วม: - -, 77 หน้า.

การเข้ารหัสวีดิทัศน์ประสิทธิภาพสูง (HEVC) คือ มาตรฐานการเข้ารหัสวีดิทัศน์ล่าสุดที่สามารถส่งผ่านวีดิทัศน์ความละเอียดสูงบนเครือข่ายไร้สาย ในวิทยานิพนธ์นี้ได้นำเสนอการเข้ารหัสวีดิทัศน์แบบทนทานต่อความผิดพลาดบนการเข้ารหัสวีดิทัศน์ประสิทธิภาพสูงโดยมีเป้าหมายสำหรับการประยุกต์ใช้งานส่งผ่านวีดิทัศน์เวลาจริง วิธีการเลือกภาพอ้างอิง(RPS)โดยใช้การป้อนกลับ ที่ถูกเรียกว่าวิธีการเลือกภาพอ้างอิงเชิงลำดับชั้น P ได้ถูกนำเสนอเพื่อให้สามารถปรับตัวเข้ากับขอบข่ายการเข้ารหัสวีดิทัศน์ประสิทธิภาพสูง เนื่องจากโครงสร้างการเข้ารหัสเชิงลำดับชั้นของการเข้ารหัสวีดิทัศน์ประสิทธิภาพสูงทำให้คุณภาพของแต่ละเฟรมในลำดับภาพแปรผันตามตำแหน่งในโครงสร้างเชิงลำดับชั้น ภาพคุณภาพสูงจะถูกใช้เป็นภาพอ้างอิงสำหรับกลุ่มภาพลำดับหลัง ขณะที่ภาพคุณภาพต่ำจะถูกใช้เป็นภาพอ้างอิงสำหรับภาพลำดับถัดไปเท่านั้น เมื่อความผิดพลาดถูกตรวจพบ ขั้นตอนวิธีการเลือกภาพอ้างอิงเชิงลำดับชั้น P จะเลือกภาพอ้างอิงที่เหมาะสมภายใต้การแลกเปลี่ยนระหว่างภาพคุณภาพต่ำกับระยะทางเชิงเวลาสั้นและภาพคุณภาพสูงกับระยะทางเชิงเวลายาว ในการเพิ่มสมรรถภาพของการเลือกภาพอ้างอิงที่นำเสนอ เรารวมขั้นตอนวิธีนี้รวมวิธีการเลือกภาพอ้างอิงโดยใช้การป้อนกลับและวิธีการคืนสภาพภายในจากบริเวณที่สนใจ (ROI) โดยการใช้ลักษณะเฉพาะใหม่บางส่วนของการเข้ารหัสวีดิทัศน์ประสิทธิภาพสูง แต่ละบล็อกในบริเวณที่สนใจถูกเข้ารหัสด้วยวิธีภายในตามสถานะของความผิดพลาดเพื่อให้สามารถเพิ่มคุณภาพของภาพในบริเวณที่สนใจ และสามารถหลีกเลี่ยงกระบวนการค้นหาการเคลื่อนที่ในพื้นที่นั้น ในการเรียกใช้งานการคืนสภาพระดับบล็อก การควบคุมอัตราถูกดัดแปลงเพื่อให้เหมาะสมกับขอบข่ายเช่นกัน ผลการทดลองแสดงให้เห็นว่าขอบข่ายที่นำเสนอสามารถเพิ่มคุณภาพทั้งทางรูปธรรมและนามธรรม เมื่อเปรียบเทียบกับวิธีการเลือกภาพอ้างอิงเชิงลำดับชั้น P ค่าเฉลี่ย PSNR ของวิธีการที่นำเสนอเพิ่มขึ้นจากวิธีการเลือกภาพอ้างอิงที่ถูกดัดแปลง 1.04 เดซิเบล และมากกว่าส่วนชุดคำสั่งอ้างอิงของการเข้ารหัสวีดิทัศน์ประสิทธิภาพสูงกับการคืนสภาพภายในแบบปรกติประมาณ 5.23 เดซิเบล

ภาควิชา วิศวกรรมไฟฟ้า ลายมือชื่อนิสิต ....

สาขาวิชา วิศวกรรมไฟฟ้า ลายมือชื่อ อ.ที่ปรึกษาหลัก ....

ปีการศึกษา 2558 ลายมือชื่อ อ.ที่ปรึกษาร่วม ....

5471469121 : MAJOR ELECTRICAL ENGINEERING

KEYWORDS: H.265/HEVC, ERROR RESILIENT VIDEO CODING, ERROR PROPAGATION, WIRELESS VIDEO TRANSMISSION

HTOO MAUNG MAUNG: ADAPTIVE ERROR-RESILIENT TECHNIQUES FOR H.265/HEVC VIDEO TRANSMISSION. ADVISOR: ASST. PROF. SUPAVADEE ARAMVITH, Ph.D., CO-ADVISOR: PROF.YOSHIKAZU MIYANAGA, Ph.D., 77 pp.

High efficiency video coding (HEVC) is the latest video coding standard that enables the transmission of high resolution videos over wireless networks. In this dissertation, HEVC based error resilient video coding is proposed to target real time video transmission applications. A feedback-based reference picture selection (RPS) method called hierarchical-P RPS is proposed such that it can adapt with the HEVC coding framework. Due to hierarchical coding structure of HEVC, the quality of each frame in the sequence varies depending on its location in the hierarchical structure. High quality pictures are used as reference for several subsequent pictures while low quality pictures are used for only one consecutive picture. When error is detected, hierarchical-P RPS algorithm will select suitable reference pictures subject to the trade-off of low quality picture with short temporal distance and high quality picture with long temporal distance. To enhance the performance of the proposed RPS, we combine feedback-based RPS and region-of-interest (ROI) based intra refresh method by using some novel features of HEVC. Each block in the ROI region is encoded with intra mode according to error status such that it can enhance the quality of picture in ROI area and can avoid the motion search process in this area. To enable block level intra refresh, rate control is also modified to fit in the framework. Experimental results demonstrate that the proposed framework can achieve better subjective and objective quality, compared to the hierarchical-P RPS algorithm. The average PSNR improvement of proposed algorithm over modified RPS algorithm is 1.04 dB and that over HEVC reference software with regular intra frame refresh is about 5.23 dB.

Department:	Electrical Engineering	Student's Signature	....
Field of Study:	Electrical Engineering	Advisor's Signature	....
Academic Year:	2015	Co-Advisor's Signature	....

ACKNOWLEDGEMENTS

First and foremost, I would like to express my sincere gratitude to my Ph.D. advisor, Assistant Professor Supavadee Aramvith, for her continuous support throughout my studies. Without her patience, kindness, encouragement, enthusiasm, and guidance, I would still be looking for a new idea to get started on my research. I would also like to thank my co-advisor, Professor Yoshikazu Miyanaga, for his support and suggestions.

My sincere thanks go to committee members, Professor Prasit Prapinmongkolkarn, Assistant Professor Suree Pumrin, Assistant Professor Widhyakorn Asdornwised, and Associate Professor Wuttipong Kumwilaisak, for their valuable comments and technical suggestions in the completion of this work.

Appreciation is extended to both friends in Hokkaido and all the members of our research group in Thailand for their support and encouragement. A special thanks also goes to Watchara Ruangsang for always helping me and also sharing good thoughts to me. I would also like to extend my gratitude to all the staffs of ISE and AUN/SEED-Net, for their support.

Last but not least, I would like to thank my parents for their love and understanding. Without their support and encouragement, this dissertation could not be possible.

This research has been supported in part by the Collaborative Research Project entitled Video Processing and Transmission, JICA Project for AUN/SEED-Net, Japan.

CONTENTS

Page
THAI ABSTRACT iv
ENGLISH ABSTRACT..v
ACKNOWLEDGEMENTS vi
CONTENTS..vii
LIST OF TABLES 1
LIST OF FIGURES 2
CHAPTER 1 INTRODUCTION 4
  1.1. Motivation and Problem Statement 4
  1.2. Contribution 8
  1.3. Outline of Thesis..9
CHAPTER 2 BACKGROUND 11
  2.1. Elements of video coding 11
  2.2. H.265/HEVC 16
    2.2.1. Quad-Tree Block and Transform Structure 17
    2.2.2. Reference Picture Set 18
    2.2.3. Intra Prediction and Coding 19
    2.2.3. Motion Estimation..20
    2.2.4. Entropy Coding 20
    2.2.5. In-Loop Filtering 20
  2.3. Comparison of Error Resilience Tools in H.264 and HEVC.........................20
  2.4. Error Resilient Video Coding 22
  2.5. Rate Control in Video Coding 26
CHAPTER 3 ADAPTIVE ERROR-RESILIENT TECHNIQUES FOR HEVC.........30
  3.1. Introduction..30
  3.2. Hierarchical-P RPS system for HEVC 31
  3.3. Proposed interactive error control for HEVC 33
    3.3.1. Moving region extraction 34
    3.3.2. Early termination of coding tree splitting process...............................35

Page

3.3.3. Coding mode selection for LCUs in MR....36
3.3.4. Modification of rate control 37
3.3.5. Summary of proposed error resilient algorithm 38
CHAPTER 4 EXPERIMENTAL RESULTS AND DISCUSSIONS....40
4.1. Experimental Setup....40
4.2. Result Analysis 42
CHAPTER 5 CONCLUSION AND FUTURE WORKS....60
REFERENCES 62
VITA....77

LIST OF TABLES

Page

Table 4.1: Summary of encoder configurations 41

Table 4.2: Target bit rates 42

Table 4.3: Results comparison for BasketballDrillText (10% PER) 53

Table 4.4: Results comparison for BasketballDrillText (5% PER) 53

Table 4.5: Results comparison for BasketballDrillText (3% PER) 53

Table 4.6: Results comparison for BQMall (10% PER) 54

Table 4.7: Results comparison for BQMall l (5% PER) 54

Table 4.8: Results comparison for BQMall (3% PER) 54

Table 4.9: Results comparison for PartyScene (10% PER) 55

Table 4.10: Results comparison for PartyScene (5% PER) 55

Table 4.11: Results comparison for (3% PER) 55

Table 4.12: Results comparison for Johnny (10% PER) 56

Table 4.13: Results comparison for Johnny (5% PER) 56

Table 4.14: Results comparison for Johnny (3% PER) 56

Table 4.15: Results comparison for FourPeople (10% PER) 57

Table 4.16: Results comparison for FourPeople (5% PER) 57

Table 4.17: Results comparison for FourPeople (3% PER) 57

Table 4.18: Results comparison for KristenAndSara (10% PER) 58

Table 4.19: Results comparison for KristenAndSara (5% PER) 58

Table 4.20: Results comparison for KristenAndSara (3% PER) 58

LIST OF FIGURES

Page
Figure 2.1: Typical block diagram of video encoder and decoder. 12
Figure 2.2: Example of error propagation...13
Figure 2.3: IPPP coding structure 14
Figure 2.4: IBBP coding structure. 14
Figure 2.5: Hierarchical-B coding structure. 14
Figure 2.6: Hierarchical-P coding structure...15
Figure 2.7: Coding tree structure of HEVC 17
Figure 2.8: Splitting modes of prediction unit 17
Figure 2.9 Reference picture set of HEVC 18
Figure 2.10 Intra prediction modes 19
Figure 2.11 NEWPRED system for round trip delay of 2 frame intervals 24
Figure 3.1: Modified A-NEWPRED system (a) a common frame loss and (b) a core frame loss. 32
Figure 3.2: Overall block diagram of a feedback-based error resilient video codec. .33
Figure 3.3: Output of moving region extraction procedure. (a) Johnny; (b) BQMall .35
Figure 3.4 Bit budget and parameters of rate control in HEVC. (a) Original R-lambda rate control; (b) Modified ROI-based rate control 37
Figure 4.1: Sample frames of HEVC test sequences used in experiments. Sequences (a) – (c) and (d) – (f) have a resolution of 832x480 and 1280x720 pixels, respectively...40
Figure 4.2: Rate-distortion performance between RPS and proposed method for 3% PER. (a) BasketballDrillText; (b) BQMall; (c) PartyScene; (d) Johnny; (e) FourPeople; (f) KristenAndSara 45
Figure 4.3: Rate-distortion performance between RPS and proposed method for 5% PER. (a) BasketballDrillText; (b) BQMall; (c) PartyScene; (d) Johnny; (e) FourPeople; (f) KristenAndSara 46
Figure 4.4: Rate-distortion performance between RPS and proposed method for 10% PER. (a) BasketballDrillText; (b) BQMall; (c) PartyScene; (d) Johnny; (e) FourPeople; (f) KristenAndSara 47

Page

Figure 4.5: Frame by frame PSNR values of BQMall sequence for 10% PER at target bit rate (a) 384kbps and (b) 2Mbps, respectively. 48

Figure 4.6: Frame by frame PSNR values of PartyScene sequence for 10% PER at target bit rate (a) 384kbps and (b) 2Mbps, respectively. 49

Figure 4.7: Frame by frame PSNR values of Johnny sequence for 10% PER at target bit rate (a) 256kbps and (b) 1.5Mbps, respectively. 50

Figure 4.8: Frame by frame PSNR values of KristenAndSara sequence for 10% PER at target bit rate (a) 256kbps and (b) 1.5Mbps, respectively. 51

Figure 4.9: Frame 137 of KristenAndSara sequence under 10% PER. (a) Original; (b) RPS_noI; (c) RPS_wI; (d) Proposed Method 59

Figure 4.10: Frame 57 of BasketballDrillText sequence under 10% PER. (a) Original; (b) RPS_noI; (c) RPS_wI; (d) Proposed Method 59

Figure A. 1: Frame by frame PSNR values of BQMall sequence for 10% PER at target bit rate (a) 512kbps, (b) 768kbps, and (c) 1.2Mbps, respectively. 67

Figure A. 2: Frame by frame PSNR values of BasketballDrillText sequence for 10% PER at target bit rate (a) 384kbps, (b) 512kbps, (c) 768kbps, (d) 1.2Mbps, and (e) 2Mbps, respectively. 69

Figure A. 3: Frame by frame PSNR values of PartyScene sequence for 10% PER at target bit rate (a) 512kbps, (b) 768kbps, and (b)1.2Mbps, respectively. 71

Figure A. 4: Frame by frame PSNR values of KristenAndSara sequence for 10% PER at target bit rate (a) 384kbps, (b) 512kbps, and (c) 850kbps, respectively..........72

Figure A. 5: Frame by frame PSNR values of Johnny sequence for 10% PER at target bit rate (a) 384kbps, (b) 512kbps, and (c) 850kbps, respectively......................74

Figure A. 6: Frame by frame PSNR values of FourPeople sequence for 10% PER at target bit rate (a) 256kbps, (b) 384kbps, (c) 512kbps, (d) 850kbps, and (e) 1.5Mbps, respectively. 76

CHAPTER 1
INTRODUCTION

1.1. Motivation and Problem Statement

According to the recent report by Cisco [1], mobile video traffic exceeded half of global mobile data traffic in 2012 and the amount of traffic is estimated to keep growing. Demands for higher coding efficiency to reduce the burden of network traffic is considerably increased. Japan has been introduced some of the world’s first 4K channels since 2014. In 2012 London Olympic, opening and closing ceremonies were shot by using 8K resolutions cameras. These trends indicate that the need for high throughput in television broadcasting. Moreover, almost every portable device nowadays includes one or more cameras. Video becomes a “must have” on mobile devices. Energy efficient design of video codec is demanding especially for mobile applications.

In order to fulfill these requirements, the latest video coding standard, High Efficiency Video Coding (HEVC) [2], has been jointly developed by ITU-T and ISO/IEC. By introducing flexible quadtree structure, sample adaptive offset (SAO), parallel processing tools, and other cutting-edge techniques, HEVC achieves high compression ratio which is two times higher than that of its predecessor, H.264/AVC [3]. New features of HEVC and their implementation friendliness can achieve high throughput with low power consumption.

Due to the rapid advances in technology, consumer devices such as smart phones, tablets, cameras, etc. are getting more and more connected to each other. These devices also have capabilities to capture and display high quality videos. Consequently, demands for sharing and consuming of high resolution videos are increased.

The high compression ratio of HEVC also brings the potential to enable HD video transmission in low bandwidth networks. The ever increasing demands for high quality video services over existing networks can be fulfilled with the help of HEVC. However, high resolution video transmission over packet-switched wireless networks, especially for real-time conversational applications, is still a challenging issue due to limitation of available bandwidth, network congestion, delay, and error prone nature of wireless channel. Error robustness of encoded bitstream is important to provide good quality of decoded video.

There are some recent papers in the literature studying about error robustness of HEVC encoded bitstreams [4-7]. In [4], the error robustness of HEVC and H.264/AVC are compared by using PSNR merit. It is shown that HEVC has less error resiliency than H.264/AVC especially for high packet error rates (PER). Pinol et al. [5] measured the error robustness of HEVC in vehicular Ad-Hoc network (VANET). The quality of video over VANET is difficult to control because of attenuation, Doppler effect, and rapid varying network topology. The quality of HEVC bitstream becomes intolerable when the packet error rate exceeds 1.3%. It is found that HEVC videos are very sensitive to packet losses even if regular I frame is added in the bitstream. In [6], a no-reference video quality assessment method for HEVC in loss-prone networks. Experimental results show that packet losses have a more severe impact on HEVC bitstram than H.264/AVC bitstream. This is because HEVC uses open GOP structure to improve coding efficiency that increases very high data dependency between frames. The errors on certain frame have the very high potential to propagate to the end of GOP due to open GOP structure. Nightingale et al. [7] investigated the impact of network impairment on quality of experience in HEVC video streaming. They used a hardware testbed that includes both wire and wireless networks. HEVC encoded videos are transmitted over the testbed with various packet error rates ranging from 1% to 5%. It is found that users can tolerate the video quality up to 1% PER. However, most users are annoying when packet loss ratio is 3% and the video quality becomes unacceptable when packet loss ratio goes beyond 3% PER. User tolerance of packet loss drastically decreases between 1% PER and 3% PER.

Thus, all studies show that HEVC bitstreams are extremely vulnerable against packet errors. To guarantee the quality of output video, a proper error resilient technique should be applied to the HEVC bitstream. According to these studies, the quality of HEVC video becomes unacceptable if the packet error rates exceeds 1% PER. Hence, this research only focuses on the high packet error rates ranging from 3% PER to 10% PER. These packet error rates are commonly used rates in the experiments for video transmission over lossy networks.

Several error resilient methods have been studied to improve the quality of video for transmission applications over lossy channels. These methods can roughly be categorized into three groups: encoder-based techniques, error concealment at decoder, and interactive error control. Feedback-based error resilient methods are in interactive error control group that use feedback information from the decoder to mitigate the network errors. Error mitigation techniques triggered by feedback messages are suitable when there is a bidirectional communication between the receiver and the sender. Comprehensive review of feedback-based techniques can be found in [8, 9].

Feedback-based retransmission technique can inherently adapt varying loss rates. Retransmissions are only triggered if the information is actually lost. The generated overhead is directly proportional to the loss rate. However, retransmission technique is not suitable for low-latency application like video telephony. Another typical feedback-based error resilient video coding techniques are intra refresh and reference picture selection (RPS). The feedback-based intra refresh method stops error propagation by switching to intra coding mode for coding blocks in the corrupted areas based on the feedback messages. In RPS method error propagation is stopped by prohibiting the affected blocks using as reference. The effectiveness of feedback-based error resilient techniques mainly depends on the amount of round trip delay.

For the high PER, intra refresh method is required to encode a large number of blocks by using Intra mode. Inserting a lot of Intra blocks reduce the coding efficiency. Meanwhile, RPS method selects the error free reference to avoid error propagation and encodes current frame by using Inter mode. Therefore, coding efficiency is not much affected except for very large round trip delay case. Since the target application in this research is conversational application, the maximum round trip delay for this application should be less than 150ms [10]. The round trip delay for this thesis is assumed as ~100ms. Therefore, in this thesis, we focus on RPS method for low delay conversational applications using HEVC encoded bitstreams because RPS method is effective for this amount of delay and feedback channel is available.

NEWPRED [11, 12] is a RPS method that uses feedback information to limit the error propagation by restricting the prediction from corrupted images. NEWPRED can operate in two modes: acknowledgement (ACK) mode and negative acknowledgement (NACK) mode. In [13], the performances of ACK mode (A-NEWPRED) and NACK mode (N-NEWPRED) are compared for various network conditions, reference distances, video contents, and Group of Pictures (GOP) lengths. It is reported that A-NEWPRED is more sensitive to round-trip delay whereas N-NEWPRED is more sensitive to packet loss. In [14], proxy-based reference picture selection method for mobile video telephony scenario is proposed. This method uses retransmission of lost packets approach for wireless downlink and adaptive reference selection approach for wireless uplink. Adaptive reference selection in [14] is an extended version of NEWPRED with extensions on slice level reference selection and slice level reference selection with error concealment at the encoder. Chenghao et al. [15] proposed a reference picture selection method by using long-term reference picture.

However, these RPS methods are mainly designed and tested for conventional IPPP coding structure and only one reference picture case is considered for motion prediction. Meanwhile, HEVC uses hierarchical-P coding structure with multiple reference pictures approach for low delay applications. In hierarchical-P coding structure, not all frames are encoded with same quality. Core frames, which arrear periodically in the sequence, are usually encoded with low quantization parameter (QP) value. The qualities of core frames are better than other frames and are normally served as reference pictures for several frames. Apart from core frames, the remaining frames are called common frames. These frames are normally encoded with high QP value. Because of its quality, common frames are normally used as reference picture for only one successive frame in hierarchical coding structure. The more details about coding structure are provided in Chapter 2. HEVC also introduces a new feature for decoded picture buffer (DPB) management called reference picture set. Therefore, reference picture signalling and DPB management are different from previous standards.

In addition, if the RPS algorithm is adopted in hierarchical coding structure, common frames may be required to use as reference for several frames depending on error status. For this case, the quality of common frame has an impact on the predicted frame. If selecting only core frames for reference pictures under error condition, the large temporal distance of core frame may require large searching area for motion vector and thereby increasing complexity at the encoder.

In this thesis, the conventional RPS technique is firstly modified in order to adapt the HEVC framework. Secondly, an algorithm is proposed to overcome the aforementioned shortcomings of RPS technique in hierarchical coding structure.

1.2. Contribution

There are four main contributions in this thesis. They are listed as follows.

1. **Feedback-based RPS algorithm for hierarchical-P coding structure:** conventional feedback-based RPS error resilient techniques are designed for IPPP coding structure and using single reference picture for motion prediction. But for HEVC, it uses hierarchical-p coding structure with multiple reference pictures for motion prediction to achieve high coding efficiency. Moreover, HEVC introduces new tool, reference picture set, for decoded picture buffer (DPB) control. The conventional RPS algorithm uses memory management control operation (MMCO) commands for buffer management. This DPB management part of RPS is also required to modify according to new reference picture set feature of HEVC. To adapt the existing RPS algorithm with HEVC encoding framework and its new features, a feedback-based RPS algorithm is proposed in this thesis.
2. **Combination of ROI-based intra refresh algorithm and modified RPS algorithm:** although the modified RPS can tackle the packet loss, some shortcomings are found due to hierarchical coding structure. As explained in previous section, two main shortcomings are using common frame as reference for several frames and using core frame which has long temporal distance as reference. To improve the performance of modified RPS, an algorithm that jointly considers ROI-based intra refresh method and RPS method is proposed. If common frame is used as reference, its quality in important region is enhanced by using intra refresh or if core frame is used as reference, the large computational cost causes by long temporal distance is reduced by introducing intra coded blocks. According to the experimental

results, the proposed algorithm outperforms the modified RPS algorithm for almost all test cases.

3. **Early termination of coding tree splitting process using MR information:** the coding tree splitting process of HEVC is done by using RDO process. Although this process can achieve best rate distortion performance, it requires large computation time. To reduce computation time for delay constraint applications, an early termination of splitting process is proposed by using ROI information.
4. **Modification of R-Lamda Rate Control:** current R-lambda rate control of HEVC computes a QP value for each frame by using a set of parameter values. In the proposed algorithm, each frame is divided into two regions. To encode with different quality in each region, different QP values are required. Firstly, bit allocation process is modified to region based bit allocation process. Then, parameter values are determined according to coding mode of each region. Finally, QP value for each region is computed. Parameter updating process is also modified from frame-based approach to region-based approach. The modified rate control scheme works well and can maintain the target bit rate. Even for the highest packet error case that requires insertion of many intra coded blocks, the increment in output bit rate is less than 0.5% of the target bit rate while showing better performance than the reference method.

1.3. Outline of Thesis

This thesis is organized into five chapters including this chapter. The following paragraphs provide brief descriptions of the remaining chapter of this thesis.

Chapter 2 provides some background about the basics of video coding and new features of H.265/HEVC video coding standard. Literature review on feedback-based error resilient techniques are descirbed. The rate control concept in video coding is briefly reviewed and R-lambda rate control of HEVC is introduced.

Chapter 3 presents the feedback-based error resilient techniques. Both hierarchical-P RPS algorithm and hierarchical-P RPS with ROI-based intra refresh algorithm are explained in details.

Chapter 4 explains the experimental setup and test sequences. Simulation results are described, analyzed, and discussed in this chapter. The impacts of PLR, reference distance, round trip delay, and characteristics of video sequences on the quality of output video are also discussed.

Chapter 5 includes conclusions and future works of the research.

CHAPTER 2
BACKGROUND

A brief review of the fundamental elements in video compression is provided in this chapter. Then the latest video coding standard, H.265/HEVC, are introduced by highlighting some important new features. Next, literature review on error resilient video coding techniques focusing on feedback-based methods are presented. A brief introduction of rate control is also provided. After that R-lambda rate control of HEVC is discussed.

2.1. Elements of video coding

Representations of video sequences in digital format require large amount of bits, which is impractical due to the limitation of storage and transmission costs. Fortunately, video signals contain a lot of redundancies that can be exploited for efficient compression. There are four kinds of redundancies in a video sequence: spatial redundancies, temporal redundancies, perceptual redundancies and statistical redundancies.

Similarities in the pixel values within the same frame are called spatial redundancies. Similarities between subsequent frames are called temporal redundancies. In general, consecutive frames are similar, except for motion of objects, panning of the camera or changes of scenes. For some details of a picture (i.e., high frequency components of the picture), a human eye cannot perceive and, therefore, removing these details cannot affect the quality of the picture. Perceptual redundancies refer to the information of a picture that a human eye cannot perceive. In entropy coding stage, the probabilities of occurrence for some code values are relatively high compare to others. That kind of data duplication is called statistical redundancies.

The amount of information to be coded is reduced by exploiting these redundancies. Predictive coding is used to exploit the special and temporal redundancies. Intra prediction predicts pixel values of a block based on neighboring pixels values. Inter prediction use motion compensation to reduce temporal redundancies. Human eyes are more sensitive to the changes in intensity than variation in color and are less sensitive to high special frequencies. To exploit perceptual redundancies, any information that human eyes cannot perceive is removed. Reducing of statistical redundancies can be carried out by using variable length coding at the entropy encoder. Variable length coding assigns shorter code lengths to the values that are more frequently appear than others. The longer code lengths are used for remaining values.

To achieve high compression ratio, a video encoder combines several coding algorithms. In block-based hybrid video coding, the predictive coding, transform coding, and entropy coding are combined. There are four main steps in block-based hybrid video coding. They are image partitioning, prediction, transform coding and entropy coding.

Each picture of raw video sequence is firstly split into small square blocks in which the block size is commonly chosen as 16x16. These blocks serve as the basic units for further processing. Either intra or inter prediction is applied for each block in order to get the prediction signal. By finding the difference between the predicted signal and the original signal, the residual signal for each block is obtained. The residual signal of each block is transformed into frequency domain by using DCT (discrete-cosine transform). Then, the coefficient values are quantized to remove high frequency components. Finally, quantized coefficient values are entropy coded to produce the output bit stream. The decoder simply reproduces the output video by applying the reverse process.

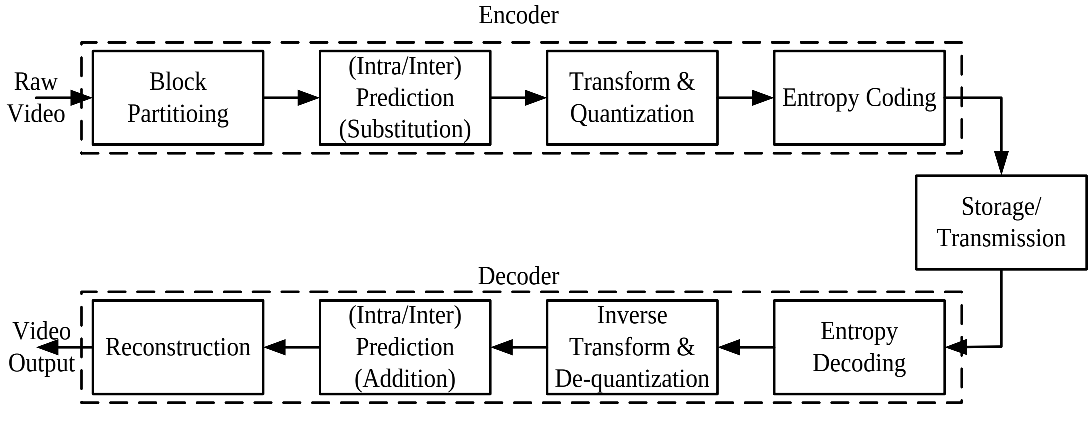


Figure 2.1: Typical block diagram of video encoder and decoder.

One of the key components of hybrid codec is inter prediction or motion compensated prediction. In order to do inter prediction, the encoder has a buffer, which is known as decoded picture buffer (DPB), to store the reconstructed version of previously encoded pictures. The pictures in DPB are then served as reference frames for motion prediction process. Motion vectors of the current block, which are very important for successful decoding process, are obtained after this process. The decoder also keeps reference pictures in its DPB. The reference pictures at the encoder must be the same with that at the decoder to get the correct output video.

If the reference picture of a frame at the decoder is not identical to the one at the encoder, distortion will occur at the decoded frame. This situation commonly happens in transmission applications where some data packets of a frame are lost by network errors. Even if the decoder applies some sort of error concealment methods, some errors still remain at the output frame. In motion predictive video coding framework, such kind of error can propagate to subsequent frames. This phenomenon is called “error propagation”. If several small errors are accumulated in this way, the quality of resulted output will be very poor.

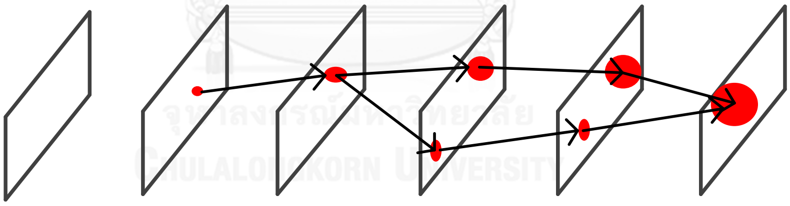


Figure 2.2: Example of error propagation.

The coding structure also plays an important role in hybrid video coding. The coding structure has considerable impact on coding efficiency, processing time, and error propagation behaviour. Some commonly used coding structures are IPPP coding structure, IBBP coding structure, hierarchical-B coding structure, and hierarchical-P coding structure.

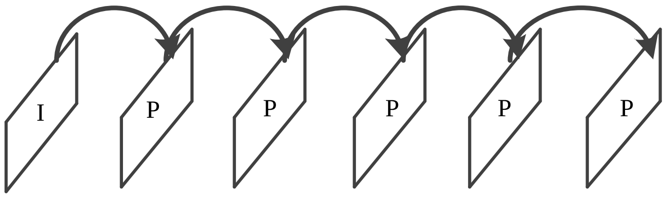


Figure 2.3: IPPP coding structure

The basic form of IPPP coding structure is shown in Fig. 2.3. In this coding structure, the reference picture of each P frame is the frame that immediately precedes it. In the figure, the direction of prediction is shown by arrows from the reference frame to the frame to be encoded. This coding structure is mainly used in low delay applications because its coding order is always same with display order and thereby no additional delay is introduced.

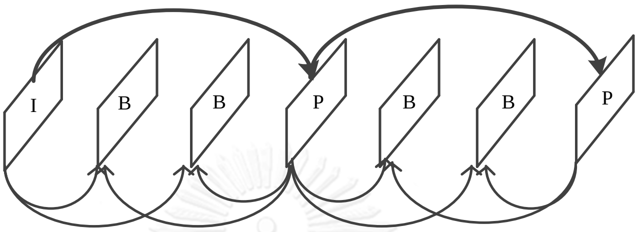


Figure 2.4: IBBP coding structure.

The IBBP coding structure is commonly used in broadcasting applications. This coding structure is aimed to increase coding efficiency than previous IPPP structure by using bi-directionally predicted picture (B-frame). To be able to encode a frame with bi-directional prediction mode, a frame that precedes in display order is required to wait until a P frame is encoded. Then by using this p frame and previously encoded I/P frame, a B frame is generated. So, the coding order and display order in this coding structure is not the same and that introduces some delay. For applications where coding efficiency is more important than processing time, IBBP structure is used.

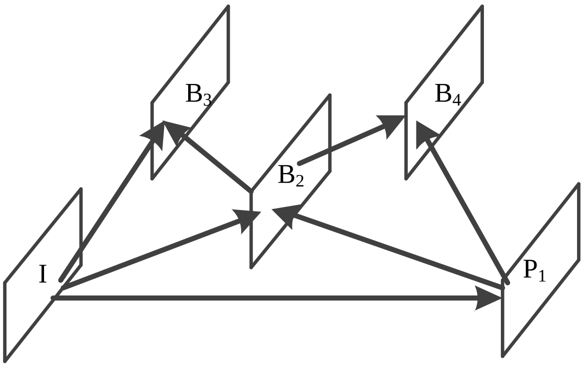


Figure 2.5: Hierarchical-B coding structure.

Since demands for better coding efficiency are kept increasing, another coding structure called hierarchical B coding structure is introduced. This coding structure is not only increased the coding efficiency but also provided the temporal scalability of the encoded bitstreams. The example in Fig. 2.5 has three different temporal layers. After I frame is encoded, frame P_1 is encoded by using I as reference. Then frame B_2 is encoded by using I and P_1 frames as reference. Next, frame B_3 and B_4 are encoded respectively. Therefore, frame I and P_1 are in temporal layer 1, which are very important frames because the frames in other layers use these frames as reference. Frame B_2 is in temporal layer 2, which is also important for encoding higher layer frames (i.e., B_3 and B_4). The frames B_3 and B_4 are in temporal layer 3. In this coding structure, each layer has different importance level. For example, layer 1 is the most important layer because the frames in that layer are important for successful decoding of the frames at higher layers. For further bit rate saving, the quality of picture at each temporal level differentiate by using differ QP values. Normally, the frames in the lowest temporal layer are encoded with the lowest QP values to achieve high quality and that of high temporal layer are encoded with high QP values. The hierarchical B coding structure can save about 20% of bit rate compared with IBBP structure [16].

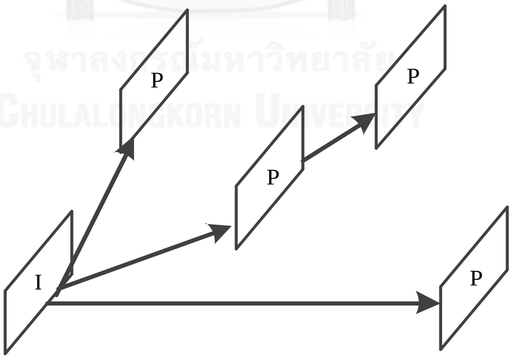


Figure 2.6: Hierarchical-P coding structure.

Although the hierarchical B achieves the high coding efficiency, it is not suitable for delay constrained applications. Hierarchical P coding structure is recently getting more attention to employ in applications that have delay constraint, computational power constraint, and energy constraint. The encoding order and display order in hierarchical P coding structure is the same. This structure also has temporal scalability. The frames in each temporal layer are encoded with different QP values to achieve further bit saving. The advantages of hierarchical P coding structure over IPPP coding structure are high coding efficiency and enhanced error resilience. This coding structure is appropriate for applications that have either delay constraint or complexity constraint. In addition, the temporal scalability of this coding structure can be used for error resilience. If the decoder detects an error at a frame from temporal layer 3, then other frames from temporal layer 1 and 2 can still be correctly decoded.

2.2. H.265/HEVC

HEVC also use the block-based hybrid video coding approach as its predecessor H.264/AVC. The improvement in coding efficiency of HEVC is acquired by combining numerous small enhancements from almost all parts of the encoder over earlier designs. Moreover, HEVC introduces several new features that have more implementation friendliness on parallel processing.

The major improvements of HEVC are summarized as follows:

1. HEVC use larger block size than macroblock of previous students. These blocks are very flexible and can be subdivided into different small sizes.
2. HEVC has greater flexibility in prediction modes and transform block sizes than earlier standards.
3. It uses more sophisticated interpolation and deblocking filters.
4. It has more sophisticated prediction and signaling of modes and motion vectors.
5. HEVC supports better parallel processing architecture.

For the emerging devices and services, HEVC aimed to improve coding efficiency, to achieve high data throughput, and to reduce power consumptions. Several new features of HEVC are developed to address these three objectives. Some of the major enhancements to gain better coding efficiency are the usage of larger coding block size, sophisticated intra prediction, larger interpolation filter for motion compensation, high throughput CABAC, and sample adaptive offset (SAO) filter. High throughput and low power objectives are achieved by using parallel deblocking filter, high level parallel tools, and parallel skip/Merge mode. Some important features of HEVC are discussed in the following subsections.

2.2.1. Quad-Tree Block and Transform Structure

In block-based video coding framework, each frame is firstly divided into several basic coding blocks before prediction, transform, and other processes. Macroblocks or 16x16 fixed size basic coding blocks are used in all previous ITU-T and ISO/IEC video coding standards including H.264/AVC. In order to adapt the various video contents at different resolutions, flexible block sizes are more desirable than fixed one. Coding tree unit (CTU) of HEVC can offer highly flexible block sizes that can configure from 64x64 to 8x8 [2].

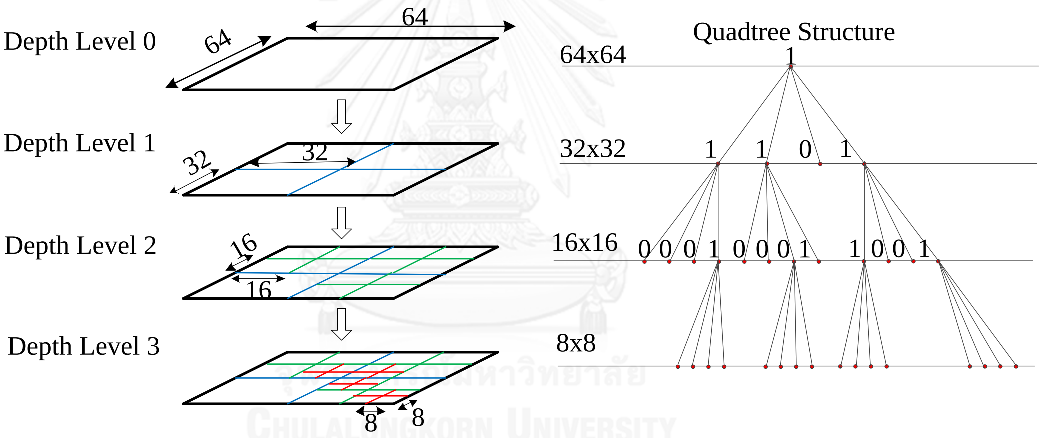


Figure 2.7: Coding tree structure of HEVC

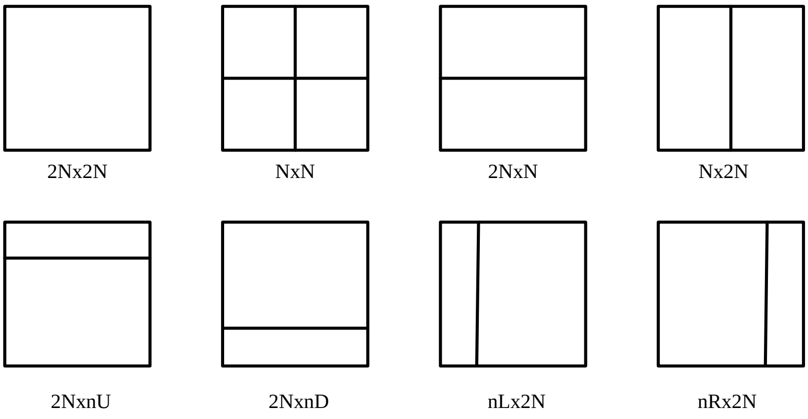


Figure 2.8: Splitting modes of prediction unit

Each CTU can be divided into four smaller coding units (CU) in a recursive manner until the size of CU reaches the minimum allowed CU size. Each CU becomes the basic unit for coding process. An example of LCU splitting process is shown in Fig. 2.7. From this example, it should be noticed that not all CUs are partitioned as the smallest sized CU. The determination condition for further splitting a CU is that the total rate-distortion costs of four split CUs is smaller than the cost of parent CU.

The size of prediction unit (PU) can be varied from 64x64 to 4x4. For Intra prediction, only square size PU are used whereas for Inter prediction, PU can be split into square, rectangular and asymmetric shapes as shown in Fig. 2.8. Similarly, the transform unit (TU) used in HEVC can have different sizes, starting from 4x4 and going up to 32x32. For the 4x4 transform unit, discrete sine transform (DST) is applied and for other sizes of TU, discrete cosine transform (DCT) is used.

2.2.2. Reference Picture Set

In H.264/AVC the control signal for reference picture management contains only specific information about the changes in decoded picture buffer (DPB). The new reference picture management approach in HEVC, reference picture set, adds a complete list of reference pictures in each slice header [2, 10, 17]. With this concept, HEVC requires no information from earlier pictures in decoding order to maintain the correct reference pictures in DPB [2].

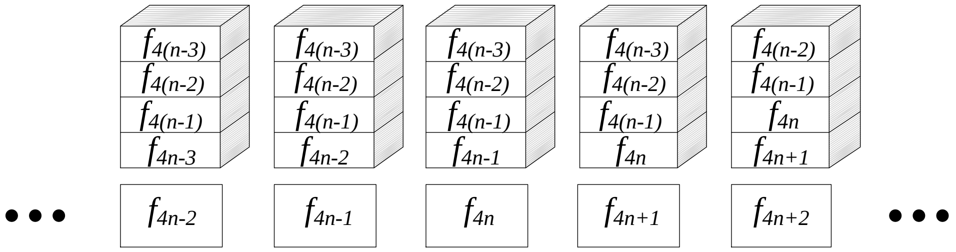


Figure 2.9 Reference picture set of HEVC

In H.264/AVC, the DPB is updated after a current picture has been decoded whereas in HEVC, the DPB is updated before decoding the current picture by using information from its slice header. It can also improve the error robustness of HEVC by avoiding usage of incorrect reference picture. The reference picture set concept for low-delay hierarchical-P coding structure is demonstrated in Fig. 2.9. There are four reference pictures in DPB for every frame, *f*, of the sequence except the first four frames. The frame index for each frame is described by a subscript. For instance, the notation f_{4n} represents the frame with a generalized index *4n*. If *n* is zero, it becomes the first frame of the sequence. The value can be zero or any positive integer.

It should be noted that some reference pictures are appeared in the DPB relatively shorter than others reference pictures. Normally, a frame its frame index is a multiple of four is kept in the buffer for a certain period in order to use as reference pictures and such kind of frames are called core frame in this paper. Other frames apart from core frames are called common frames. In the DPB, three core frames and one immediate past frame are stored. For encoding the core frame in hierarchical structure, the lowest QP value is used.

2.2.3. Intra Prediction and Coding

HEVC supports much more number of angular intra prediction modes than H.264: HEVC contains 33 directions whereas H.264 has only 8 directions. This can improve not only coding efficiency but also visual quality by reducing the ringing artifacts. A new interpolative prediction is also introduced such that the visual quality is enhanced by avoiding contouring artifacts.

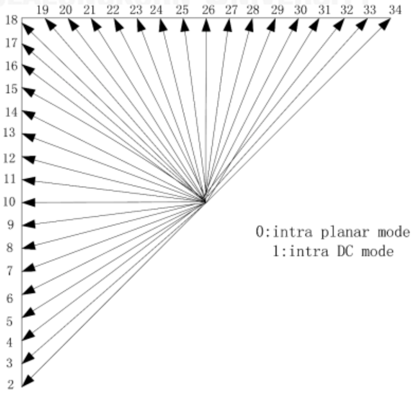


Figure 2.10 Intra prediction modes

2.2.3. Motion Estimation

The motion merge mode of HEVC is somehow similar to DIRECT mode of H.264/AVC. However, merge mode not only uses the motion information of neighbouring blocks but also exploits the flexible coding tree structure. Moreover, motion estimation in HEVC is designed for better parallel implementations. In order to achieve more precision, the interpolation filter length is increased from 6-tap to 8-tap. Although this interpolation filter can help improving the coding efficiency, it also increases the complexity of the encoder.

2.2.4. Entropy Coding

Context Adaptive Binary Arithmetic Coding (CABAC) is the only entropy coding method used in HEVC whereas H.264/AVC has both CABAC and Contex-Adaptive Variable Length Coding (CAVLC). The design of CABAC of HEVC contains several enhancements from that of H.264/AVC so that it can achieve better compression ratio and more friendliness to parallel architectures.

2.2.5. In-Loop Filtering

The deblocking filter of HEVC is modified to reduce complexity than that of H.264/AVC. This deblocking filter is similar to a smoothing filter to reduce the blocking artifacts that usually appear near the edges. To get better performance, HEVC introduces a new filter namely Sample Adaptive Offset (SAO) that can adaptively add offset value to each pixel and can also serve as a de-ringing filter. With the help of SAO, the visual quality of output video is improved.

2.3. Comparison of Error Resilience Tools in H.264 and HEVC

H.264/AVC provides some error resiliency schemes in Video Coding Layer (VLC) [3, 18]. They are:

1. Flexible Macroblock Ordering (FMO): In this tool, picture can be partitioned into regions (slice). Each slice can be independently decoded. The purpose of this tool is stopping the propagation of errors between slices.

2. Arbitrary Slice Ordering (ASO): Since each slice is independently decodable, slices can be sent and received out of order. This can improve end-to-end delay time on certain networks.
3. Data Partitioning: In H.264/AVC data partitioning mode, each slice can be segmented into header and motion information, intra information, and inter texture information by simply distributing the syntax elements to individual data units. This information is mapped into three partitions A, B and C.
4. Redundant Picture: A picture marked as “redundant” contains a redundant representation of part or all of a coded picture. In normal operation, the decoder reconstructs the frame from “primary” (non-redundant) pictures and discards any redundant pictures. However, if a primary coded picture is damaged, the decoder may replace the damaged area with decoded data from a redundant picture if available.
5. Switching Pictures: A new feature in H.264/AVC consists of picture types that allow exact synchronization of the decoding process of some decoders with an ongoing video stream produced by other decoders without penalizing all decoder with the loss of efficiency resulting from sending an Intra-coded picture. This can be enable switching a decoder between representation of the video content that used different data rates, recovery from data losses or errors as well as enabling fast-forward and fast-reverse playback functionality.

HEVC inherits some important error resilient tools from H.264/AVC such as slices, parameter set, NAL unit, etc. Other error resilience tools of H.264/AVC, such as flexible macroblock ordering, arbitrary slice ordering redundant slices, data partitioning, and SP/SI picture have been removed due to their rare usage [10, 17].

Among many SEI messages of H.264/AVC, scene information SEI message is still available in HEVC because it can assist detection of scene changes at the decoder. Moreover, HEVC introduces a new parameter set called video parameter set (VPS) which will use together with sequence parameter set (SPS) and picture parameter set (PPS).

In HEVC, reference picture management is different from that of H.264/AVC. Instead of using sliding window and adaptive memory management control (MMCO), a reference picture list construction (RPLC) based mechanism is used.

Clean Random Access (CRA) picture is the new picture type that introduces in HEVC. The main purposes of CRA are to support random access and to improve coding efficiency. CRA pictures in HEVC allows pictures that follow the CRA picture in decoding order but precede it in output order to use pictures decoded before the CRA picture as reference and still allow similar clean random access functionality as an instantaneous decoder refresh (IDR) picture.

2.4. Error Resilient Video Coding

Generally, techniques that have been developed for error resilient video coding can be divided into three categories: encoding only techniques, decoder error concealment, and interactive error control [9, 19].

Intra refresh encoding schemes improve the error robustness by reducing the temporal dependency between consecutive frames. Periodical inserting I-frame is a special case of intra refresh. However, coding too many I frames will generate a large amount of bits and cause a heavy bit rate fluctuation. Instead of encoding the entire frame as an I-frame, some blocks within the frame can be forced to encode with Intra mode. Those intra refresh methods include random intra refresh [20], periodical intra refresh [21], end-to-end rate distortion model-based intra refresh [20], and attention-based intra refresh [22]. Rate-distortion (RD) optimized intra refresh method produces better performance than random intra refresh method because coding mode selection for each block is done based on RD framework. So this method can improve error resiliency and can maintain the output bitrate. In attention-based intra refresh, the coding blocks in some important areas of the frame are intra coded such that subjective quality of the decoded video is improved.

Recursive optimal per-pixel estimate (ROPE) proposed in [23] estimates the end-to-end distortion by using a mathematical model for given transmission channel characteristics. In this method, per-pixel distortion is estimated by calculating the first and second order moments of its decoded value.

In layered coding or scalable coding [24], a video is encoded into a base layer and one or several enhancement layers. The base layer provides only acceptable but low level of quality and each additional enhancement layer can increase the level of output video quality.

In multiple description coding [25], a video is encoded into several sub-streams or descriptions. Each description is correlated and has similar importance. Therefore, every description can provide a basic level of quality. The quality of output video can be improved by combining multiple descriptions together. But this method is only suitable for systems that have two or more different transmission paths with different characteristics. Furthermore, when a video frame of one description is lost, it is recovered by approximating from timely nearby frames in other descriptions. However, typically not every error can be recovered. Hence, error propagation can still occur.

RD-optimized mode decision is used widely for error robustness when no feedback channel is available. When the decoder is able to communicate with the encoder via feedback message or back-channel message, feedback-based error resilient techniques becomes the most appropriate error resilient coding approaches. The comprehensive reviews of feedback-based error resilient techniques have been presented in [8, 9]. The two typical feedback-based error resilient methods are error tracking [8] and reference picture selection (RPS) [11, 12].

In error tracking method, the encoder can track the impact of the damaged areas in nth frame on decoded blocks in next frames (i.e., frame n+1 and onwards). To stop the error propagation, the encoder can encode the blocks in frame n+d that would have used for prediction from damaged area in frame n+d-1 by INTRA-mode. Another possible ways are to avoid using the affected area for prediction and to perform the same type of error concealment as the decoder for affected area before prediction.

In RPS, the encoder learns from feedback messages about damaged areas of a previous frame, it can stop the error propagation to the next frame by selecting an old reference picture that is also available at the decoder instead of choosing the most recent one.

An early proposed method for RPS is NEWPRED [11, 12] that uses the feedback information about correctly received packets or lost packets to reduce the effect of error propagation caused by corrupted pictures. There are two types of NEWPRED system: A-NEWPRED and N-NEWPRED. If the decoder sends back an acknowledgement signal (ACK) for every correctly received packet, the system is called A-NEWPRED and if a negative acknowledgement signal (NACK) is transmitted whenever the decoder detects a corrupted or lost packet, the system is called N-NEWPRED. Based on the feedback information, the encoder selects the reference picture so that the video quality drop caused by errors is suppressed.

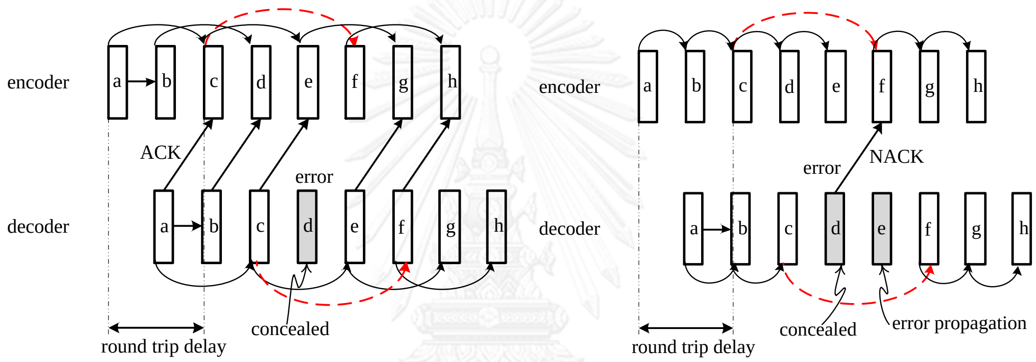


Figure 2.11 NEWPRED system for round trip delay of 2 frame intervals
(a) A-NEWPRED system (b) N-NEWPRED system

A-NEWPRED system and N-NEWPRED system for round trip delay of 2 frame intervals are illustrated in Fig. 2.11. Consider the case where the frame d is lost or corrupted, the encoder in A-NEWPRED system will know the status by receiving no ACK and that of N-NEWPRED system will know the status by detecting NACK before encoding frame *f*. Hence, the encoders in both systems will select the last correctly received frame as the reference frame, which is frame c in above example.

In case errors occur in the feedback channel, the A-NEWPRED system will react by selecting the last successfully decoded frame. This may slightly reduce in coding efficiency but does not affect the video quality much. However, in the N-NEWPRED system, NACK is lost due to error and the encoder does not recognize about the packet error at the decoder. The error at the decoder will propagate to next frames until the encoder receive another NACK. Therefore, the degradation in video quality is considerably large.

In [14], proxy-based reference picture selection method for mobile video telephony scenario is proposed. This method considers three different cases: the transmission involves only wireless uplink case, the transmission involves only wireless downlink case, and the transmission involves both wireless uplink and downlink case. For wireless downlink case, feedback signal from mobile device is sent back to base station when error occurs. The transmitter uses fix distance RPS scheme for encoding. If the base station receive feedback message, retransmission technique is used to stop error propagation. For wireless uplink case, the base station sends feedback message to the transmitter when packet loss occurs. The transmitter select appropriate reference picture from the buffer to stop error propagation. This approach is same with NEWPRED method with the extension of slice level reference selection. For the end-to-end wireless transmission case where both uplink and downlink involve wireless channels, the combination of above two methods is also presented.

Chenghao et al. [15] proposed a reference picture selection method by using long-term reference picture. This method considers for long feedback delay case. A feedback transmission method for Real-time Transport Control Protocol (RTCP) is firstly proposed. Then, reference picture selection method using long-term reference picture is developed. DPB management scheme is also provided. This method can improve quality in terms of PSNR for long delay cases but large DPB size is required. The errors can further supress at the decoder by applying error concealment techniques as a post processing step. Error concealment methods estimate and replace the missing data in an attempt to conceal errors in the decoded stream. There are several error concealment methods that have been proposed in the literature [19].

Spatial Interpolation [19] is a simple error concealment method used for recovering corrupted data. The intensity value of a single pixel is spatially interpolated by using intensity values of nearby pixels which are correctly decoded.

Another error concealment approach is using motion compensation (MC) temporal prediction for error recovery. This method copies the pixel values from the same spatial location of the previous frame. This method is effective only for some video scenes where no significant motion involves.

In motion compensated temporal interpolation method, the motion vector (MV) of damage block needs to be estimated. There are several approaches for MV estimation. One approach is to assume the lost MVs to be zeros. Other approaches include using the MVs of the corresponding block in the previous frame, using the average of the MVs from spatially adjacent blocks, using the median of MVs from the spatially adjacent blocks, etc.

2.5. Rate Control in Video Coding

Rate control is an important module of video encoder because the output bit rate needs to be maintained at given target bit rate which usually depends on applications. In the previous coding standard, rate control method in [26] is used. The primary purpose of rate control is to select the proper quantization parameters (QP) to regulate the output bit rate according to the channel status such that the encoder is able to maximize the video quality. Generally, rate control algorithms can be divided into two steps: bit allocation and QP computation.

In the bit allocation step, the suitable number of bits is assigned to each coding level. In general, these levels include group of pictures (GOP) level, picture level, and basic unit level. In this step, the buffer status is also taken into consideration.

In order to achieve the target bitrate, encoder needs to select proper coding parameters. Each coding parameter has more or less impact on the amount of generated bits of the coded video. Among many coding parameters, one of the most commonly used parameters is QP. While other coding parameters are set as fixed values, larger QP generally leads to smaller bitrate and vice versa.

In the design of rate control algorithms, Rate-Distortion (R-D) performance is one of the fundamental considerations. One of the critical problems in rate control is to estimate the R-D function of the video to be encoded.

Some rate control algorithms are designed based on the relationship between bitrate (R) and quantization (Q). This kind of rate control methods are called Q-domain rate control [27]. In this approach, it is assumed that an encoder can select the corresponding QP value to achieve the desired target bitrate. That means QP is the critical parameter on determining the amount of generated bits. The typical R-D model of Q-domain rate control is

$$R = aQ^{-1} + bQ^{-2} \tag{1}$$

where *a* and *b* are two model parameters related to the contents of input video. This model is also known as quadratic R-D model. Rate control in reference software [26] of H.264/AVC adopts this model.

Another group of rate control algorithms use the relationship between the percentage of zeros among the quantized transform coefficients (ρ) and coding bitrate. This kind of rate control algorithms is called ρ-domain rate control algorithm [28]. It is assumed that there is a one-to-one correspondence between ρ and QP. Hence, this assumption also implies QP is the critical factor on determining desired target bitrate. The typical R-D model of ρ-domain rate control is

$$R = \theta \cdot (1 - \rho) \tag{2}$$

where θ is a parameter related to the video content. As aforementioned, Q is the critical factor for both Q-domain rate control and ρ-domain rate control. Since there are many coding parameters that have impact on the out bitrates, varying Q is only effective when other parameters are fixed. Otherwise, the performance may not be accurate.

The coding tools of HEVC are designed to have high flexibility in order to adapt various video contents in various services. To overcome the limitations of Q-domain rate control and ρ-domain rate control, a new λ-domain rate control [29] is proposed. There is a more robust correspondence between λ and R. The relationship between λ and R can be expressed by using a Hyperbolic function.

$$D(R) = CR^{-K} \tag{3}$$

where C and K are model parameters related to the characteristics of the source.

In this section, a brief review of R-lambda rate control method [29] which is adopted in current reference software of HEVC is provided. It can configure as either frame level or CTU level rate control. This method can be divided into two main parts: bits allocation and adjusting encoding parameters to achieve allocated bits.

For the bit allocation, there are GOP level, picture level, and LCU level bit allocation processes. In GOP level bit allocation, both target bitrate and current buffer status are taken into account.

$$T_{CurrPic} = \frac{T_{GOP} - Coded_{GOP}}{\sum_{NotCodedPictures} \omega p_i} \cdot \omega p_{CurrPic} \tag{4}$$

$$T_{CurrLCU} = \frac{T_{CurrPic} - Bit_{header} - Coded_{Pic}}{\sum_{NotCodedLCUs} \omega b_i} \cdot \omega b_{CurrLCU} \tag{5}$$

where ϖp is the weight of each picture, T_{GOP} is target bits for current GOP, $T_{CurrPic}$ is target bits for current picture, $Coded_{GOP}$ is generated bits for coded frames in current GOP, and ϖb is the weight of each LCU. For picture level, weight value ϖp is depending on the position of picture in hierarchical coding structure. In LCU level equation, $Coded_{Pic}$ and Bit_{header} represent generated bits for coded LCUs in current frame and generated header bits, respectively.

From the Hyperbolic R-D function, the slope of R-D curve, λ, is calculated by

$$\lambda = -\frac{\partial D}{\partial R} = CK \cdot R^{-K-1} = \alpha R^{\beta} \tag{6}$$

where α and β are parameters related to the video source. Every frame as well as every CTU has its own parameters. By using this two parameters and bit per pixel (bpp) value, lambda can be computed by (6). Depending on the target bit rate, bpp value varies.

$$\lambda = \alpha \cdot bpp^{\beta} \tag{7}$$

Once lambda is obtained, QP value for respective frame or CTU can be computed by using equation (7).

$$QP = 4.2005 \ln \lambda + 13.7122 \tag{8}$$

After encoding each frame or a CTU, the corresponding α and β values are updated based on actual generated bits, QP value and λ value by using (9) to (11).

$$\lambda_{comp} = \alpha_{old} \cdot bpp_{real}^{\beta_{old}} \tag{9}$$

$$\alpha_{new} = \alpha_{old} + \delta_{\alpha} \cdot (\ln \lambda_{real} - \ln \lambda_{comp}) \cdot \alpha_{old} \tag{10}$$

$$\beta_{new} = \beta_{old} + \delta_{\beta} \cdot (\ln \lambda_{real} - \ln \lambda_{comp}) \cdot \ln bpp_{real} \tag{11}$$

where bpp_{real} is obtained from actual generated bits. α_{old} and β_{old} are α and β values used in coded frame. δ_{α} and δ_{β} are 0.1 and 0.05, respectively.

CHAPTER 3
ADAPTIVE ERROR-RESILIENT TECHNIQUES FOR HEVC

In this chapter, proposed feedback-based error resilient algorithm for low delay conversational applications is explained in details. Moreover, a feedback-based RPS method for HEVC low-delay-P framework called hierarchical-P RPS is also presented. This hierarchical-P RPS is served as reference method for evaluating the proposed algorithm.

3.1. Introduction

The main advantage of interactive error control which uses feedback messages from the decoder to mitigate the network error is its ability to inherently adapt the varying packet loss rates. This technique is suitable for applying communication system where feedback or back-channel message is available. Due to the bidirectional nature of conversational services like video telephony, feedback-based error resilient techniques are commonly used. The performances of these techniques are good when the feedback delay is low. The typical one-way latency requirement for such application is normally in the range of 150-250 ms.

RPS is a low complexity feedback-based error resilient technique which is suitable for applications with delay constraint. If the feedback delay is low enough, RPS can stop spatio-temporal error propagation completely. Many reference pictures are stored in DPB in RPS algorithm. The encoder selects the suitable reference picture from the DPB before encoding current frame. This selection is done according to error status from the decoder. DPB management is also important for RPS technique. In the previous video coding standard, DPB management is carried out by sliding window method or by using memory management control operation.

However, HEVC introduces new reference picture management tool called reference picture set and selects hierarchical-P coding structure with multiple reference pictures for motion compensation. Hence, previous RPS methods are required to modify in accordance with HEVC framework. If the delay is large, packet errors propagate to several frames. If the time is long, the effect of accumulated error increases considerably and user gets annoying. In this thesis, a feedback-based RPS method for hierarchical-P coding structure is proposed in order to adapt with HEVC framework and a new error resilient algorithm that combines hierarchical-P RPS with ROI-based intra refresh is proposed. The hierarchical-P RPS algorithm and proposed error resilient algorithm are explained in next sections.

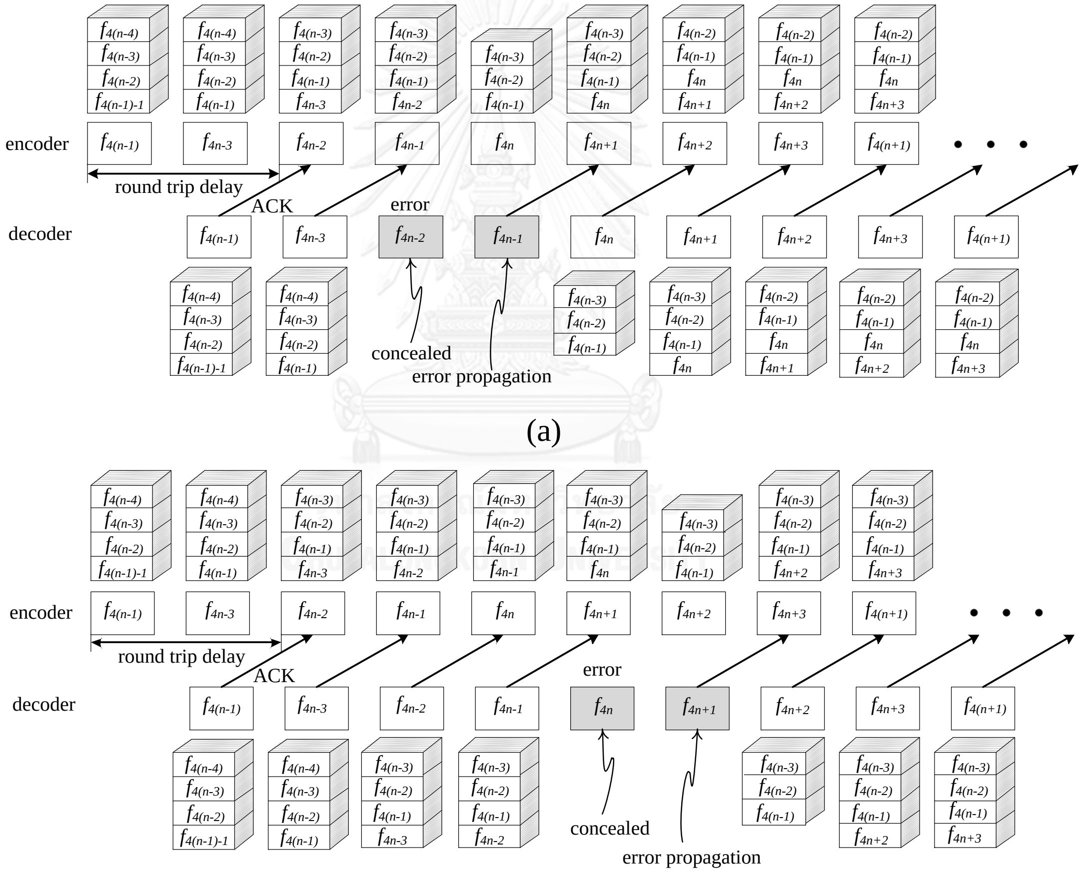
3.2. Hierarchical-P RPS system for HEVC

A conventional RPS method, A-NEWPRED [11, 12], is selected to use as reference method in this work because it shows good performance when the feedback delay is low and its performance does not affect by feedback channel error. The review of RPS methods can be found in section 2.3.

As aforementioned, A-NEWPRED and other previous RPS methods are mostly designed for IPPP coding structure and use only one reference picture for motion compensation although multiple reference pictures are stored in the buffer. On the other hand, the default low-delay-p configuration of HEVC uses hierarchical-P coding structure with multiple reference pictures. Moreover, HEVC uses new reference picture set feature for DPB management. In order to adapt with HEVC framework, the conventional RPS method, A-NEWPRED, is modified. This modification has two parts: reference picture selection and DPB management. For reference picture selection, the concept is same as previous methods. The encoder removes the unreliable reference picture from the buffer based on the feedback information. If the index of a reference picture is greater than or equal to the index of error picture, this reference picture is defined as unreliable reference picture.

Since the default configuration of HEVC low-delay-p keeps four reference pictures in the DPB for each picture, the DPB management part of hierarchical-P RPS algorithm also maintains four reference pictures in the buffer. To accomplish this task, encoder looks for some reliable pictures that are still remained in the DPB. The reliable picture can be either previous core frame or common frame. For better explanation, a common frame loss scenario and a core frame loss scenario are discussed.

The example in Fig. 3.1(a) shows a condition in which a common frame has error. In this example, frame f_{4n-2} has error so the ACK signal for that frame is not transmitted. Due to round trip delay, the encoder can detect this error only before encoding frame f_{4n}. Normally, the reference picture set of that frame will include frames { f_{4n-1}, $f_{4(n-1)}$, $f_{4(n-2)}$, $f_{4(n-3)}$}. But frame f_{4n-1} is considered as an unreliable reference frame because this frame is encoded by using the error frame (f_{4n-2}) as reference. To avoid possible error propagation, frame f_{4n-1} is removed from the current reference picture set. Hence, the new reference picture set for frame f_{4n} is { $f_{4(n-1)}$, $f_{4(n-2)}$, $f_{4(n-3)}$}. The frame f_{4n} can be used as reference for upcoming frames. If no error occurs for upcoming frames, the DPB updating process can proceed by keeping three core frames and one common frame in the buffer.



(a)

(b)

Figure 3.1: Modified A-NEWPRED system (a) a common frame loss and (b) a core frame loss.

The updating process is a little bit more complicated when a core frame was hit by error as shown in Fig. 3.1(b). After handling the detected error by removing one or more reference pictures from the list, the encoder is locking for the new possible member of the list so that it can keep four reference pictures for each frame. The new member can be either core or common frame which is still available in the DPBs of both encoder and decoder. If both core and common frames are available, core frame has given the first priority. The example in Fig. 3.1(b), reference picture $f_{4(n-3)}$, for error free case, will be removed from reference picture set of frame f_{4n+2}. But in this example, it is still served as reference picture in the DPB because it is the only reliable picture that remained in DPBs of both encoder and decoder. At frame $f_{4(n+1)}$, it is possible to substitute reference frame $f_{4(n-3)}$ with frame f_{4n+2}. That means the DPB has two reliable pictures and has to select only one picture for current frame. The encoder decided to keep frame $f_{4(n-3)}$ because it is core frame and marked frame f_{4n+2} as "unused for reference". The encoder will continue to use frame $f_{4(n-3)}$ as reference until the ACK signal of next core frame is detected.

3.3. Proposed interactive error control for HEVC

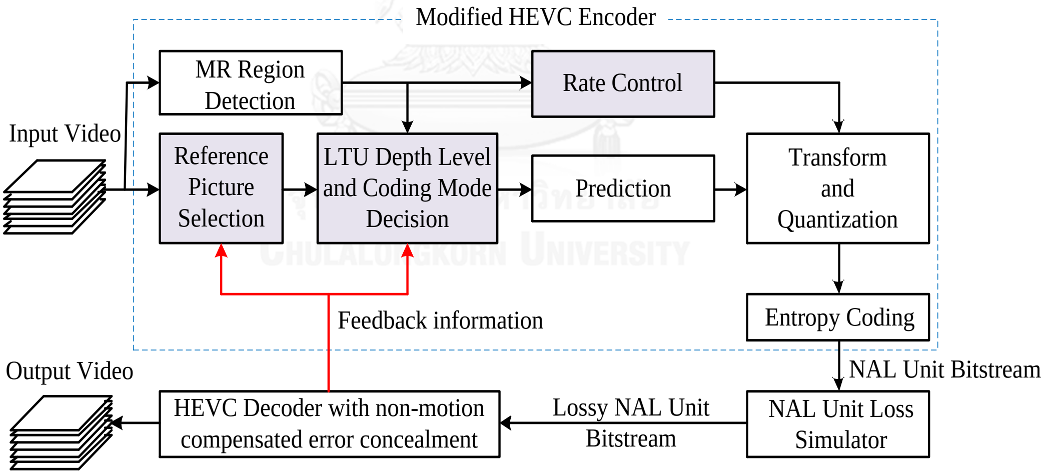


Figure 3.2: Overall block diagram of a feedback-based error resilient video codec.

The overall block diagram of HEVC video transmission system is shown in Fig. 3.2. A feedback channel is assumed such that the feedback messages are well protected and are error free. The proposed error-resilient algorithm requires two inputs: feedback message and ROI information. Since moving region is served as ROI in this research, MR-map generation step is firstly applied before the picture partitioning step. MR-map is used for both LCU depth level decision and mode decision processes. Based on the feedback information and MR-map, coding mode for each LCU is selected. If Intra mode is chosen for current LCU, the modified parameter updating process of the rate control is called. The output of the encoder is NAL unit bitstream. In this thesis, each NAL unit is considered as a data packet. For packet loss simulation, NAL unit loss simulator [30] is used with predefined loss patterns. Since the current HEVC reference software cannot handle corrupted bitstream, a co-located block copying technique is added to the decoder for error concealment.

3.3.1. Moving region extraction

Most of the cases, a moving region attracts more attention in a video frame. In order to extract the moving region, frame differencing method is used. But to reduce the effect of camera motion, motion compensation is done to the previous frame by using a global motion vector (GMV). GMV is obtained by Gray Projection Method (GPM) [31]. GPM is a simple and effective method to estimate global motion vector. A two-dimensional frame is projected into two independent one-dimensional projection curves: column projection curve and row projection curve. The maximum cross-correlation of the projection curves between the previous frame and the current frame are computed. GMV is estimated by maximum cross-correlation value. Since GPM relies on statistical information, it has robustness of local object movements. In addition, the computation cost of GPM is quite low and it is suitable for real-time applications. Pixels from previous frame are motion compensated by using estimated GMV. Then the difference between current frame and previous frame for each coding block is calculated by

$$\Delta_k(p) = \frac{1}{N_{block}} \times \sum_{(i,j)\in p} \left| F_k(i,j) - F_{k-1}(i + GMV_k^x, j + GMV_k^y) \right| \qquad (12)$$

where $\Delta_k(p)$ denotes the difference values of pixels for p^{th} block of k^{th} frame. The size of block is same as the size of LCU in encoder configuration. N_{block} is the number of pixels in block p. $F_k(i,j)$ is the luminance value of the pixel (i,j) in the k^{th} frame.

(GMV_k^x, GMV_k^y) represent the horizontal and vertical components of the GMV, respectively.

MR is extracted with a predefined threshold based on the weighted averaging of pixel difference as shown in (13). The weight values, ω_{MR}, are assigned based on the location of the block. A frame is divided into three regions: central region, border region, and transition region which is the region in between central region and border region. Different ω_{MR} values are heuristically assigned for different regions. The weight values for this research are 1.0 for central region, 0.1 for border region, and 0.55 for transition region, respectively.

$$MR_k(p) = \begin{cases} 1, & \text{if}\, \omega_{MR} \times \Delta(p) / \Delta_k^{avg} > th, \\ 0, & \text{otherwise.} \end{cases} \tag{13}$$

where Δ_k^{avg} represents the average difference value of blocks for frame k, *th* stands for threshold value. Threshold value is also heuristically chosen as 0.75. If the value of $MR_k(p)$ is 1, the block *p* is in moving region or ROI for this thesis.

(a) (b)

Figure 3.3: Output of moving region extraction procedure. (a) Johnny; (b) BQMall

3.3.2. Early termination of coding tree splitting process

In order to adopt the various video contents which may contain large smooth areas or small areas with lots of details, HEVC uses quad-tree based variable-size CTU rather than fixed size marcoblock of previous standards. The largest configurable size of CTU is 64x64 and the smallest size is 8x8. Therefore, if the LCU size is 64x64, there are four depth levels for this CTU to split into smaller size CU.

CU partitioning is done in a recursive manner. A CU can be divided into four equal size CUs in lower depth level until it reaches the lowest level. Each CU can be used as basic unit for encoding process. Thus, the size of basic unit in HEVC is always varied depending on the complexity of the video content and coding mode. The decision for further splitting of a CU is done by comparing rate-distortion cost of CU at current level and sum of four rate-distortion costs of CUs at lower level. This process requires a large amount of computation time although it can achieve the best rate-distortion performance. However, HEVC allows setting of maximum depth level for each LCU so that the splitting process cannot go beyond that level. This LCU depth level option may reduce computation time.

Since MR region usually will get more attention than non-MR region, the depth level of CUs in non-MR region is decided to limit. If the depth level of CUs in non-MR region is limited, this will be able to save a lot of encoding time with some quality loss. But according to HVS, the loss of some details in non-MR region has only a little impact on user experience. Therefore, a constraint on maximum depth level of LCUs in non-MR region is added. Based on MR-map, the depth level of each CTU is computed by

$$DL(p) = \begin{cases} \max\_\ depth, & \text{if } p \in MR, \\ 1, & \text{otherwise.} \end{cases} \tag{14}$$

where *DL(p)* is depth level for p^{th} block or LCU, the value of max_*depth* depends on the size of LCU in encoder configuration. For 64x64 LCU, max_*depth* is 4.

3.3.3. Coding mode selection for LCUs in MR

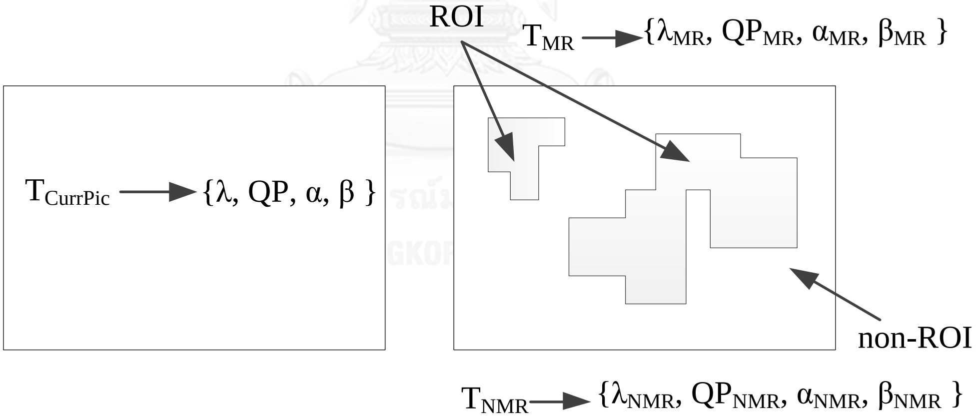
If error is detected from the feedback information, the reference picture buffer is updated according to modified RPS algorithm in section 3.2. The RPS algorithm is very fast to response the error but due to feedback delay, error propagation still occurs in some frames and that will affect the quality of perceived video. Moreover, if the temporal distance of reference picture is too large, the coding efficiency is also affected. To stop the error propagation and to enhance the quality especially in the MR region, a CTU-level intra coding algorithm is proposed, which forced to encode

CTUs in MR region with intra mode if the error is detected and the distance from the last intra refresh frame is greater than equal to 4. It should be noted that inserting too much intra coded blocks will affect the overall performance of the decoded video when encoding under rate constraint. To ensure this, the distance from last intra refresh frame should be high enough. This intra refresh help reducing error propagation when the feedback delay is long. Coding mode, *CM*, of each CTU can be determined by

$$CM(p) = \begin{cases} Intra, \text{if } p \in MR \text{ and } errorfound \text{ and } idst \geq 4, \\ Inter, \text{otherwise.} \end{cases} \tag{15}$$

where *idst* is the distance from last intra refresh frame, *errorfound* is true when an error is detected from feedback message.

3.3.4. Modification of rate control



(a) (b)

Figure 3.4 Bit budget and parameters of rate control in HEVC. (a) Original R-lambda rate control; (b) Modified ROI-based rate control

Because of ROI, a picture is divided into two regions. Since ROI region can attract more attention, the quality of this region should be enhanced by assigning more bits to each block in that region. In order to maintain the target bit budget of a frame, the blocks in non-ROI region allocate fewer bits than the blocks in ROI region. Hence, the target bit budget equation for a frame can be rewritten as

$$T_{CurrPic} = T_{MR} + T_{NMR} \tag{16}$$

where TMR is target bits for moving region, TNMR is target bits for non-moving region, and TCurrPic is target bits for current picture which is obtained by equation (4). For ROI-based rate control, different α, β, and λ values are required for different region in a frame. If the LCUs in MR region are decided to encode with Intra mode, the allocated bits for that LCUs are required to refine. Firstly, target bit rate for MR region is determined from the available bit budget of current frame by

$$T_{MR} = \frac{(T_{CurrPic} \cdot N_{MR})}{N_{Pic}} \tag{17}$$

where N_{Pic} is total number of pixels in whole frame, and N_{MR} is number of pixels in MR region.

T_{MR} value is used for assigning new bit budget for Intra coded MR region. The bit refinement process is the same process as in reference software. The only difference is instead of giving the bit budget for an entire frame as input, we use bit budget for MR region. The α and β values assigned for Intra frame are used for MR region where as the α and β values assigned for current frame are used for non-MR region. By using these α and β values, λ and QP values for each region are computed.
After encoding one frame or LCU, model parameters are updated. For parameter updating, bpp_{real} from actual generated bits is computed. In our approach, there are two bpp_{real} values: bpp_{real_MR} and bpp_{real_NMR}. The two average λ_{real} values, λ_{real_MR} and λ_{real_NMR}, are also computed.

Parameters for each region can be updated by (9) to (11) where δ_α and δ_β are 0.1 and 0.05, respectively.

3.3.5. Summary of proposed error resilient algorithm

The proposed algorithm is summarized as follows:

(1). Apply modified A-NEWPRED reference picture selection algorithm if an error is detected from feedback information.

(2). Depth level decision for each LCU is carried out based on MR-map and error status.

(3). Coding mode for MR region is selected according to error status and temporal distance of last encoded I frame/MR region.

(4). If coding mode for MR is Intra Mode, bit refinement process for MR region is executed before encoding and region-based parameter updating process is used after encoding.

The proposed error resilient algorithm is described in pseudo code as follow:

```
Algorithm: Proposed Error Resilient Algorithm
Detect MR and Generate MR-map
Read feedback
if ACK signal is detected then
    error_found = true
else
    error_found = false
end if
Apply RPS error resilient method
for all CU in frame do
    if CU is in MR and error_found and $idst \geq 4$ then
        Depth = max_depth level
        Set CU coding mode to intra
        Apply Bit budget refinement for MR
    else if CU is in MR and $idst \geq 12$ then
        Depth = max_depth level
        Set CU coding mode to intra
    else
        Depth = 1
        Set CU coding mode to inter
    end if
end for
Process Next frame
```

CHAPTER 4

EXPERIMENTAL RESULTS AND DISCUSSIONS

In this chapter, the performance of proposed algorithm is evaluated by comparing with modified version of conventional RPS algorithm (i.e., modified A-NEWPRED from section 3.2). The impacts of target bitrates, video contents, and packet loss rates on the quality of decoded video are analyzed and discussed. There are two main sections in this chapter. In the first section, the test sequences and simulation setup are explained. In the second section, the discussions of experimental results are provided.

4.1. Experimental Setup

Both modified RPS algorithm and proposed algorithms are implemented by using HEVC reference software HM15.0 [32]. Six HEVC test sequences were used in the experiments, with different spatial and temporal characteristics, resolutions, and frame rate. A sample frame for each test sequence is shown in Fig. 4.1. Each video is 10 second long and stored as raw, progressively scanned video file, with YCbCr 4:2:0 colour sampling and 8 bits per sample.

(a) BasketBallDrillText (50fps) (b) BQMall (60fps) (c) PartyScene (50fps)

(d)FourPeople (60fps) (e) Johnny (60fps) (f) KristenAndSara (60fps)

Figure 4.1: Sample frames of HEVC test sequences used in experiments. Sequences (a) – (c) and (d) – (f) have a resolution of 832x480 and 1280x720 pixels, respectively.

All experiments are carried out by using “lowdelay_P_main configuration” of HEVC in which GOP size is 4 frames and LCU size is 64x64 pixels. Multiple-slices-per-frame configuration is used for all simulations. Since flexible macroblock ordering tool is not included in HEVC, a slice in HEVC is only a group of LCUs in raster scan order. There are three options to determine the size of a slice in HEVC: number of LCUs per slice, number of bits per slice, and number of tiles per slice. In this thesis, slice structure based on number of LCUs per slice argument is used. Number of LCUs per slice depends on the resolution of the input video. Blocks per slice values used in this work are 13 for WVGA sequences and 20 for 720p sequences, respectively. Rate control is turned on and other remaining parameters are set as default.

Table 4.1: Summary of encoder configurations

	RPS_noI and Proposed	RPS_wI
Profile	main	Main
Maximum LCU size	64x64	64x64
Maximum LCU Partition Depth	4	4
Intra Frame Period	-1 (First frame of the sequence)	48 (for 50fps sequences) 60 (for 60fps sequences)
GOP Size	4	4
Number of active reference pictures in DPB	4	4
Slice Mode	1	1
Argument for Slice Mode	13 (for WVGA sequences) 20 (for 720p sequences)	13 (for WVGA sequences) 20 (for 720p sequences)
Rate Control	Enable	Enable

Modified A-NEWPRED RPS algorithm is used as reference method. Two configurations of A-NEWPRED algorithm, RPS_nI and RPS_wI, were used. In RPS_nI configuration, only the first frame of the sequence is encoded as Intra frame but in RPS_wI configuration, a regular Intra frame is inserted approximately every one second. The detail encoder configurations for each algorithm are summarized in Table 4.1. Each sequence is encoded with five different target bit rates as recommended in [33]. Target bit rates are listed in Table 4.2.

Table 4.2: Target bit rates

Resolutions	Rate 1	Rate 2	Rate 3	Rate 4	Rate 5
WVGA sequences	834 kbit/s	512 kbit/s	768 kbit/s	1.2 Mbit/s	2Mbit/s
720p sequences	256 kbit/s	384 kbit/s	512 kbit/s	850 kbit/s	1.5 Mbit/s

All experiments are carried out for three different packet loss rates, 3%, 5%, and 10%, where the packet loss trace files and NAL unit loss software are obtained from [30]. These packet loss trace files are also used as feedback inputs for encoder. To introduce the round trip time in the simulation, the input from packet loss trace file is updated only after some delay. The round trip delay for all our experiments is set as four frames period which is equivalent to 80ms for 50fps sequences and 67ms for 60fps sequences, respectively.

Packet errors are introduced to the encoded bitstream by using NAL unit loss software. Since current HEVC decoder cannot handle corrupted bitstreams, it is modified by adding a error concealment procedure such that it can decode lossy bitstreams. For error concealment, a simple collocated block copying technique is added at the decoder.

4.2. Result Analysis

Each test sequence is encoded with five different bit rates for each packet loss rate. Rate-distortion performances under different packet loss rates for each sequence are shown in Fig. 4.2 to Fig. 4.4. Firstly, the performances of two configurations of modified A-NEWPRED algorithm are compared. RPS_noI configuration performs better than RPS_wI for almost all cases. This is because RPS_wI introduces lots of intra coded blocks that consumes large portion of available bit budget and as a result large QP values are used to meet the bitrate constraint. But in PartyScene sequence, it is found that the performance of RPS_wI is better than RPS_noI under 10% PER. When error occurs, RPS_noI may select common frame as reference. For this case, complex scene of PartyScene sequene and low quality reference picture can cause low quality output picture. Moreover, round trip delay can cause additional error propagation until next control frame is received. This kind of error propagation is more likely to happen very often under high packet error rate. But for RPS_wI configuration, the regular introduction of I frame not only reduce the temporal distance between current frame and reliable reference frame but also improve the quality of reference frame. Therefore, for PartyScene sequence at 10% PER, the RPS_wI outperforms RPS_noI.

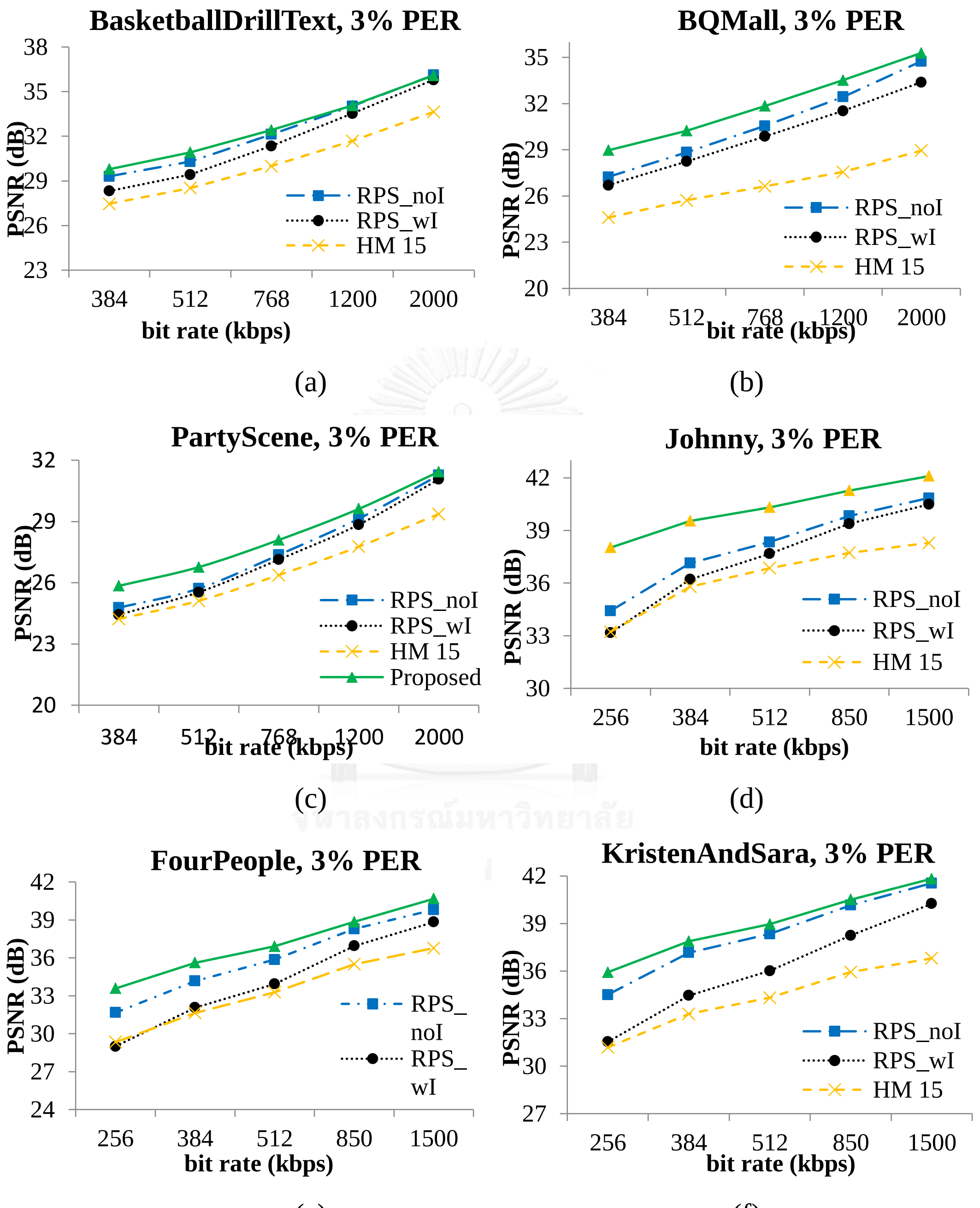
For all cases, the performance of proposed algorithm is better than both RPS_noI and RPS_wI. In the proposed algorithm, the number of intra coded blocks is significantly less than RPS_wI. Therefore, the highest value of QP is not required to use for achieving target bit rate. The rate control can maintain the quality of inter frames. Moreover, intra coded blocks can stop error propagation that reduces motion prediction error and can also enhance quality in MR region. It is noted that modified rate control scheme works well and it produces only negligible amount of extra bits.

The quality of proposed method is more noticeable in low target bit rates than high ones for all packet error rates. This is because as the target bit rate increased, the bit budget for each frame also increased. As a result, the quality difference between frames in hierarchical coding structure is getting narrow. So, using core frame as reference picture and common frame as reference picture has only small different. Normally, the strength of core frame is its quality because of using low QP but the weakness is its temporal distance from the current frame which is higher than four frames in current HEVC configuration.

For WVGA sequences, the average PSNR improvement for all test cases is 0.7 dB. The maximum and minimum values of improvement are 1.57 dB and -0.03 dB, respectively. In BQMall sequence, the performance improvement of proposed algorithm is significantly higher than modified RPS algorithm in all bit rates. In PartyScene sequence, the improvement can be seen more clearly at low bit rate cases than high bit rate cases. As the bit rate increased, the quality of common frame also increased. Thus, the quality of predicted picture using common frame also increased. As a result, the performance gap between proposed and modified RPS is narrow. But for BasketballDrillText sequence where many players running in the scene, the performance of proposed algorithm is almost same with that of RPS_noI for most cases. In low target bit rates cases, the proposed algorithm shows some improvement, however, these improvements are about 0.5 dB only. For the highest target bit rate (2Mbps), even RPS_noI is slightly better than proposed algorithm about 0.03 dB for 3% PER and 5% PER cases. But for 10% PER, proposed algorithm shows PSNR improvement about 0.11 dB over RPS_noI at the highest target bit rate. The performance gap between proposed algorithm and RPS_noI is very narrow in BasketballDrillText sequence because proposed algorithm focuses on the improvement in MR region which gives more favour for the blocks in the central region of the frame by assigning more weights for that area in this thesis. The moving objects in BasketballDrillText sequence locate not only in central region but also in border region. Therefore, some moving objects are not covered by the detected region. As a result, the errors in that region only handle by control picture of RPS algorithm. High motions and feedback delay cause error propagation more significant. Since the errors in that region have huge impact on the video quality, the performance improvement of proposed algorithm is small.

For 720p sequences, the average PSNR improvement of proposed algorithm over RPS_noI for all test cases is 1.35 dB. The maximum PSNR improvement is 3.61 dB while the minimum value is 0.28 dB. For Johnny sequence, the performance improvement of proposed algorithm is considerably high. This is because the content of this sequence has only one person with very few motions and the most important region (i.e., face) is located at the central region of the frame. So the proposed algorithm is effective for all target bit rates and all packet error rates. For KristenAndSara sequence, there are two people in the scene but they are located a little bit outside the central region. So the average PSNR improvement of the proposed algorithm for all cases is only 0.82 dB with maximum improvement of 1.42 dB and minimum of 0.28 dB. For FourPeople sequence, two people are in central region and other two are outside the central region. The improvement for this sequence is higher than KristenAndSara sequence but lower than Johnny sequence.

Average PSNR improvement is 1.21 dB with maximum of 1.91 dB and minimum of 0.54 dB.

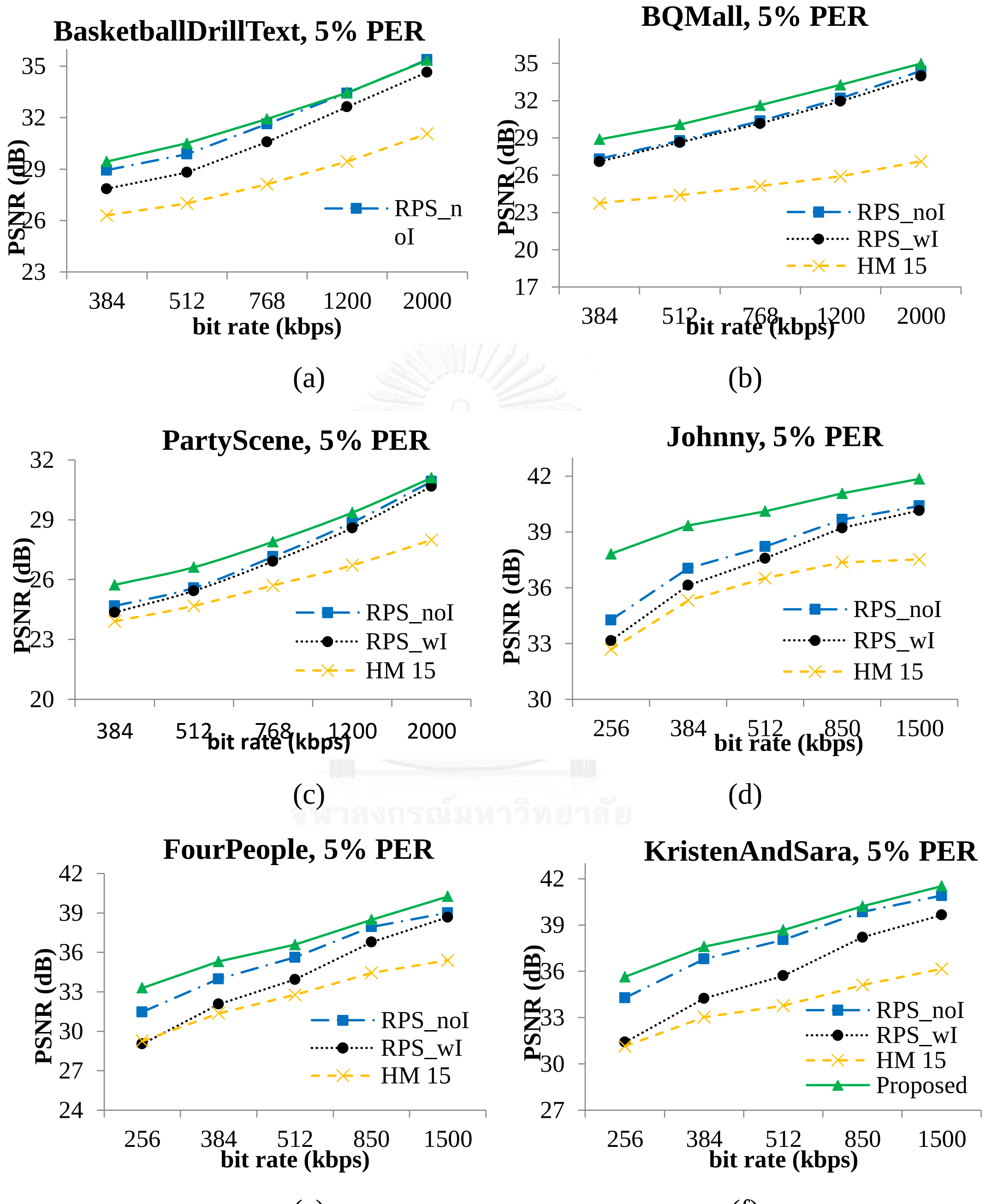


(a) (b)

(c) (d)

(e) (f)

Figure 4.2: Rate-distortion performance between RPS and proposed method for 3% PER. (a) BasketballDrillText; (b) BQMall; (c) PartyScene; (d) Johnny; (e) FourPeople; (f) KristenAndSara

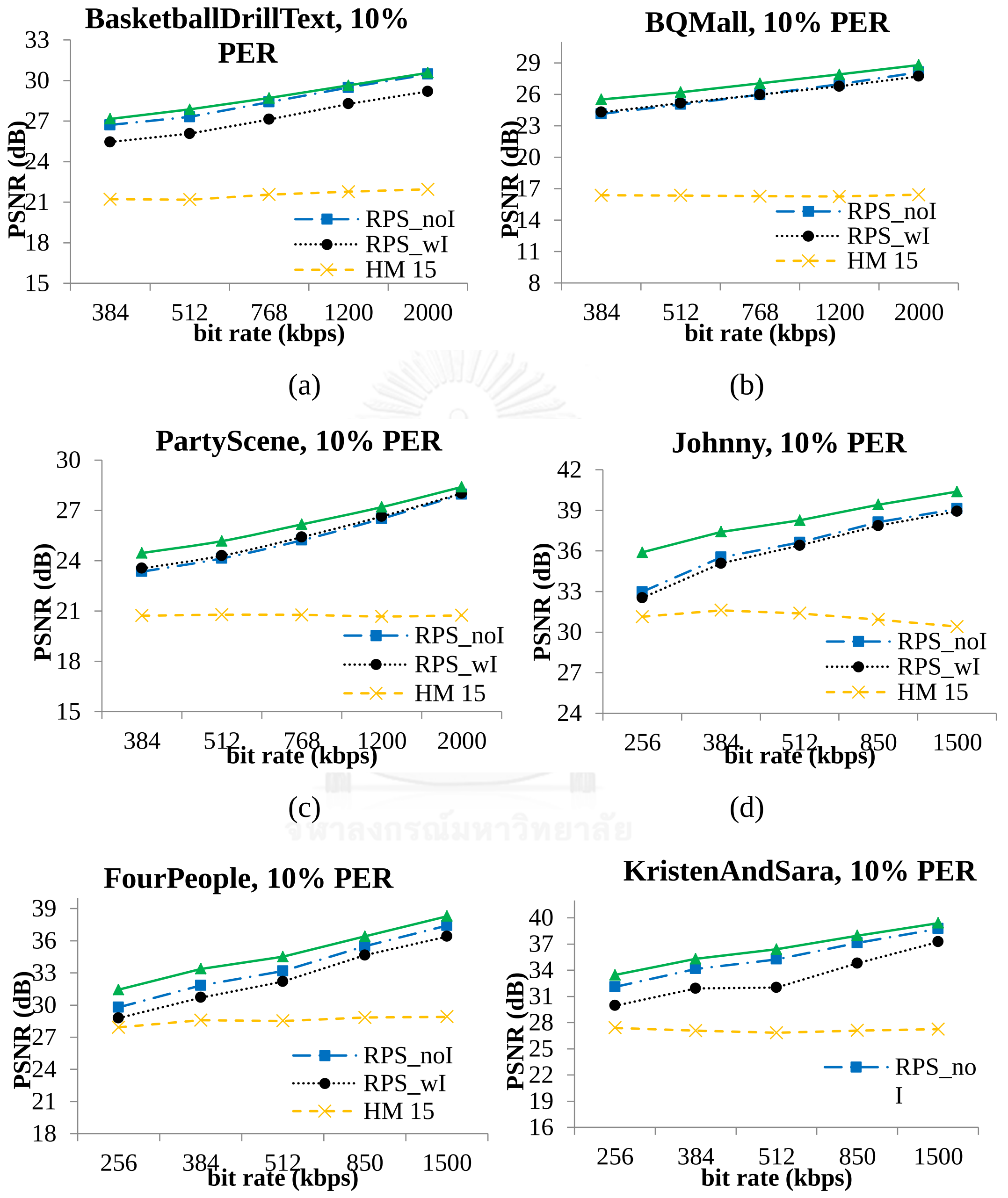


(a) (b)

(c) (d)

(e) (f)

Figure 4.3: Rate-distortion performance between RPS and proposed method for 5% PER. (a) BasketballDrillText; (b) BQMall; (c) PartyScene; (d) Johnny; (e) FourPeople; (f) KristenAndSara



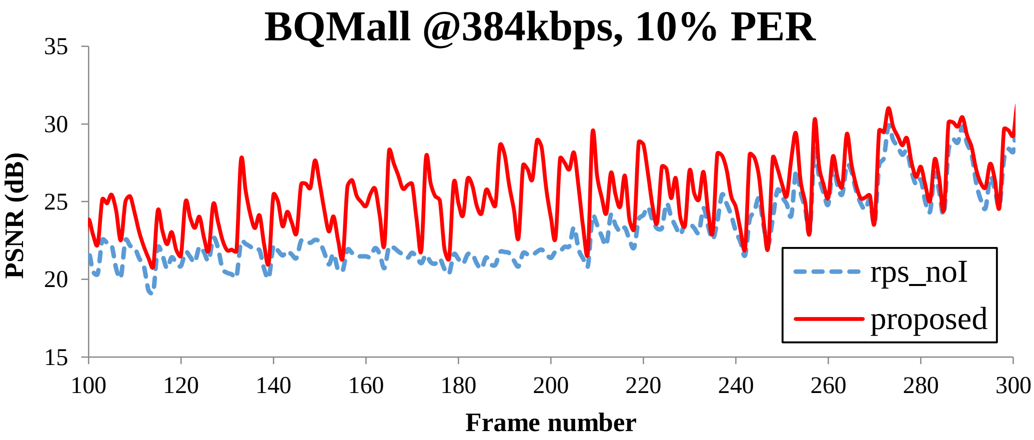
(a) (b)

(c) (d)

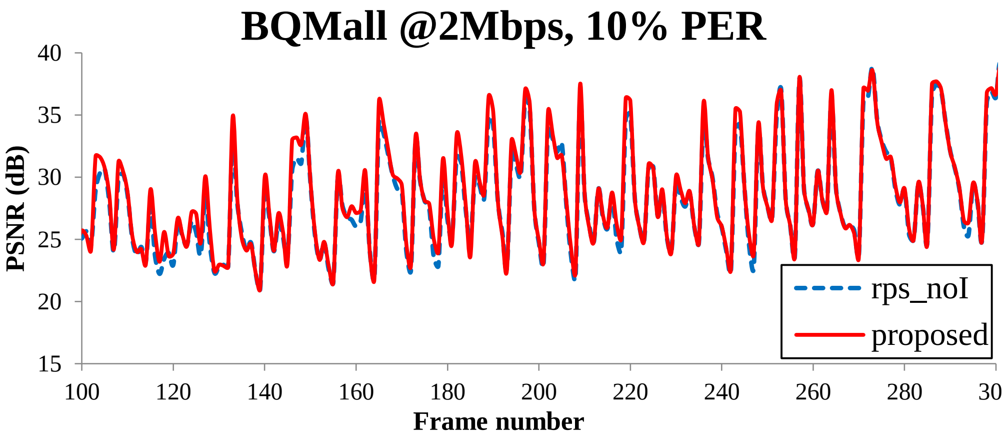
(e) (f)

Figure 4.4: Rate-distortion performance between RPS and proposed method for 10% PER. (a) BasketballDrillText; (b) BQMall; (c) PartyScene; (d) Johnny; (e) FourPeople; (f) KristenAndSara

The comparison of frame-by-frame PSNR values between proposed algorithm and RPS_noI algorithm for some test sequences are shown in Fig. 4.5 to 4.. For low bit rate case, almost every frame of each sequence encoded with proposed algorithm has significant PSNR improvement over that of RPS_noI. PSNR improvement can be seen at only some frames for high bit rate case.

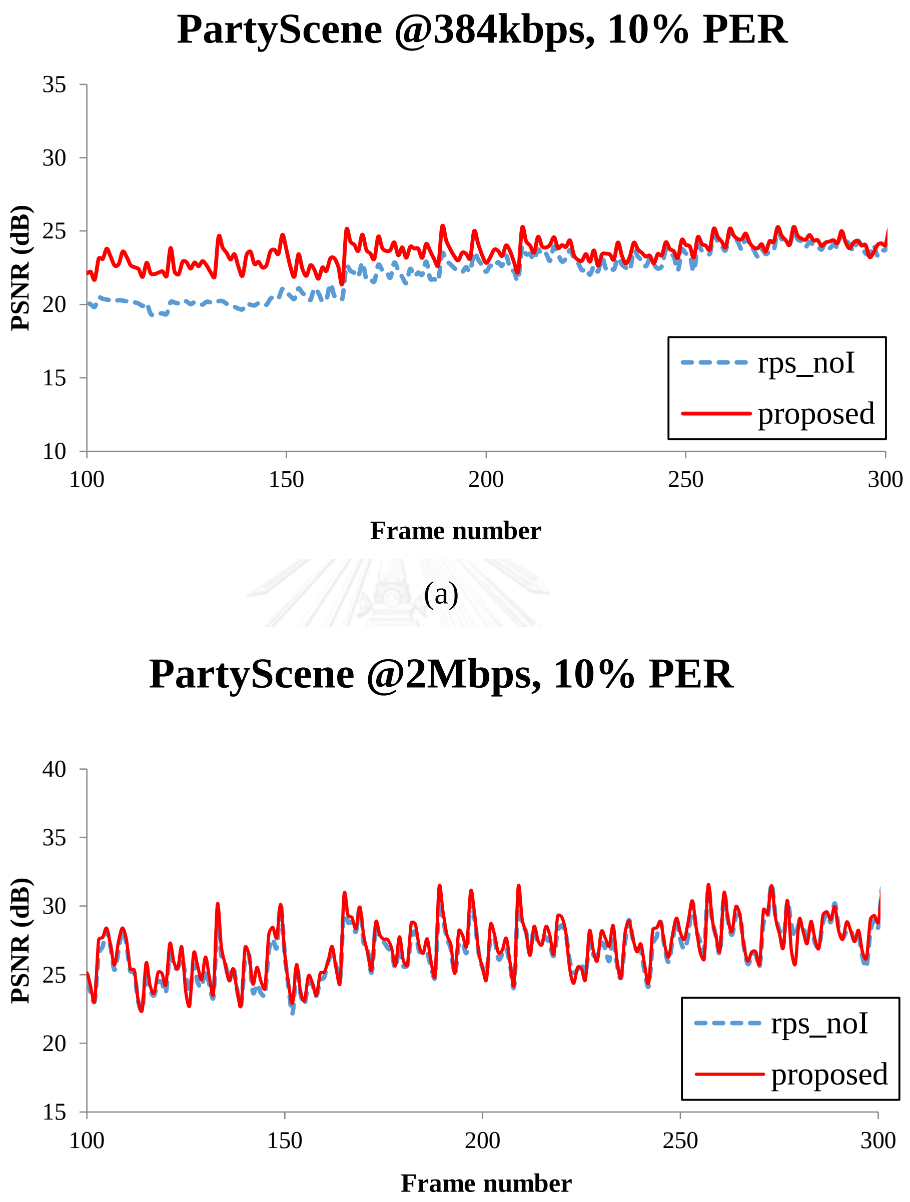


(a)



(b)

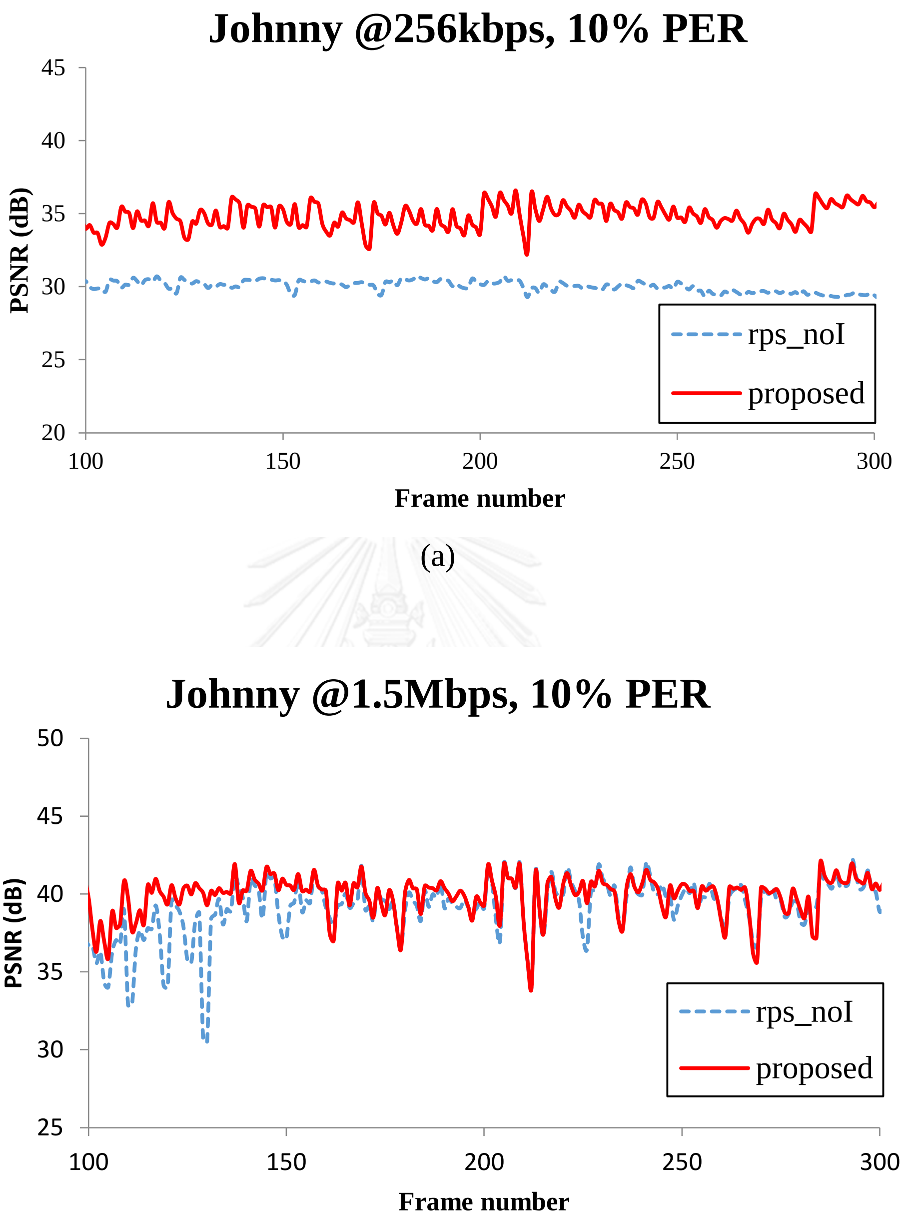
Figure 4.5: Frame by frame PSNR values of BQMall sequence for 10% PER at target bit rate (a) 384kbps and (b) 2Mbps, respectively.



(a)

(b)

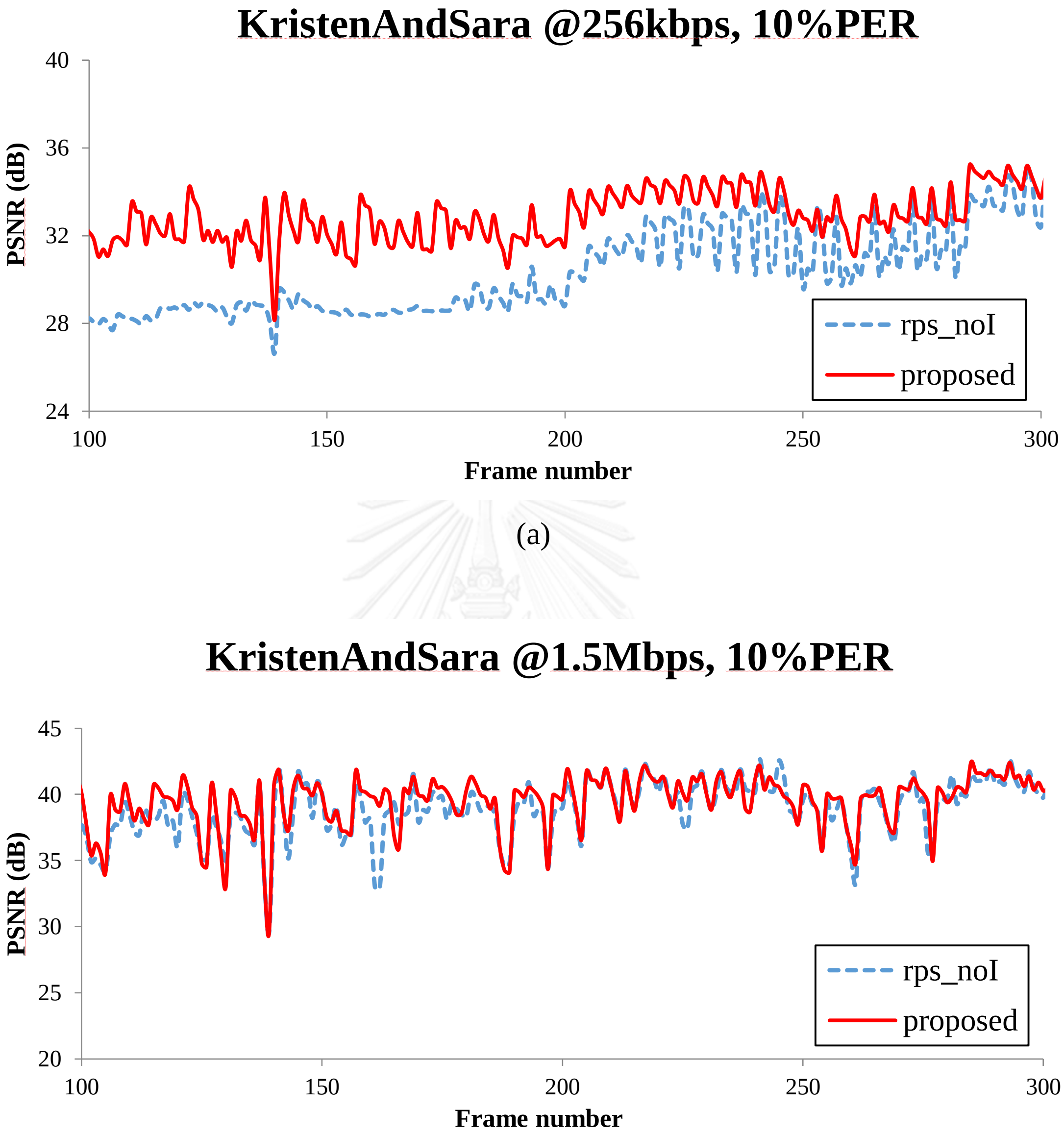
Figure 4.6: Frame by frame PSNR values of PartyScene sequence for 10% PER at target bit rate (a) 384kbps and (b) 2Mbps, respectively.



(a)

(b)

Figure 4.7: Frame by frame PSNR values of Johnny sequence for 10% PER at target bit rate (a) 256kbps and (b) 1.5Mbps, respectively.



(a)

(b)

Figure 4.8: Frame by frame PSNR values of KristenAndSara sequence for 10% PER at target bit rate (a) 256kbps and (b) 1.5Mbps, respectively.

The performance of proposed algorithm is also evaluated based on generated bits. Basically, some additional data need to add a bit stream to increase the error resiliency. For rate constrained transmission, the amount of overhead bits should be controlled. From Table. 4.3. to Table. 4.5, the generated bits for proposed algorithm, original HM-15 encoder (error free), and modified A-NEWPRED are compared. According to the results, the amount of bit rate increment is less than 0.5% of generated bit rate of original encoder which is only a negligible amount. Therefore, it is obvious that modified rate control algorithm works very well. For WVGA sequences, the generated bit rates of proposed algorithm for all test cases are almost the same as that of original HM-15 encoder. For 720p sequences, the generated bit rates of proposed algorithm for 3% PER and 5% PER cases are almost same as that of original encoder but for 10% PER case, the amount of overhead bits of proposed algorithm is about 0.4% of generated bit rate of original HM.15 encoder. This increment is mainly due to insertion of intra coded blocks. Since the packet error rate is increased, the number of intra coded blocks is also increased.

Based on these results, it can be concluded that the proposed algorithm is more suitable for low bit rate applications. The effectiveness of proposed algorithm also depends on the characteristics of input video. If the video contains a lot of motion, a reference picture with long temporal distance may not be useful. However, for conversational videos with still background (e.g., Johnny, FourPeople, and KristenAndSara), the proposed algorithm can perform significantly better than others. The results of some selected frames are also shown in Fig. 4.7 and 4.8. The improvement in visual quality of proposed algorithm can be seen from that figures.

Table 4.3: Results comparison for BasketballDrillText (10% PER)

Target Bit Rate (kbps)	Average Y-PSNR (dB)				Bit-Rate (kbps)			
	Error Free HM 15	RPS_noI	RPS_wI	Proposed	Error Free HM 15	RPS_noI	RPS_wI	Proposed
384	29.84	26.71	25.45	**27.16**	393.99	393.90	393.99	393.94
512	30.86	27.31	26.07	**27.86**	522.06	521.89	521.96	512.97
768	32.82	28.40	27.12	**28.70**	778.10	777.79	777.93	777.99
1200	34.87	29.47	28.27	**29.63**	1210.14	1209.77	1209.82	1209.99
2000	37.13	30.46	29.19	**30.57**	2010.14	2009.63	2009.84	2010.05

Table 4.4: Results comparison for BasketballDrillText (5% PER)

Target Bit Rate (kbps)	Average Y-PSNR (dB)				Bit-Rate (kbps)			
	Error Free HM 15	RPS_noI	RPS_wI	Proposed	Error Free HM 15	RPS_noI	RPS_wI	Proposed
5% PER								
384	29.84	28.94	27.85	**29.44**	393.99	394.02	395.77	394.00
512	30.86	29.87	28.81	**30.50**	522.06	522.04	522.03	522.01
768	32.82	31.63	30.58	**31.92**	778.10	778.10	778.05	778.09
1200	34.87	33.43	32.63	**33.45**	1210.14	1210.13	1210.06	1210.11
2000	37.13	**35.38**	34.65	35.35	2010.14	2010.15	2010.13	2010.13

Table 4.5: Results comparison for BasketballDrillText (3% PER)

Target Bit Rate (kbps)	Average Y-PSNR (dB)				Bit-Rate (kbps)			
	Error Free HM 15	RPS_noI	RPS_wI	Proposed	Error Free HM 15	RPS_noI	RPS_wI	Proposed
384	29.84	29.31	28.32	**29.79**	393.99	394.02	394.03	394.04
512	30.86	30.30	29.42	**30.92**	522.06	521.99	522.01	522.00
768	32.82	32.13	31.35	**32.42**	778.10	778.03	778.06	778.07
1200	34.87	34.03	33.53	**34.07**	1210.14	1210.13	1210.05	1210.13
2000	37.13	**36.12**	35.79	36.09	2010.14	2010.14	2010.11	2010.16

Table 4.6: Results comparison for BQMall (10% PER)

Target Bit Rate (kbps)	Average Y-PSNR (dB)				Bit-Rate (kbps)			
	Error Free HM 15	RPS_noI	RPS_wI	Proposed	Error Free HM 15	RPS_noI	RPS_wI	Proposed
384	28.04	24.14	24.30	**25.52**	396.05	396.05	396.16	396.65
512	29.70	25.03	25.16	**26.21**	524.04	524.05	524.16	524.68
768	31.58	25.97	25.96	**27.05**	780.05	780.05	780.05	781.01
1200	33.83	26.97	26.76	**27.91**	1212.08	1212.05	1212.06	1213.62
2000	36.41	28.12	27.72	**28.80**	2012.09	2012.08	2012.07	2014.79

Table 4.7: Results comparison for BQMall l (5% PER)

Target Bit Rate (kbps)	Average Y-PSNR (dB)				Bit-Rate (kbps)			
	Error Free HM 15	RPS_noI	RPS_wI	Proposed	Error Free HM 15	RPS_noI	RPS_wI	Proposed
384	28.04	27.31	27.09	**28.88**	396.05	396.04	396.01	396.03
512	29.70	28.78	28.64	**30.07**	524.04	524.04	524.02	524.03
768	31.58	30.36	30.15	**31.63**	780.05	780.05	780.05	780.05
1200	33.83	32.18	31.95	**33.27**	1212.08	1212.04	1212.05	1212.05
2000	36.41	34.38	33.97	**34.98**	2012.09	2012.06	2012.10	2012.05

Table 4.8: Results comparison for BQMall (3% PER)

Target Bit Rate (kbps)	Average Y-PSNR (dB)				Bit-Rate (kbps)			
	Error Free HM 15	RPS_noI	RPS_wI	Proposed	Error Free HM 15	RPS_noI	RPS_wI	Proposed
384	28.04	27.24	26.70	**28.97**	396.05	396.05	396.04	396.01
512	29.70	28.84	28.24	**30.23**	524.04	524.03	524.04	524.04
768	31.58	30.55	29.87	**31.84**	780.05	780.05	780.05	780.05
1200	33.83	32.44	31.53	**33.52**	1212.08	1212.05	1212.04	1212.05
2000	36.41	34.75	33.39	**35.28**	2012.09	2012.08	2012.08	2012.04

Table 4.9: Results comparison for PartyScene (10% PER)

Target Bit Rate (kbps)	Average Y-PSNR (dB)				Bit-Rate (kbps)			
	Error Free HM 15	RPS_noI	RPS_wI	Proposed	Error Free HM 15	RPS_noI	RPS_wI	Proposed
384	24.90	23.35	23.54	**24.46**	394.05	394.04	393.31	394.01
512	25.84	24.14	24.30	**25.17**	521.97	522.05	522.03	522.05
768	27.60	25.22	25.41	**26.17**	778.10	777.55	777.91	778.00
1200	29.47	26.51	26.64	**27.20**	1210.08	1209.43	1209.86	1210.07
2000	31.73	27.96	28.01	**28.40**	2010.14	2009.59	2009.69	2009.94

Table 4.10: Results comparison for PartyScene (5% PER)

Target Bit Rate (kbps)	Average Y-PSNR (dB)				Bit-Rate (kbps)			
	Error Free HM 15	RPS_noI	RPS_wI	Proposed	Error Free HM 15	RPS_noI	RPS_wI	Proposed
384	24.90	24.68	24.36	**25.74**	394.05	394.06	397.69	394.05
512	25.84	25.59	25.44	**26.62**	521.97	522.06	527.27	522.04
768	27.60	27.15	26.92	**27.90**	778.10	778.08	780.00	778.09
1200	29.47	28.87	28.59	**29.36**	1210.08	1210.09	1211.66	1210.04
2000	31.73	30.92	30.68	**31.10**	2010.14	2010.10	2010.06	2010.12

Table 4.11: Results comparison for (3% PER)

Target Bit Rate (kbps)	Average Y-PSNR (dB)				Bit-Rate (kbps)			
	Error Free HM 15	RPS_noI	RPS_wI	Proposed	Error Free HM 15	RPS_noI	RPS_wI	Proposed
384	24.90	24.78	24.44	**25.85**	394.05	394.07	397.51	394.05
512	25.84	25.72	25.54	**26.77**	521.97	522.08	527.48	522.07
768	27.60	27.37	27.14	**28.10**	778.10	778.07	779.94	778.13
1200	29.47	29.14	28.85	**29.62**	1210.08	1210.04	1211.66	1210.08
2000	31.73	31.27	31.07	**31.44**	2010.14	2010.15	2010.07	2010.13

Table 4.12: Results comparison for Johnny (10% PER)

Target Bit Rate (kbps)	Average Y-PSNR (dB)				Bit-Rate (kbps)			
	Error Free HM 15	RPS_noI	RPS_wI	Proposed	Error Free HM 15	RPS_noI	RPS_wI	Proposed
384	35.31	32.97	32.53	**35.89**	256.02	273.99	273.96	274.58
512	37.52	35.53	35.08	**37.41**	384.05	402.37	402.12	402.85
768	38.66	36.62	36.40	**38.26**	512.05	530.59	530.28	531.08
1200	40.07	38.13	37.86	**39.43**	850.04	869.15	868.68	869.33
2000	41.13	39.12	38.93	**40.39**	1500.12	1519.57	1519.33	1520.80

Table 4.13: Results comparison for Johnny (5% PER)

Target Bit Rate (kbps)	Average Y-PSNR (dB)				Bit-Rate (kbps)			
	Error Free HM 15	RPS_noI	RPS_wI	Proposed	Error Free HM 15	RPS_noI	RPS_wI	Proposed
384	35.31	34.27	33.16	**37.82**	256.02	273.70	273.83	273.74
512	37.52	37.04	36.13	**39.34**	384.05	401.73	401.73	401.70
768	38.66	38.22	37.58	**40.11**	512.05	529.77	529.70	529.80
1200	40.07	39.67	39.22	**41.07**	850.04	867.81	867.72	867.81
2000	41.13	40.40	40.15	**41.85**	1500.12	1217.81	1517.80	1517.90

Table 4.14: Results comparison for Johnny (3% PER)

Target Bit Rate (kbps)	Average Y-PSNR (dB)				Bit-Rate (kbps)			
	Error Free HM 15	RPS_noI	RPS_wI	Proposed	Error Free HM 15	RPS_noI	RPS_wI	Proposed
384	35.31	34.42	33.17	**38.03**	256.02	273.75	273.94	273.74
512	37.52	37.14	36.21	**39.54**	384.05	401.75	401.71	401.71
768	38.66	38.34	37.68	**40.32**	512.05	529.78	529.73	529.81
1200	40.07	39.83	39.38	**41.28**	850.04	867.81	867.73	867.81
2000	41.13	40.85	40.49	**42.10**	1500.12	1517.82	1517.81	1517.91

Table 4.15: Results comparison for FourPeople (10% PER)

Target Bit Rate (kbps)	Average Y-PSNR (dB)				Bit-Rate (kbps)			
	Error Free HM 15	RPS_noI	RPS_wI	Proposed	Error Free HM 15	RPS_noI	RPS_wI	Proposed
384	32.56	29.79	28.78	**31.44**	273.75	273.91	280.89	274.45
512	34.81	31.82	30.70	**33.37**	401.67	402.06	402.11	402.96
768	36.42	33.17	32.19	**34.5**	529.62	530.23	530.33	531.53
1200	38.91	35.48	34.65	**36.4**	867.64	868.48	868.78	871.04
2000	40.39	37.41	36.39	**38.29**	1517.87	1519.30	1519.54	1522.95

Table 4.16: Results comparison for FourPeople (5% PER)

Target Bit Rate (kbps)	Average Y-PSNR (dB)				Bit-Rate (kbps)			
	Error Free HM 15	RPS_noI	RPS_wI	Proposed	Error Free HM 15	RPS_noI	RPS_wI	Proposed
384	32.56	31.47	29.03	**33.30**	273.75	273.73	275.19	273.74
512	34.81	33.98	32.06	**35.31**	401.67	401.70	401.69	401.73
768	36.42	35.62	33.94	**36.58**	529.62	529.57	529.73	529.672
1200	38.91	37.94	36.78	**38.48**	867.64	867.43	867.60	876.61
2000	40.39	39.01	38.67	**40.26**	1517.87	1217.55	1517.58	1517.81

Table 4.17: Results comparison for FourPeople (3% PER)

Target Bit Rate (kbps)	Average Y-PSNR (dB)				Bit-Rate (kbps)			
	Error Free HM 15	RPS_noI	RPS_wI	Proposed	Error Free HM 15	RPS_noI	RPS_wI	Proposed
384	32.56	31.67	29.00	**33.58**	273.75	273.73	274.19	273.72
512	34.81	34.17	32.07	**35.62**	401.67	401.65	401.70	401.66
768	36.42	35.86	33.94	**36.91**	529.62	529.64	529.73	529.66
1200	38.91	38.29	36.95	**38.85**	867.64	867.52	867.71	867.61
2000	40.39	39.81	38.84	**40.68**	1517.87	1517.91	1517.63	1517.79

Table 4.18: Results comparison for KristenAndSara (10% PER)

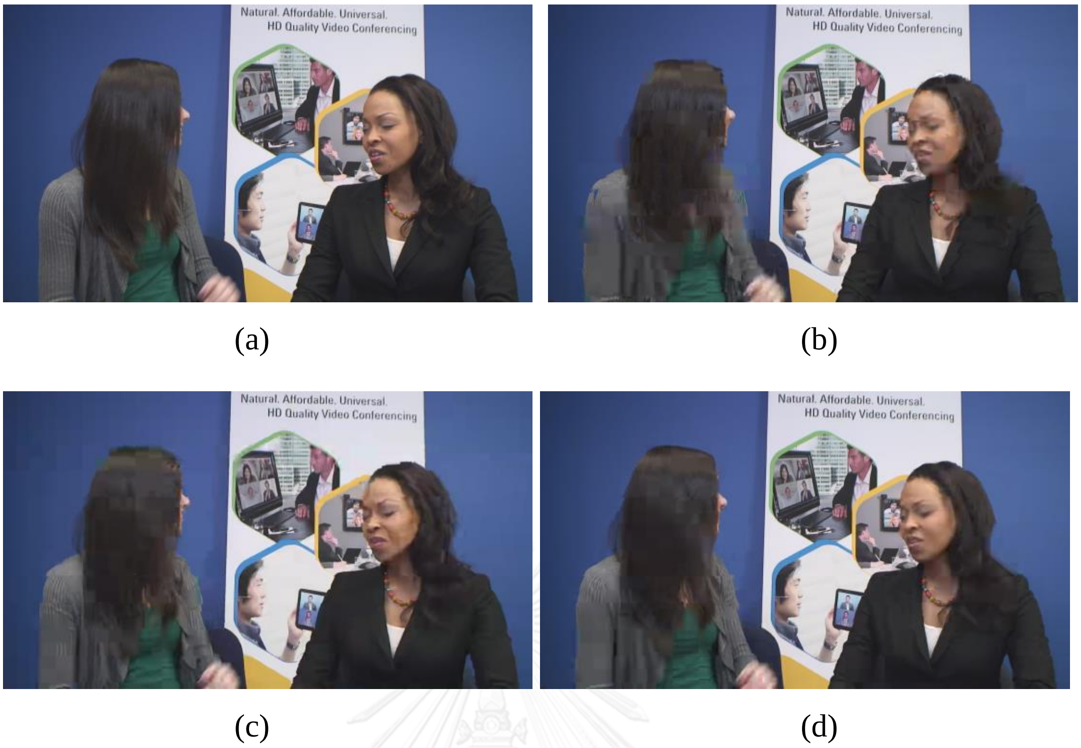
Target Bit Rate (kbps)	Average Y-PSNR (dB)				Bit-Rate (kbps)			
	Error Free HM 15	RPS_noI	RPS_wI	Proposed	Error Free HM 15	RPS_noI	RPS_wI	Proposed
384	35.39	32.07	29.93	**33.45**	273.91	274.13	274.04	275.29
512	37.75	34.14	31.90	**35.29**	401.85	402.34	402.23	403.37
768	39.02	35.25	32.00	**36.40**	529.88	530.48	530.24	531.44
1200	40.76	37.11	34.78	**37.95**	867.90	868.91	868.63	869.50
2000	42.09	38.74	37.24	**39.40**	1517.99	1519.85	1519.38	1520.25

Table 4.19: Results comparison for KristenAndSara (5% PER)

Target Bit Rate (kbps)	Average Y-PSNR (dB)				Bit-Rate (kbps)			
	Error Free HM 15	RPS_noI	RPS_wI	Proposed	Error Free HM 15	RPS_noI	RPS_wI	Proposed
384	35.39	34.28	31.40	**35.64**	273.91	273.86	273.83	273.86
512	37.75	36.80	34.24	**37.60**	401.85	401.87	401.84	401.86
768	39.02	38.04	35.71	**38.67**	529.88	529.87	529.85	529.87
1200	40.76	39.85	38.20	**40.21**	867.90	867.89	867.88	867.90
2000	42.09	40.90	39.65	**41.52**	1517.99	1217.94	1217.90	1517.98

Table 4.20: Results comparison for KristenAndSara (3% PER)

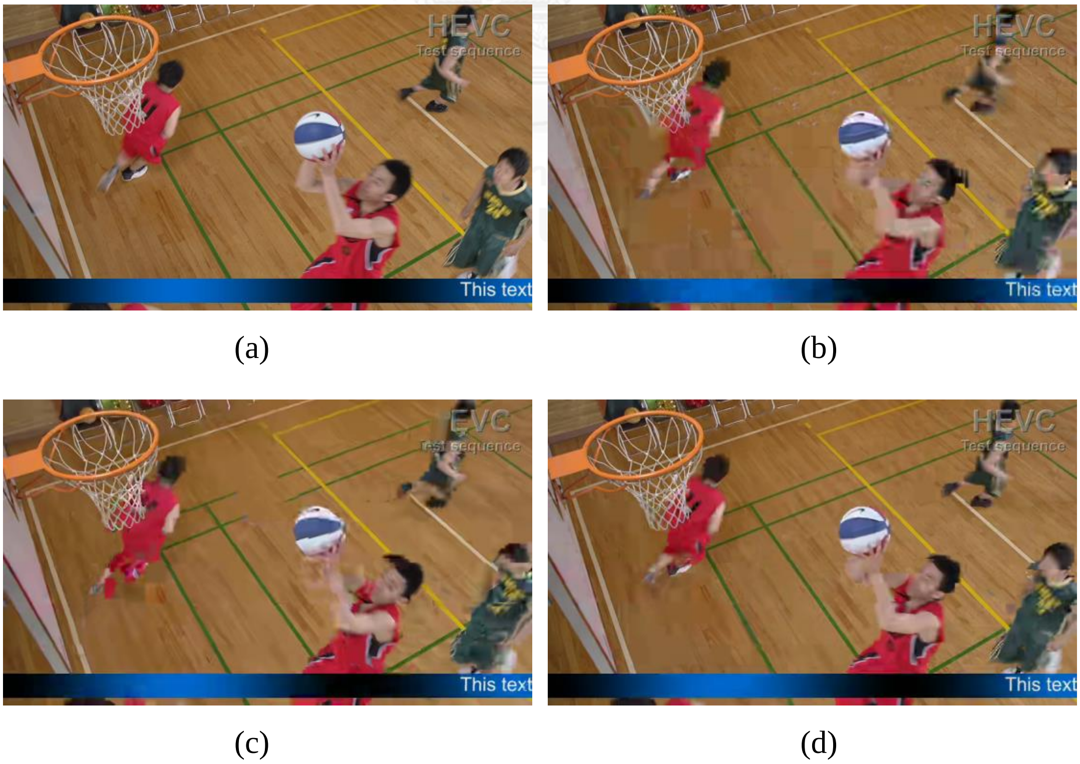
Target Bit Rate (kbps)	Average Y-PSNR (dB)				Bit-Rate (kbps)			
	Error Free HM 15	RPS_noI	RPS_wI	Proposed	Error Free HM 15	RPS_noI	RPS_wI	Proposed
384	35.39	34.50	31.53	**35.92**	273.91	273.90	273.89	273.90
512	37.75	37.16	34.46	**37.87**	401.85	401.84	401.84	401.86
768	39.02	38.34	36.01	**38.95**	529.88	529.87	529.85	529.87
1200	40.76	40.15	38.24	**40.51**	867.90	867.90	867.88	867.90
2000	42.09	41.54	40.25	**41.82**	1517.99	1517.97	1517.97	1517.92



(a) (b)

(c) (d)

Figure 4.9: Frame 137 of KristenAndSara sequence under 10% PER. (a) Original; (b) RPS_noI; (c) RPS_wI; (d) Proposed Method



(a) (b)

(c) (d)

Figure 4.10: Frame 57 of BasketballDrillText sequence under 10% PER. (a) Original; (b) RPS_noI; (c) RPS_wI; (d) Proposed Method

CHAPTER 5
CONCLUSION AND FUTURE WORKS

In this thesis, the new features of H.265/HEVC video coding standard are studied. Due to high compression, HEVC encoded bitstreams are more sensitive to packet errors. To get better error resilience for low delay conversational applications, an interactive error control method for HEVC is proposed.

Firstly, a conventional RPS algorithm is modified in order to adapt with the HEVC framework. This modified RPS is used as reference method in this thesis. When the feedback is large, the performance of RPS is reduced and error propagation occurs at some frames. To overcome this problem, long-term reference picture is used in previous approaches. However, if the reference distance is long, the coding efficiency is reduced.

To address this problem, a feedback-based error resilient algorithm for HEVC video transmission is proposed in this thesis. Each video frame is divided into MR region and non-MR region. Based on region information, maximum depth level of CTU is computed in order to reduce computational complexity of HEVC, while maintaining the quality in MR region. If an error is detected from the feedback, RPS algorithm is firstly applied. After that, for further improvement in quality, all CUs in MR region are forced to encode in Intra mode if the last intra refresh distance is greater than or equal to 4. This last intra refresh distance condition is added to make sure the number of intra coded blocks is within the acceptable rage.

When no error message is received from the decoder, the blocks in MR are forced to encode with intra mode if the last intra refresh frame is greater than or equal to 12. This force intra refresh can reduce error propagation effect for large feedback delay case.

In addition, we modified the R-lambda rate control model of HEVC so that it can adaptively allocate bits according to region information and coding mode. The visual quality of MR region is enhanced by using proposed method. The experimental results demonstrated that performence of proposed method outperforms the conventional RPS technique especially in low bit rate applications.

Possible future work of this research is to exploit the coding tree structure of HEVC in MR region representation such that more accurate region information is obtained. Currently, a frame is divided into only two regions: MR region and non-MR region. Using more regions per frame together with coding tree structure based representation will enable to apply packet prioritization method or unequal error protection method to the bitstream. This will further improve the error resiliency of the bitstream.

The proposed error resilient algorithm can be combined with an end-to-end distortion estimation model so that it can apply for applications where no feedback channel is available.

REFERENCES

[1] Cisco, "Global Mobile Data Traffic Forecast Update 2014–2019."

[2] G. J. Sullivan, J. Ohm, H. Woo-Jin, and T. Wiegand, "Overview of the High Efficiency Video Coding (HEVC) Standard," *Circuits and Systems for Video Technology, IEEE Transactions on,* vol. 22, pp. 1649-1668, 2012.

[3] T. Wiegand, G. J. Sullivan, G. Bjøntegaard, and A. Luthra, "Overview of the H. 264/AVC video coding standard," *Circuits and Systems for Video Technology, IEEE Transactions on,* vol. 13, pp. 560-576, 2003.

[4] B. Oztas, M. T. Pourazad, P. Nasiopoulos, and V. C. M. Leung, "A study on the HEVC performance over lossy networks," in *Electronics, Circuits and Systems (ICECS), 2012 19th IEEE International Conference on*, 2012, pp. 785-788.

[5] P. Pinol, A. Torres, O. Lopez, M. Martinez, and M. P. Malumbres, "Evaluating HEVC video delivery in VANET scenarios," in *Wireless Days (WD), 2013 IFIP*, 2013, pp. 1-6.

[6] M. A. Aabed and G. AlRegib, "No-reference quality assessment of HEVC videos in loss-prone networks," in *Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on*, 2014, pp. 2015-2019.

[7] J. Nightingale, W. Qi, C. Grecos, and S. Goma, "The impact of network impairment on quality of experience (QoE) in H.265/HEVC video streaming," *Consumer Electronics, IEEE Transactions on,* vol. 60, pp. 242-250, 2014.

[8] B. Girod and N. Farber, "Feedback-based error control for mobile video transmission," *Proceedings of the IEEE,* vol. 87, pp. 1707-1723, 1999.

[9] W. Yao and Z. Qin-Fan, "Error control and concealment for video communication: a review," *Proceedings of the IEEE,* vol. 86, pp. 974-997, 1998.

[10] T. Schierl, M. M. Hannuksela, Y.-K. Wang, and S. Wenger, "System layer integration of high efficiency video coding," *Circuits and Systems for Video Technology, IEEE Transactions on,* vol. 22, pp. 1871-1884, 2012.

[11] S. Fukunaga, T. Nakai, and H. Inoue, "Error resilient video coding by dynamic replacing of reference pictures," in *Global Telecommunications Conference, 1996. GLOBECOM'96.'Communications: The Key to Global Prosperity*, 1996, pp. 1503-1508.

[12] TelenorR&D, "An error resilience method based on back channel signalling and FEC," in *ITU-T/SG15/LBC-96-033*, ed. San Jose, January 1996.

[13] Y. Wang, M. Claypool, and R. Kinicki, "modeling RPS and evaluating video repair with VQM," *Multimedia, IEEE Transactions on,* vol. 11, pp. 128-137, 2009.

[14] W. Tu and E. Steinbach, "Proxy-based reference picture selection for error resilient conversational video in mobile networks," *Circuits and Systems for Video Technology, IEEE Transactions on,* vol. 19, pp. 151-164, 2009.

[15] L. Chenghao, Y.-K. Wang, M. M. Hannuksela, C. Ying, M. Sujeet, and M. Gabbouj, "RTP/AVPF compliant feedback for error resilient video coding in conversational applications," in *Communications and Information Technology, 2009. ISCIT 2009. 9th International Symposium on*, 2009, pp. 218-223.

[16] D. Hong, M. Horowitz, A. Eleftheriadis, and T. Wiegand, "H. 264 hierarchical P coding in the context of ultra-low delay, low complexity applications," in *Picture Coding Symposium (PCS), 2010*, 2010, pp. 146-149.

[17] R. Sjoberg, Y. Chen, A. Fujibayashi, M. M. Hannuksela, J. Samuelsson, T. K. Tan, *et al.*, "Overview of HEVC high-level syntax and reference picture management," *Circuits and Systems for Video Technology, IEEE Transactions on,* vol. 22, pp. 1858-1870, 2012.

[18] S. Kumar, L. Xu, M. K. Mandal, and S. Panchanathan, "Error resiliency schemes in H. 264/AVC standard," *Journal of Visual Communication and Image Representation,* vol. 17, pp. 425-450, 2006.

[19] Y. Wang, S. Wenger, J. Wen, and A. K. Katsaggelos, "Error resilient video coding techniques," *Signal Processing Magazine, IEEE,* vol. 17, pp. 61-82, 2000.

[20] G. Côté and F. Kossentini, "Optimal intra coding of blocks for robust video communication over the Internet," *Signal Processing: Image Communication,* vol. 15, pp. 25-34, 1999.

[21] P. Haskell and D. Messerschmitt, "Resynchronization of motion compensated video affected by ATM cell loss," in *Acoustics, Speech, and Signal Processing, 1992. ICASSP-92., 1992 IEEE International Conference on*, 1992, pp. 545-548.

[22] C. Quqing, C. Zhibo, G. Xiaodong, and C. Wang, "Attention-based adaptive intra refresh for error-prone video transmission," *Communications Magazine, IEEE,* vol. 45, pp. 52-60, 2007.

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

[24] B. Girod, U. Horn, and B. Belzer, "Scalable video coding with multiscale motion compensation and unequal error protection," in *Multimedia Communications and Video Coding*, ed: Springer, 1996, pp. 475-482.

[25] J. G. Apostolopoulos, "Reliable video communication over lossy packet networks using multiple state encoding and path diversity," in *Photonics West 2001-Electronic Imaging*, 2000, pp. 392-409.

[26] K.-P. Lim, G. Sullivan, and T. Wiegand, "Text description of joint model reference encoding methods and decoding concealment methods, jvt-o079," 2005.

[27] S. Ma, W. Gao, and Y. Lu, "Rate-distortion analysis for H. 264/AVC video coding and its application to rate control," *Circuits and Systems for Video Technology, IEEE Transactions on,* vol. 15, pp. 1533-1544, 2005.

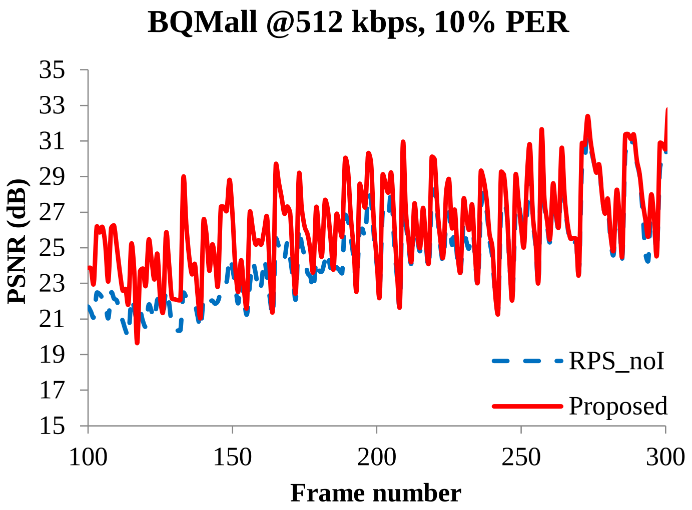
[28] Z. He, Y. K. Kim, and S. K. Mitra, "Low-delay rate control for DCT video coding via ρ-domain source modeling," *Circuits and Systems for Video Technology, IEEE Transactions on,* vol. 11, pp. 928-940, 2001.

[29] B. Li, H. Li, L. Li, and J. Zhang, "Domain Rate Control Algorithm for High Efficiency Video Coding," *Image Processing, IEEE Transactions on,* vol. 23, pp. 3841-3854, 2014.

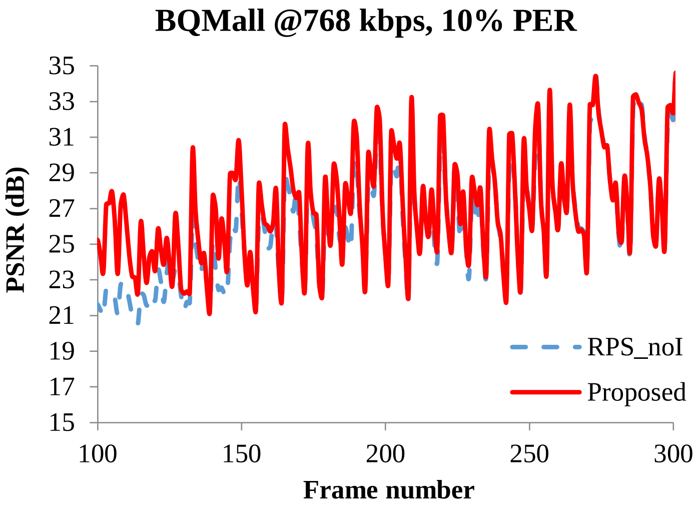
[30] ITU-T, "Nal unit loss software," in *SG16 WP3/ISO/IEC JTC1/SC29/WG11*, ed, 2012.

[31] G. Ren, P. Li, and G. Wang, "A novel hybrid coarse-to-fine digital image stabilization algorithm," *Information Technology Journal*, vol. 9, pp. 1390-1396, 2010.
[32] JCT-VC. HM 15.0 [Online]. Available: https://hevc.hhi.fraunhofer.de
[33] I.-T. a. ISO/IEC, "Joint call for proposals on video compression technology," ed, 2010.

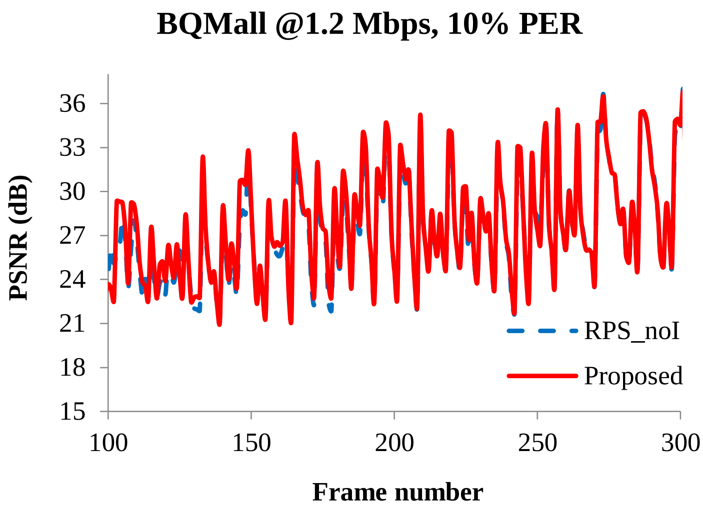
APPENDIX


BQMall @512 kbps, 10% PER
PSNR (dB)
35
33
31
29
27
25
23
21
19
17
15
100
150
200
250
300
Frame number
RPS_noI
Proposed

(a)

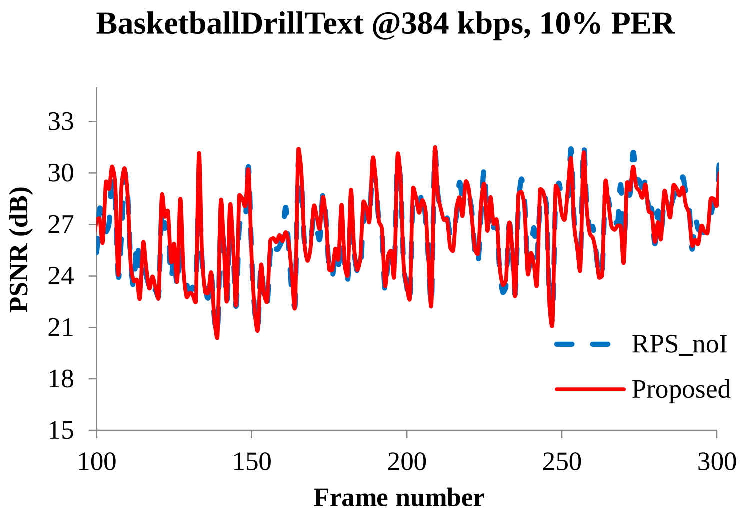

BQMall @768 kbps, 10% PER
PSNR (dB)
35
33
31
29
27
25
23
21
19
17
15
100
150
200
250
300
Frame number
RPS_noI
Proposed

(b)

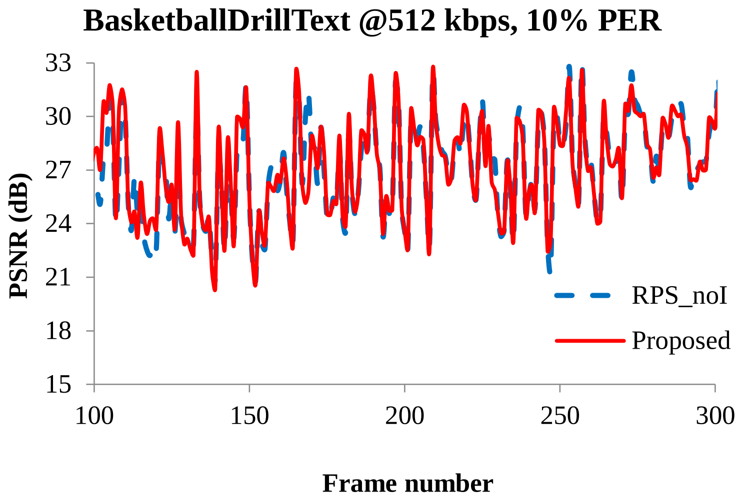


(c)

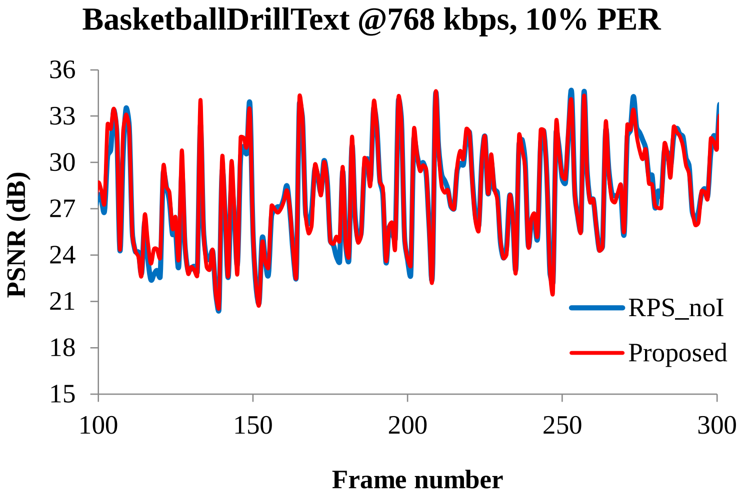
Figure A. 1: Frame by frame PSNR values of BQMall sequence for 10% PER at target bit rate (a) 512kbps, (b) 768kbps, and (c) 1.2Mbps, respectively.



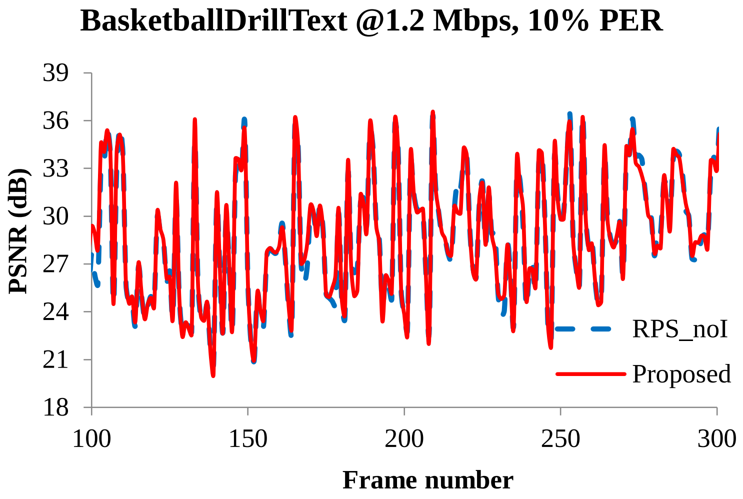
(a)


BasketballDrillText @512 kbps, 10% PER
PSNR (dB)
33
30
27
24
21
18
15
100
150
200
250
300
Frame number
RPS_noI
Proposed

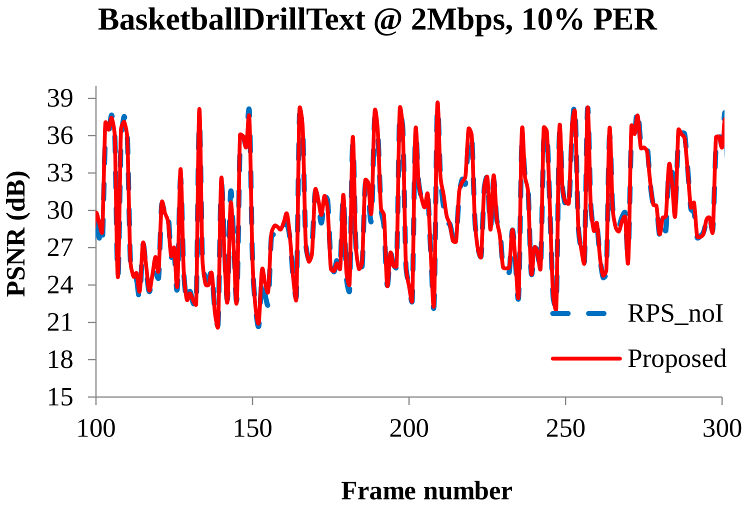
(b)


BasketballDrillText @768 kbps, 10% PER
PSNR (dB)
36
33
30
27
24
21
18
15
100
150
200
250
300
Frame number
RPS_noI
Proposed

(c)

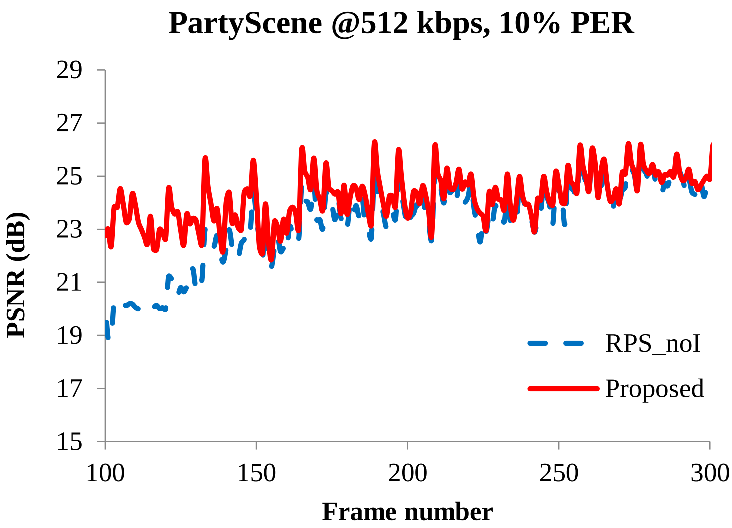


(d)

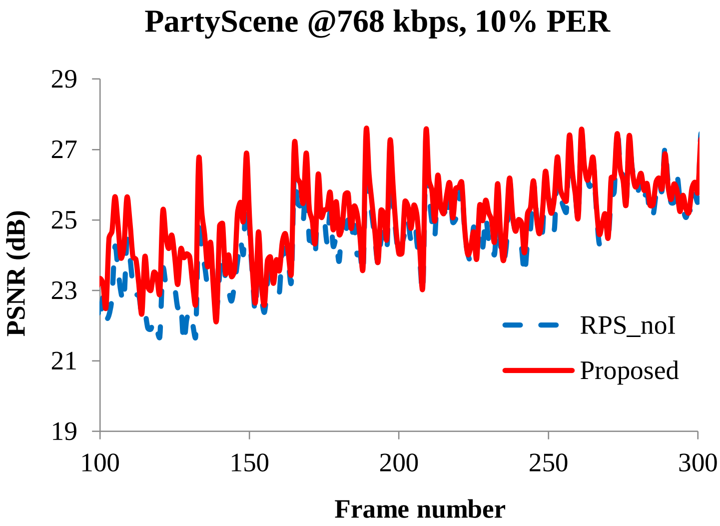


(e)

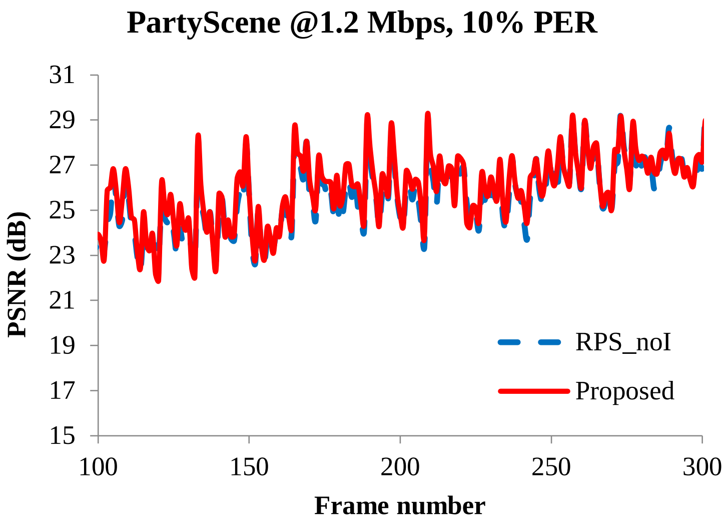
Figure A. 2: Frame by frame PSNR values of BasketballDrillText sequence for 10% PER at target bit rate (a) 384kbps, (b) 512kbps, (c) 768kbps, (d) 1.2Mbps, and (e) 2Mbps, respectively.


PartyScene @512 kbps, 10% PER
PSNR (dB)
29
27
25
23
21
19
17
15
100
150
200
250
300
Frame number
RPS_noI
Proposed

(a)

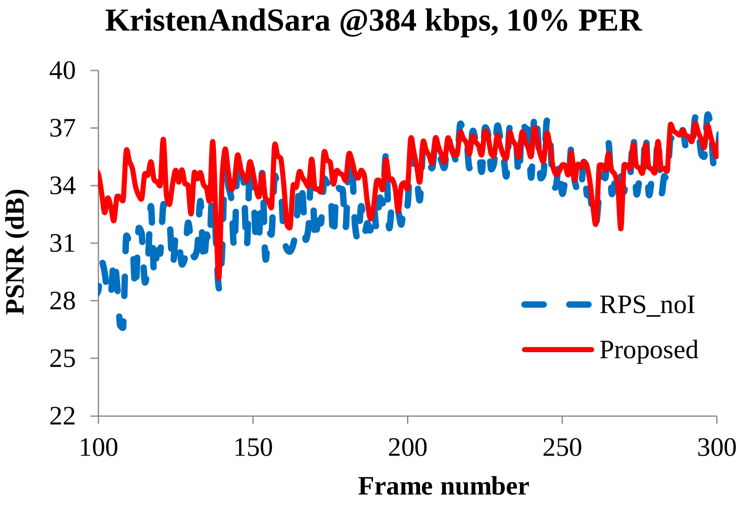

PartyScene @768 kbps, 10% PER
PSNR (dB)
29
27
25
23
21
19
100
150
200
250
300
Frame number
RPS_noI
Proposed

(b)

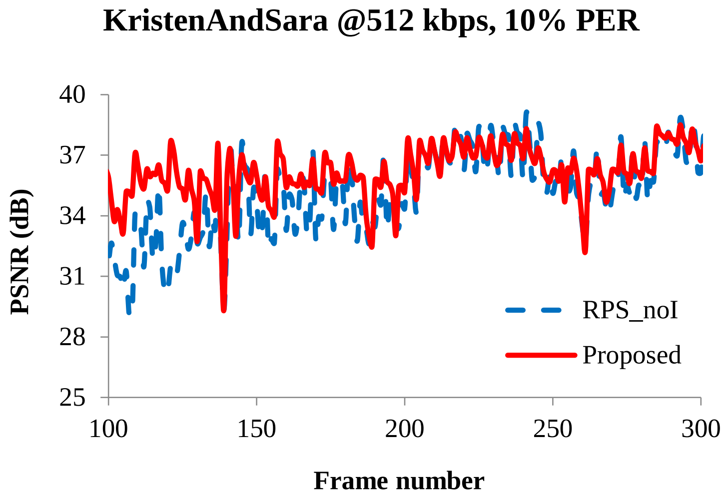

PartyScene @1.2 Mbps, 10% PER

(c)

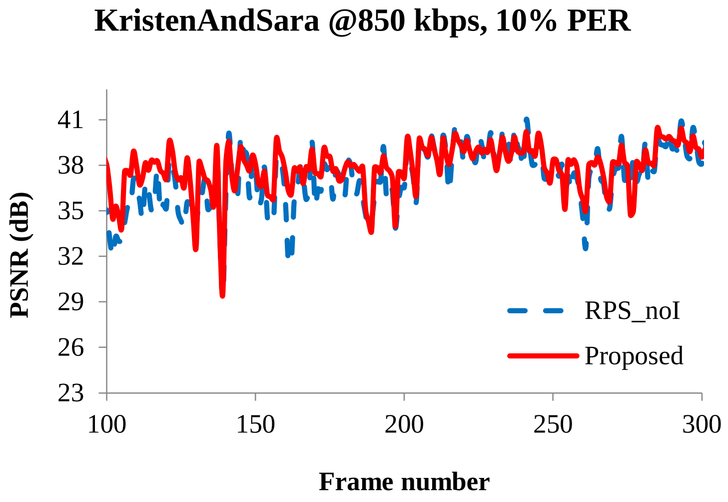
Figure A. 3: Frame by frame PSNR values of PartyScene sequence for 10% PER at target bit rate (a) 512kbps, (b) 768kbps, and (b)1.2Mbps, respectively.


KristenAndSara @384 kbps, 10% PER

(a)

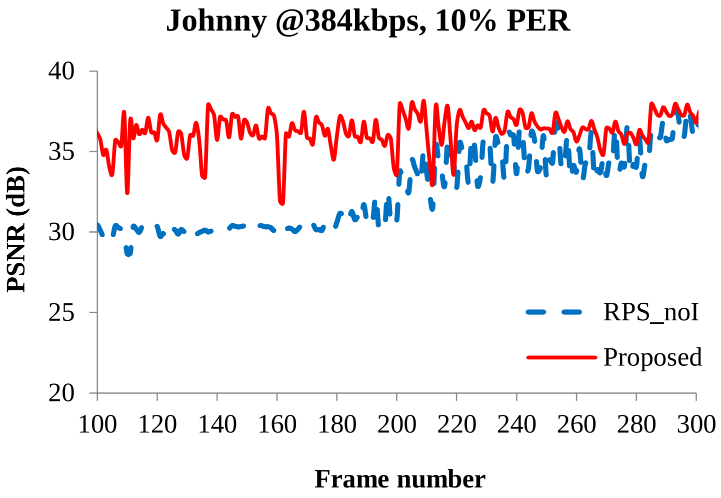

KristenAndSara @512 kbps, 10% PER

(b)

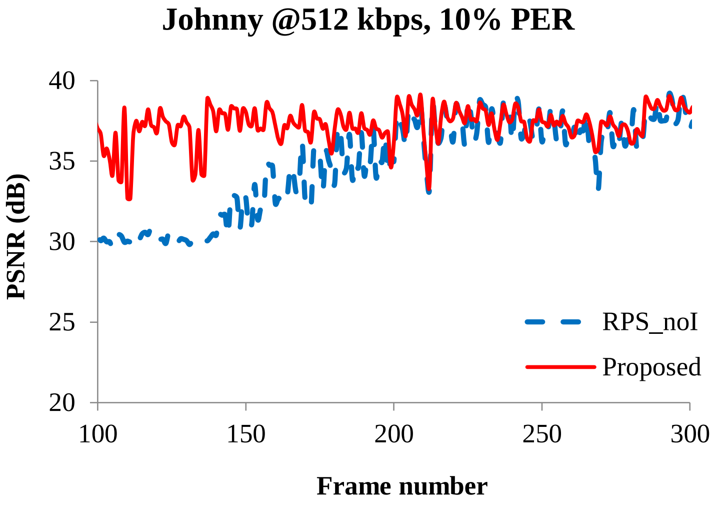

KristenAndSara @850 kbps, 10% PER

(c)

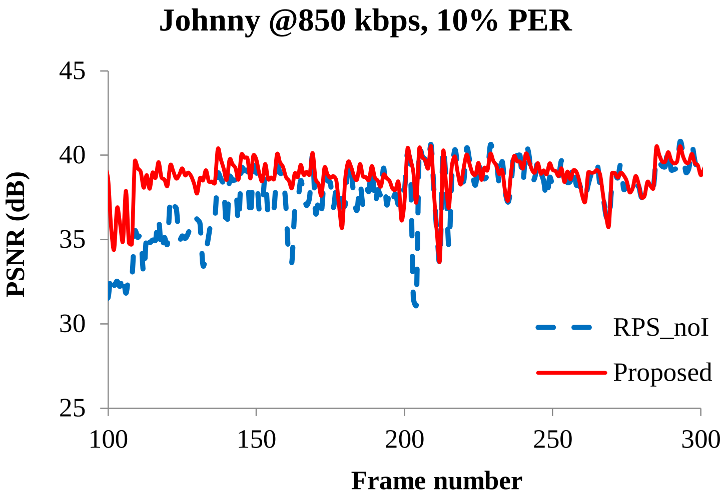
Figure A. 4: Frame by frame PSNR values of KristenAndSara sequence for 10% PER at target bit rate (a) 384kbps, (b) 512kbps, and (c) 850kbps, respectively.


Johnny @384kbps, 10% PER
PSNR (dB)
40
35
30
25
20
100
120
140
160
180
200
220
240
260
280
300
Frame number
RPS_noI
Proposed

(a)

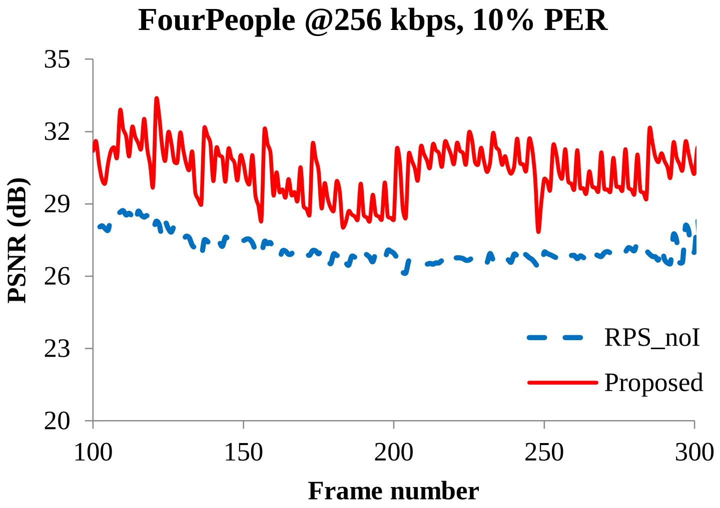

Johnny @512 kbps, 10% PER
PSNR (dB)
40
35
30
25
20
100
150
200
250
300
Frame number
RPS_noI
Proposed

(b)

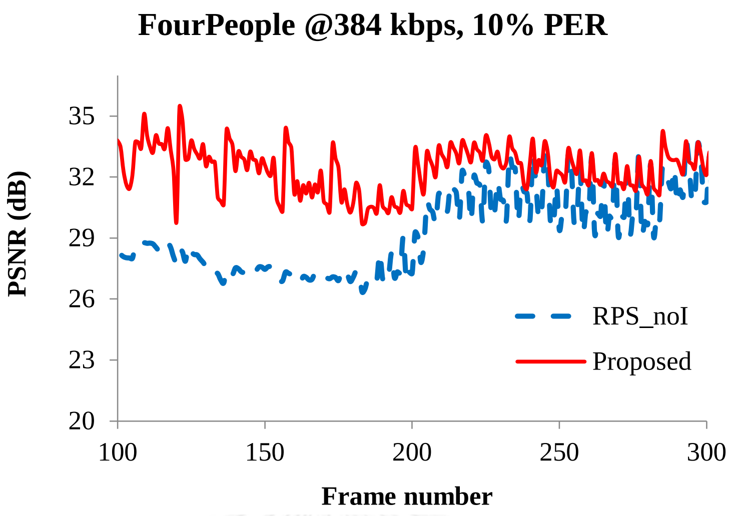


(c)

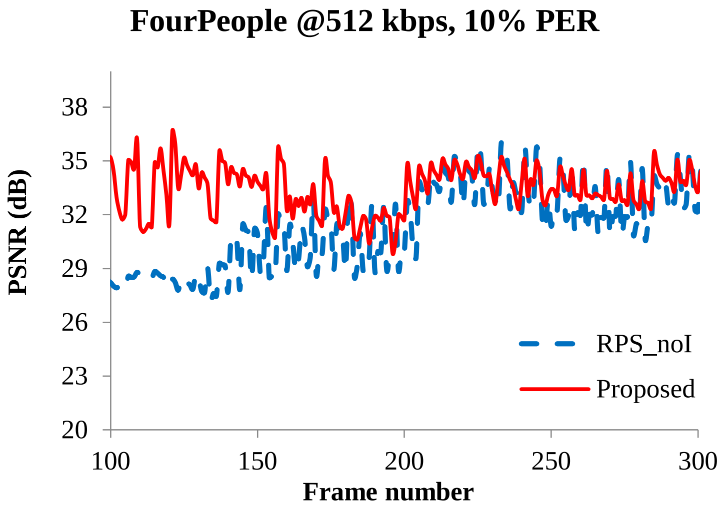
Figure A. 5: Frame by frame PSNR values of Johnny sequence for 10% PER at target bit rate (a) 384kbps, (b) 512kbps, and (c) 850kbps, respectively.



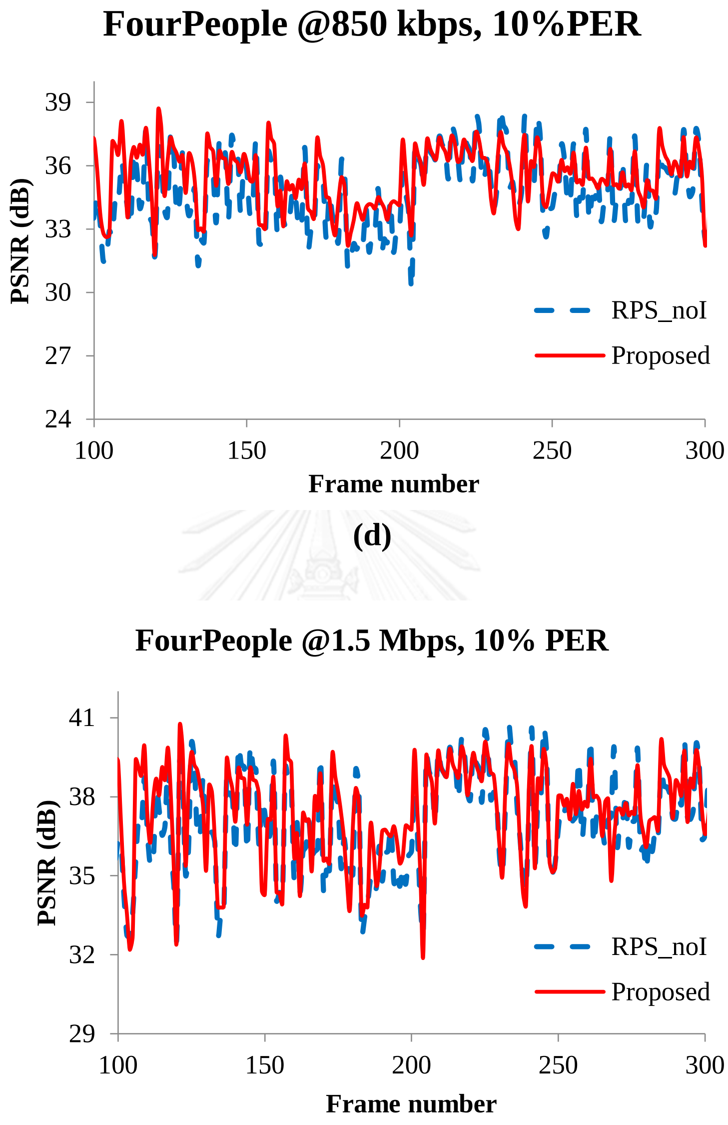
(a)


FourPeople @384 kbps, 10% PER
PSNR (dB)
35
32
29
26
23
20
100
150
200
250
300
Frame number
RPS_noI
Proposed

(b)


FourPeople @512 kbps, 10% PER
PSNR (dB)
38
35
32
29
26
23
20
100
150
200
250
300
Frame number
RPS_noI
Proposed

(c)


FourPeople @850 kbps, 10%PER

(d)

FourPeople @1.5 Mbps, 10% PER

(e)

Figure A. 6: Frame by frame PSNR values of FourPeople sequence for 10% PER at target bit rate (a) 256kbps, (b) 384kbps, (c) 512kbps, (d) 850kbps, and (e) 1.5Mbps, respectively.

VITA

Htoo Maung Maung was born in Yangon, Myanmar, in 1983. He received his B.E., and M.E., degrees from Mandalay Technological University, Mandalay, Myanmar, in 2005 and 2007, respectively. He has been granted a scholarship by AUN/SEED-Net, JICA to pursue his Doctoral degree in electrical engineering at the Digital Signal Processing Research Laboratory, Department of Electrical Engineering, Faculty of Engineering, Chulalongkorn University, Thailand, since 2012. His research interest includes Video coding and wireless video transmission.