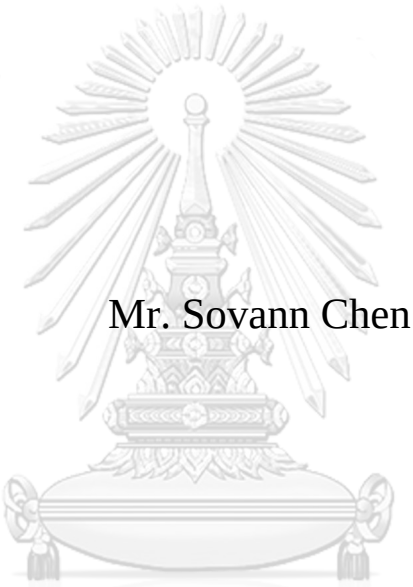
Learning-Based Approach for Visual Quality Enhancement on High Efficiency Video Coding Standard

Mr. Sovann Chen



A Dissertation Submitted in Partial Fulfillment of the Requirements
for the Degree of Doctor of Philosophy in Electrical Engineering
Department of Electrical Engineering
FACULTY OF ENGINEERING
Chulalongkorn University
Academic Year 2020

วิธีการภายใต้การเรียนรู้เพื่อเพิ่มคุณภาพสำหรับภาพตามมาตรฐานการเข้ารหัสวิดีโอประสิทธิภาพสูง

นายโซวาน เชน

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรดุษฎีบัณฑิต
สาขาวิชาวิศวกรรมไฟฟ้า ภาควิชาวิศวกรรมไฟฟ้า
คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย
ปีการศึกษา 2563

Thesis Title	Learning-Based Approach for Visual Quality Enhancement on High Efficiency Video Coding Standard
By	Mr. Sovann Chen
Field of Study	Electrical Engineering
Thesis Advisor	Associate Professor Dr. SUPAVADEE ARAMVITH
Thesis Co Advisor	Miyanaga Yoshikazu

Accepted by the FACULTY OF ENGINEERING, Chulalongkorn University in Partial Fulfillment of the Requirement for the Doctor of Philosophy

..... Dean of the FACULTY OF ENGINEERING
(Associate Professor Dr. SUPOT TEACHAVORASINSKUN)

DISSERTATION COMMITTEE

..... Chairman
(Associate Professor Dr. NISACHON TANGSANGIUMVISAI)

..... Thesis Advisor
(Associate Professor Dr. SUPAVADEE ARAMVITH)

..... Thesis Co-Advisor
(Miyanaga Yoshikazu)

..... Examiner
(Assistant Professor Dr. SUREE PUMRIN)

..... Examiner
(Assistant Professor Dr. WIDHYAKORN ASDORNWISED)

..... External Examiner
(Dr. Sitapa Rujikietgumjorn)

โซวาน เชน : วิธีการภายใต้การเรียนรู้เพื่อเพิ่มคุณภาพสำหรับภาพตามมาตรฐานการเข้ารหัสวิดีโอประสิทธิภาพสูง. (Learning-Based Approach for Visual Quality Enhancement on High Efficiency Video Coding Standard) อ.ที่ปรึกษาหลัก : รศ. ดร.สุภาวดี อร่ามวิทย์, อ.ที่ปรึกษาร่วม : มิยานากะ โยชิคาซู

การเข้ารหัสวิดีทัศน์แบบประสิทธิภาพสูง (เฮชอีวีซี) มีการปรับปรุงประสิทธิภาพการเข้ารหัสอย่างมาก เมื่อเทียบกับมาตรฐานการเข้ารหัสวิดีทัศน์ก่อนหน้า ดังเช่น เอชจุดสองหกสี่/เอวีซี อย่างไรก็ตาม การควบคุมอัตราที่มีอยู่จะปรับปรุงพารามิเตอร์ตามการกำหนดค่าเริ่มต้นแบบคงที่ ซึ่งอาจทำให้เกิดการทำนายค่าของการจัดสรรบิตผิดพลาดให้แต่ละ หน่วยเข้ารหัสแบบต้นไม้ (ซีทียู) ในเฟรม ในบทความนี้ เสนอการจัดผังภายใต้การเรียนรู้ ระหว่างพารามิเตอร์ของการควบคุมอัตราการเข้ารหัสและเนื้อหาวิดีทัศน์ เพื่อให้ได้อัตราบิตเป้าหมายที่แม่นยำและคุณภาพวิดีทัศน์ดี กรอบงานที่เสนอประกอบด้วยการเข้ารหัสโครงสร้างหลักสองอย่าง ได้แก่ การเข้ารหัสเชิงพื้นที่และการเข้ารหัสเชิงเวลา โดยเรากำหนดค่าการเพิ่มประสิทธิภาพภายใต้การเรียนรู้ของกลุ่มอนุภาคที่มีประสิทธิภาพ สำหรับการเข้ารหัสเชิงพื้นที่และเวลา เพื่อกำหนดพารามิเตอร์ที่เหมาะสมที่สุดในระดับซีทียู สำหรับการเข้ารหัสเชิงพื้นที่ที่ระดับภาพ เรานำเสนอการใช้ข้อมูลแบบตกค้างในเชิงความหมาย ในกระบวนการปรับปรุงพารามิเตอร์เพื่อควบคุมบิตภาพจริงได้อย่างถูกต้อง ผลการทดลองแสดงให้เห็นว่า ขั้นตอนวิธีที่เสนอนั้น มีประสิทธิภาพดีในการเข้ารหัสแบบเฮชอีวีซี และมีประสิทธิภาพดีกว่าการควบคุมอัตราในซอฟต์แวร์อ้างอิงการเข้ารหัสเฮชอีวีซีในปัจจุบันแบบ เฮชเอ็มขีดสิบหกจุดสิบ โดยค่าเฉลี่ย 0.19 เดซิเบลและสูงสุดถึง 0.41 เดซิเบลสำหรับโครงสร้างการเข้ารหัสแบบพีที่มีความล่าช้าต่ำ

สาขาวิชา	**วิศวกรรมไฟฟ้า**	**ลายมือชื่อนิสิต** .....
ปีการศึกษา	2563	**ลายมือชื่อ อ.ที่ปรึกษาหลัก** .....
		ลายมือชื่อ อ.ที่ปรึกษาร่วม .....

6071452321 : MAJOR ELECTRICAL ENGINEERING
KEYWOR
D:

Sovann Chen : Learning-Based Approach for Visual Quality Enhancement on High Efficiency Video Coding Standard. Advisor: Assoc. Prof. Dr. SUPAVADEE ARAMVITH Co-advisor: Miyanaga Yoshikazu

High Efficiency Video Coding (HEVC) has dramatically enhanced the coding efficiency compared to the previous video coding standard, H.264/AVC. However, the existing rate control updates its parameters according to a fixed initialization, which can cause error prediction of bit allocation to each coding tree unit (CTU) in frames. In this work, a learning-based mapping between rate control parameters and video contents is proposed to achieve an accurate target bit rate and good video quality. The proposed framework contains two main structural codings, including spatial and temporal coding. We initiate an effective learning-based particle swarm optimization for both spatial and temporal coding to determine the optimal parameters at the CTU level. For temporal coding at the picture level, we introduce semantic residual information into the parameter updating process to regulate the bit correctly on the actual picture. Experimental results indicate that the proposed algorithm is effective for HEVC and outperforms the state-of-the-art rate control in the HEVC reference software (HM-16.10) by 0.19 dB on average and up to 0.41 dB for low delay P coding structure.

Field of Study:	Electrical Engineering	Student's Signature .....
Academic Year:	2020	Advisor's Signature .....
		Co-advisor's Signature .....

ACKNOWLEDGEMENTS

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

Sovann Chen

TABLE OF CONTENTS

Page

LIST OF FIGURE

LIST OF TABLE

CHAPTER 1
INTRODUCTION

1.1.Motivation and Problem Statement

Every five years, the global internet traffic has been reported by Cisco [1]. Table 1.1 shows Gigabytes (GB) traffic data from 1992 to the estimated data in 2022. In 2022, the traffic data will reach about 46600 GB per second which is a massive amount of data compared to the traffic data in the year 2007. Furthermore, it will reach a CAGR of 26 percent in 2022. CAGR is the compound annual growth rate, and it is computed by taking the total number of traffic flow data at the end year to divide the total number of traffic flow data at start year, then power to the inverse of the length of year difference. In addition, the most attention data which takes much more consumption of traffic flow is the video data including Ultra High Definition (UHD) video, High Definition (HD) video, Standard Definition (SD) video, *etc*. The report is indicated that the video traffic flow is increased exponentially and reached 325 Exabytes per month, as shown in Figure 1.1.

UHD video data will be increased from 3 percent to 22 percent of total video data traffic flow, and HD video data will be increased from 46 percent to 57 percent within five years. Additionally, the video application traffic is also reported, including the video surveillance and real-time video transmission over multimedia, as shown in Figure 1.2. As a result, the live video traffic can consume from 5 percent up to 17 percent throughout the period from 2017 to 2022. Suppose an uncompressed or raw color video contains 100 frames with the resolution 1920 x 1080, and there is 8 bit for a pixel. Then, the total bytes assigns to that uncompressed video equal to 1920 x 1080 x 3 x 100 = 622.08 MB.

Table 1.1 Historical Internet Context

Year	Global internet traffic
1992	100 GB per day
1997	100 GB per day
2002	100 GB per day
2007	2000 GB per second
2017	46600 GB per second
2022	150700 GB per second

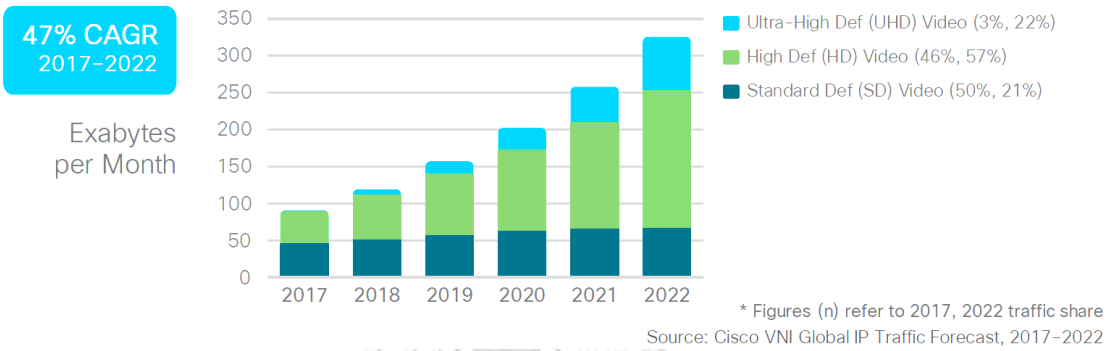


Figure 1.1. Global Video Traffic [1]

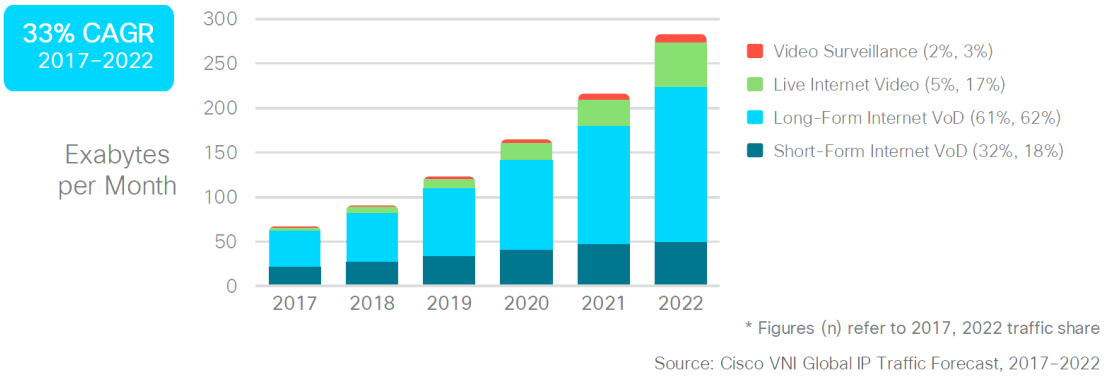


Figure 1.2 Global Video Application Traffic [1]

Due to this massive bit consumption of raw video data storage, which is going enormous increase in 2022 with limited bandwidth, the raw video is needed to reduce huge bits before storing or transmitting it to the receiver. The technique to reduce the bits of the video storage has been called the video compression technique. There is a compression ratio to set whether the target bit is a lossless or lossy reduction. The higher compression ratio is the minor bit consumption, but it can lead

to high video distortion. The video compression technique has been applied and upgraded to accomplish a high compression ratio since 1990 up till now [2], [3], [4], [5], [6], [7], and [8]. The first video compression technique, named H.261 [2], was published by the International Telecommunication Union (ITU). H.261 was built for the transmission over integrated services digital network (ISDN) lines on which the data rates are multiples of 64 Kbit per second. The video compression technique, named High-Efficiency Video Coding (HEVC) [8], has been published in 2013 to accomplish the requirements above. There are several advanced techniques [9] which embed in HEVC to lead the performance of compression better than the previous one, H.264/AVC [7] about 50 percent bit deduction at the same object quality [10] and [11].

Moreover, those video surveillance and real-time video transmission are generally transmitted via a constant bit rate (CBR) channel. The encoder controller known as rate control is worked as the main rule to obtain the best rate-distortion (R-D) performance. In the HEVC reference software, the relationship between the target bit and Lagrange multiplier λ is modeled. The Lagrangian method is used to achieve the optimal trade-off between rate and distortion on the encoder side. However, two main difficulties are observed in [8], such as the inaccurate bit allocation for a coding tree unit (CTU) by the first several CTUs. This inaccurate bit allocation also leads to the inaccurate λ estimation. Figure 1.3 indicates the heat map of bit allocation for CTUs with a red hover box—the higher intensity of red means the higher bit allocation consumption. As a result, the most high-frequency components like the small edge locations are assigned with very high bit allocation, which is unnecessary for visual perception. So the wrong bit allocation to CTU location can degrade the final rate control result.

Figure 1.3 The Example of Bit Allocation in a Frame

Thus, this work proposes a novel learning-based framework to enhance the visual quality in HEVC.

1.2.Objectives

In this work, there are several objectives as follows:

- ❖ Investigate learning-based approach in video coding
- ❖ Enhance HEVC encoder to improve the visual quality of compressed video
- ❖ Evaluate the performance of visual quality in the proposed algorithm with the HEVC reference encoder software

1.3.Scope of Work

- ❖ Analyze the relationship of rate control with the neural network
- ❖ Propose the learning-based approach instead of the traditional rate control updating parameters to enhance the visual quality
- ❖ Examine the performance of visual quality in the proposed algorithm with HEVC reference software based on PSNR

1.4.Research Procedures

- ❖ Doing a literature review about neural network and video coding methods
- ❖ Collecting datasets of surveillance videos
- ❖ Doing simulation to check the performance of default HEVC reference software
- ❖ Proposing and implementing an algorithm to enhance the visual quality of the compressed video
- ❖ Taking proposal examination
- ❖ Writing a journal
- ❖ Writing a thesis paper
- ❖ Taking final thesis defense

CHAPTER 2
BACKGROUND AND LITERATURE REVIEW

This chapter presents four main topics to briefly describe the essential background of video coding or compression and the related work on fast coding in HEVC. The first introduction is about the fundamental video coding standard, and then the high-efficiency video coding (HEVC) is presented. The third part explains the version of rate control algorithms with its standard. The literature review is the last part of this chapter to indicate the current work on fast coding.

2.1. Video Coding Standard Overview

A multimedia application utilizes multiple media sources like sound/audio, text, graphics, images, and video in an application. There are numerous multimedia applications used in our daily lives, including Digital Video Disc (DVD), television (TV), video telephony, video games, teleconferencing, mobile phone, and computer. Digital video is one of the multimedia sources hugely utilized in multimedia applications, which leads to advanced video compression algorithms. Generally, digital videos have formed in various video coding formats known as compressed video to be utilized in multimedia applications like MPEG [3], h.261 [2], h.263 [5], h.264 [7], *etc*. There are four different criteria taken into compression technique to achieve high performance of bit saving as same as high object quality. Those criteria are considered redundancy techniques, including spatial redundancy, temporal redundancy, perceptual redundancy, and statistical redundancy. Spatial redundancy is the process of pixels similarity reduction in a still image or frame known as intra-frame coding. Contrast with the temporal redundancy is the exploration of pixels similarity reduction in two consecutive frames with the same values in the same position known as inter-frame coding. Apart from spatial and temporal redundancy, some detailed information in the picture that the human visual system (eye) could not perceive, especially the high-frequency components. Hence, the process to diminish the number of high-frequency components in the picture is called perceptual redundancy. The last criterion is statistical redundancy defined in the entropy coding

to form the data into the bitstream by deducting the value of data fields based on the probability of content. As a result, video compression targets two main objectives: lossless compression and lossy compression. Lossless compression is a type of data compression which is careful on original data to be precisely reconstructed from the compressed data or the decompressed data and the original data are identical. This technique is primarily used in graphically designed websites like the image with a PNG extension (Portable Network Graphics) or GIF (Graphics Interchange Format).

In contrast to the lossy compression, the decompressed data is extracted to be approximated to the original data. Lossy compression can reduce a massive amount of data storage or a better compression ratio. For this reason, lossy compression is commonly used in real-time communication, including streaming media and internet telephony. It is also played in the role of capacity storage shrinking. These two compression targets are utilized worldwide following the video coding standard configuration. Ordinarily, the video coding standard framework is constructed based on the block-based hybrid video codec principle, which is the successful coding tool to achieve bitstream saving as much as possible by shorting the redundant information from the data signal.

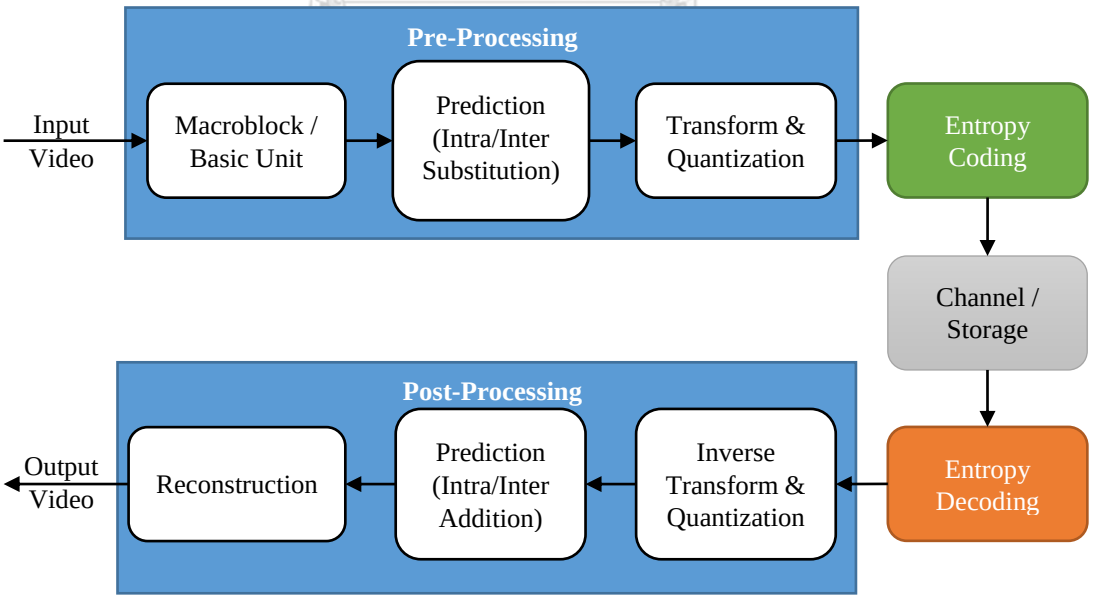


Figure 2.1 Block-based hybrid video codec framework

The hybrid video codec principle is built up from the idea of redundancy properties in the video source. To sum up, this video codec has two core methods at the encoder side, such as pre-processing and entropy coding method, as shown in Figure 2.1. The encoder function generates the represented bitstream of the input video, where bitstream is stored or transmitted over the channel according to the target. Moreover, the decoder side of the video codec consists of the entropy decoding and the post-processing method. The decoder tries to reconstruct the bitstream from the encoder to get the approximated video of the input video (lossy compression) or reconstruct the input identity (lossless compression).

Meanwhile, the pre-processing method contains the macroblock/coding unit partitioning, the prediction (Intra/Inter), the transformation, and the quantization technique. In the brief of the encoder, the input video frame is split into non-overlapped macroblock or basic units with 16x16 pixels commonly used. The prediction block is assigned to remove redundancy block in both still frame and temporal frame known as Intra Coding and Inter Coding. The result of the prediction block is passed through the transformation technique to convert the pixel domain into the frequency domain in the reason for the signal de-correlation. This transformation is used the discrete-cosine transform (DCT) to understand the pattern of frequency from DC (Direct Current) component to AC (Alternative Current) components. The DC component represents the average of the pixel value or Zero frequency component, and the AC components represent the independence of the pixel values or Non-Zero frequency components.

Furthermore, the transformation provides efficient coding by shrinking much of the signal in the pixel domain into more minor of the signal in the frequency domain called the coefficient values that need to encode. These coefficient values with their frequency component types are determined by the quantization gate, which allows or does not pass through the entropy coding to execute the statistical redundancy and generate the bitstream output objecting to the target bit rate. The results of that transformation and quantization are described how the perceptual redundancy applies to the codec before the statistical redundancy performs. Aside from the encoder side, the bitstream is recoded to reconstruct the video back by doing the reverse process of the encoder on the decoder side.

The video coding standard has been published several efficiency standards to compete with the requirements of real-world multimedia applications, as described in Table 2.1 by ITU-T, Motion Picture Experts Group (ISO/IEC MPEG) organization. The joint collaborative team designs some video coding standards on video coding (JCT-VC) of ITU-T and ISO/IEC MPEG organization. The following section presents the overview algorithms in the video codec, HEVC, to comprehend this achieve codec and the outperformance comparison with the previous standard, H.264/MPEG-4 Advanced Video Coding (AVC).

Table 2.1 Video Coding Standard and Applications

No	Years	Organization	Standards	Applications
1	1990	ITU-T	H.261	Video Conferencing
2	1993	ISO/IEC MPEG	MPEG-1	CD-ROM (video storage), video file transfer over the Internet
3	1995	JCT-VC	H.262/MPEG-2	DVD, Video broadcast (digital television, satellite)
4	1996	ITU-T	H.263	Video Conferencing, Surveillance
5	1999	ISO/IEC MPEG	MPEG-4	Surveillance, DVD, Interactive graphics applications (Digital Still Cameras, Digital Video Camcorders, Cellular Media), Interactive multimedia (World Wide Web)
6	1998	ITU-T	H.263+	Video Conferencing, Surveillance
7	2000	ITU-T	H.263++	Video Conferencing, Surveillance

No	Years	Organization	Standards	Applications
8	2003	JCT-VC	H.264/MPEG-4 AVC	Surveillance, Video Conferencing, DVD, Satellite, DSL-based Video On Demand, Digital Still Cameras, Cellular Media
9	2013	JCT-VC	H.265/HEVC (High-Efficiency Video Coding)	Mobile, HDTV, Surveillance, Video Conferencing, DVD, Satellite, DSL-based Video On Demand, Digital Still Cameras, Cellular Media

2.2. High Efficiency Video Coding

JCT-VC published the video coding standard in 2013, known as High-Efficiency Video Coding (HEVC), to code high-resolution and ultra-high-resolution video applications that are ineffective in the previous standard. Consecutively, this HEVC consumes fewer bits than the existent standard, H.264/MPEG-4 AVC, about half of the bit consumptions for equal visual video quality. Moreover, HEVC is also designed to achieve other goals like data loss resilience and parallel processing architectures applications. In briefly, HEVC is built up from various vital features, including flexible coding quadtree [12], flexible prediction modes [13], advanced motion vector prediction (AMVP) [11], the improvement of fractional-sample position interpolation in motion compensation [14], the improvement of the in-loop filter, and novel sample adaptive offset (SAO) [15], and the use only of context-based adaptive binary arithmetic coding (CABAC) [11] to accomplish the current requirements. The framework of the HEVC encoder and its built-in decoder is indicated in Figure 2.2, where the encoder control is the primary role in checking whether the bit is no fluctuation or overflow.

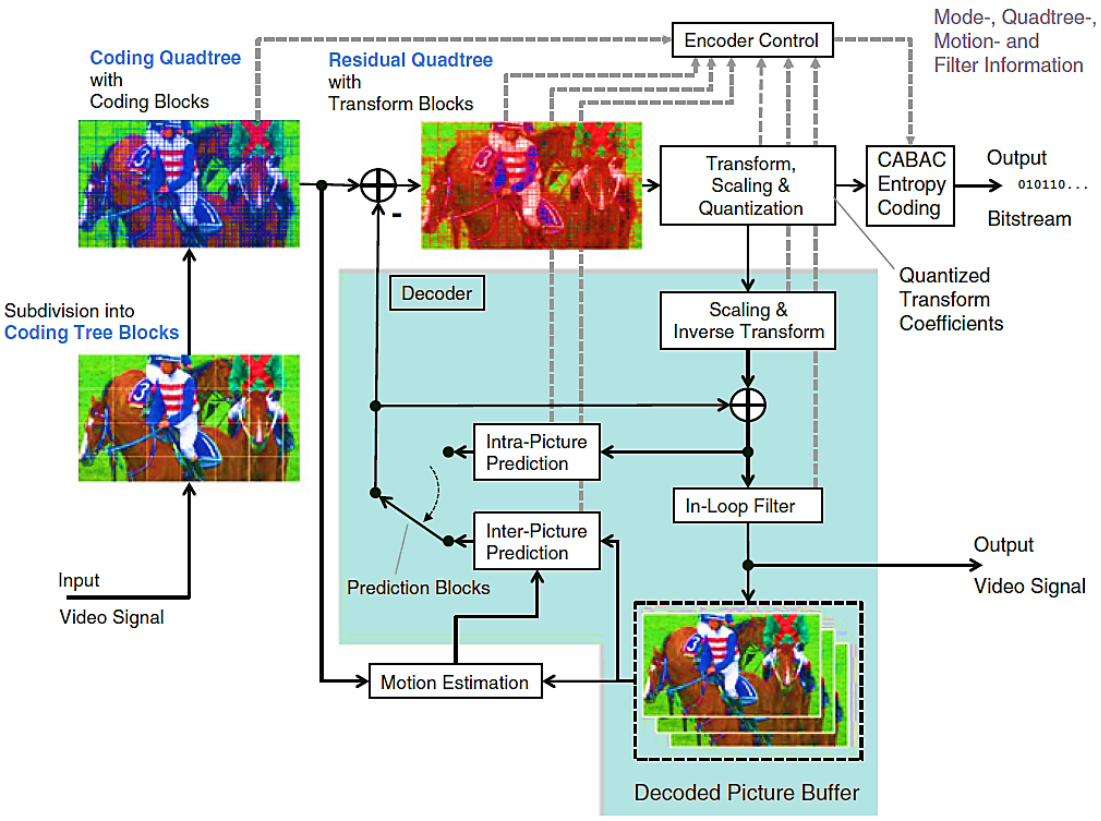


Figure 2.2 HEVC System Overview [11]

In the following subsections, a description of the critical features is presented in concisely understandable.

2.2.1. Picture Partitioning

The high-level segmentation of a still frame in HEVC is performed similarly to the previous one, H.264/MPEG-4 AVC, grounded on the slice conception. The core concepts of slice segment are to achieve error robustness of video packet transmission, to adapt to the maximum transmission unit network constraint, and to be able to apply in parallel processing. HEVC offers two new tools of picture segmentation, such as tiles and wavefront parallel processing (WPP), to overcome the limitations of the parallelization technique engaged in the previous standard. The picture partitioning techniques are described in the following subsections.

a. Slice

The slice segmentation in HEVC remains the same as H.264/MPEG-4 AVC, where each picture can be divided into slices, and these slices are independent of each other except the cross-slice border in-loop filtering. The slice segmentation structure comprises the independent slice segment, dependent slice segment, slice segment boundary, and the slide boundary, as described in Figure 2.3. Slice segmentation decreases the end-to-end delay for ultra-low delay applications; however, it would lead to coding inefficiency if multiple slices are assigned.

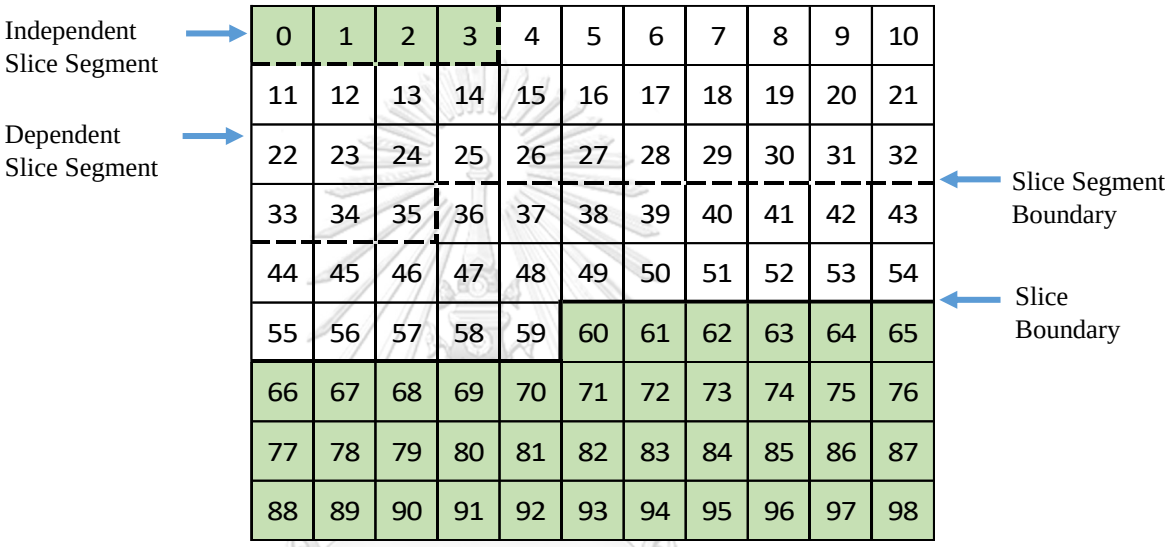


Figure 2.3 Slice Segmentation Structure of a Picture on CTU blocks in HEVC

b. Tile

Tile is another design of the picture partitioning mechanism which is similar to slices. It can divide a picture into multiple non-overlapped columns or rows, as indicated in Figure 2.4. A picture splits into nine different tile sizes following a tile-based raster scan order of CTUs. The double red lines are marked as tile boundaries. Tile usually offers a better coding efficiency than slice since it reduced the spatial distances in tiles, leading to higher spatial correlations between samples within a tile. However, if the number of tiles is enormous, it also leads the coding inefficiency as similar to the slice technique.

0	1	2	3	16	17	24	25	26	27	28
4	5	6	7	18	19	29	30	31	32	33
8	9	10	11	20	21	34	35	36	37	38
12	13	14	15	22	23	39	40	41	42	43
44	45	46	47	52	53	56	57	58	59	60
48	49	50	51	54	55	61	62	63	64	65
66	67	68	69	78	79	84	85	86	87	88
70	71	72	73	80	81	89	90	91	92	93
74	75	76	77	82	83	94	95	96	97	98

Figure 2.4 An Example of Tile Picture Partitioning

c. Wavefront Parallel Processing (WPP)

WPP is the newest picture partitioning for parallel processing, which is enabled in HEVC. It presents many CTU rows in a picture as threads to process the individual CTU rows. This technique proceeds two consecutive CTUs from the top left to the bottom right corner of the picture, as illustrated in Figure 2.5. This process is called **"wavefront"**. As a result, the wavefront dependencies do not let all threads of processing CTU rows start decoding altogether. To simplify, WPP requires storing the content of all CABAC context variables of the second CTU in each CTU row or thread to process the CTU in the following thread. In all, WPP can achieve minor coding efficiency loss due to the propagation of context variables at the second CTU in each CTU row resulting small WPP bitstream compared to a nonparallel bitstream.

In general, WPP is a better technique than slice and tile because it allows a high number of picture partitions with relatively low coding efficiency losses. It does not need any additional pass of in-loop filtering like the other two. However, tile is also suitable in some applications like conversational applications because tiles combined with a tracking algorithm can adjust the size and error protection of the region of interests (ROIs).

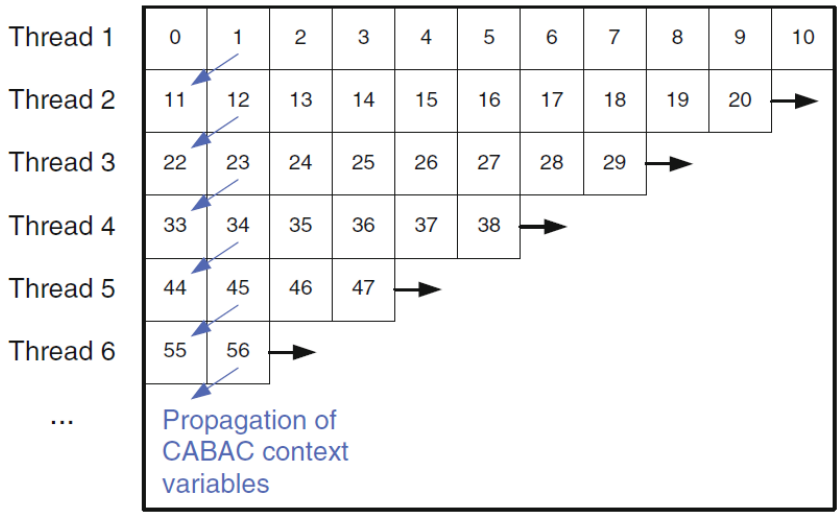


Figure 2.5 Wavefront Parallel Processing Example

2.2.2. Coding Quadtree

The previous standard of video coding like H.264/MPEG-4 AVC, the still picture is generally subdivided from 16x16 block size of luma component to 8x8 block size of luma component, where these blocks are called macroblocks. This concept is to achieve the coding efficiency on picture patterns with different block sizes. However, the coding of high and ultra-high-resolution video becomes progressively essential in multimedia applications. The large coding block size is considered better for motion-compensated prediction and transform coding to compete for efficient coding of such high resolutions. However, the local picture pattern is also necessary to address such details for better perceptual retrieval. This idea leads to the proposal of hierarchical coding block partitioning, called Coding Quadtree [12], in HEVC. The quadtree-based block partitioning is determined by the fast optimal tree pruning in the encoder in Lagrangian rate-distortion cost to get the best partner of hierarchical coding block partitioning. The largest coding block is called the largest coding unit (LCU) or coding tree unit (CTU). Nevertheless, the larger the CTU size, the more encoder/decoder delay may occur. This CTU can be split into multiple coding units (CUs) of variable sizes, and it can vary from 64x64 block size to 8x8 block size of luma samples by using quadtree syntax as shown in example Figure 2.6.

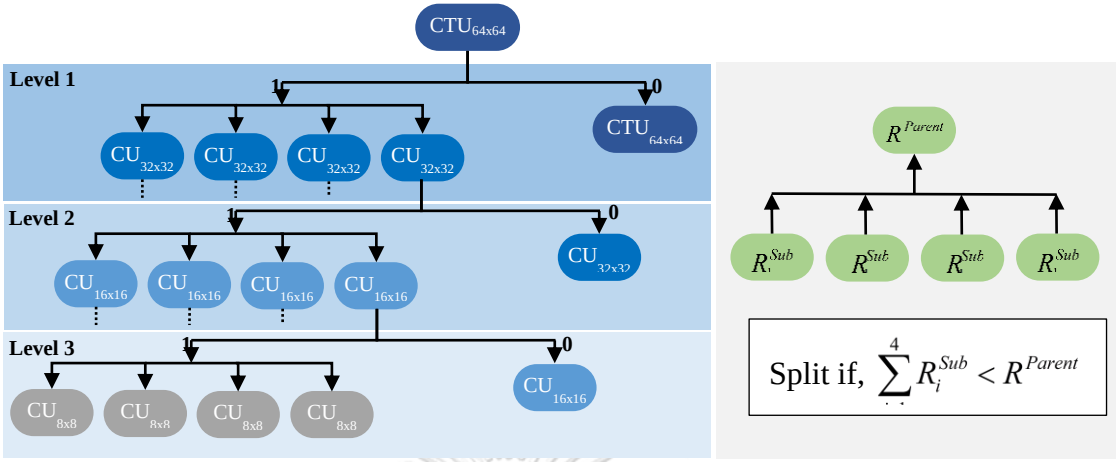


Figure 2.6 Coding Tree Unit (CTU) partitioning example

The CUs procedure is referred to the processing units to which a coding mode is given. The coding mode is the intra-picture prediction mode or motion-compensated prediction mode called the inter-picture prediction mode. Typically, the CUs can be further split into smaller square blocks or non-square blocks according to the prediction modes and the transform coding to get reliable coding structure in each depth of picture pattern.

2.2.3. Prediction Modes

The prediction mode is a signal which assigns in the bitstream to declare whether the prediction is in intra-picture coding mode or inter-picture coding mode. These two modes are designed in HEVC to lead the high performance of decreasing coding redundancy on the still frame (Intra-Picture) and the temporal frames (Inter-Picture).

a. Intra-Picture Prediction

The intra-picture prediction is proposed in HEVC to decrease spatial redundancy to achieve high coding efficiency on a still picture. In this intra-picture prediction mode, the CU with size $N \times N$ would be split into four square block sizes $\frac{N}{2} \times \frac{N}{2}$ or just a single block size $N \times N$ to produce the prediction units (PUs). There

are three main steps to observe the prediction pixel in the intra-picture prediction. The reference sample array construction is the first step of intra-picture prediction to generate the reference pixel in both directions, horizontal and vertical. There are two filtering techniques to get suitable reference pixels, as illustrated in Figure 2.7. The first filtering technique, called strong-intra smoothing filter, generates the reference pixels by applying linear interpolation between the three corner reference samples, *p*[-1][63], *p*[-1][-1], and *p*[63][-1] in case of the prediction block size is equal to 32x32. The reference samples are observed to be sufficiently flat. Two inequalities property is defined as Eq. (2.1) to determine the flatness of the reference sample. Otherwise, another filtering is applied. That filter is a three-tap smoothing filter, and it is computed as Eq. (2.2),

$$|p[-1][-1]+p[2N-1][-1]-2p[N-1][-1]|<(1<<(b-5)) \qquad (2.1)$$

$$p[-1][-1]=(p[-1][0]+2p[-1][-1]+p[0][-1]+2)>>2 \qquad (2.2)$$

, where *b* represents the sample bit depth, *N* is the block size, and "<< and >>" indicates the arithmetic left shift and right shift operation, respectively.

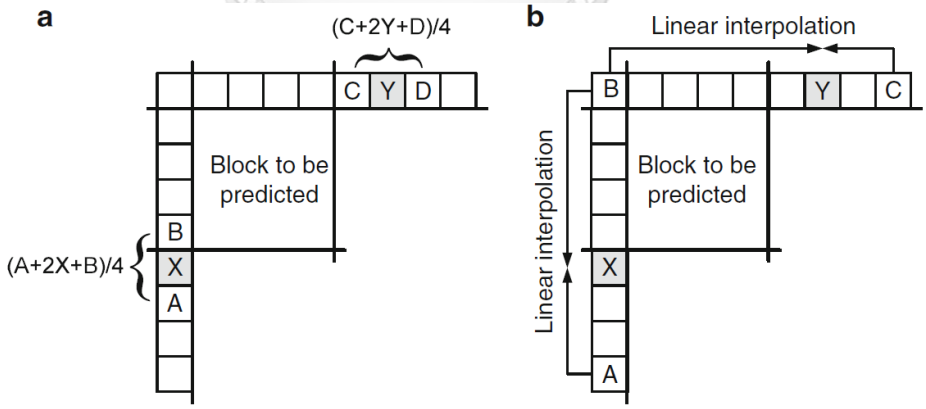


Figure 2.7 Reference Sample Filtering: (a) shows a strong-intra smoothing filter. (b) shows a three-tap filtering

The second step is the sample prediction designed into two main categories in the intra-picture prediction mode. The total of prediction modes in intra-picture prediction is 35 modes, as demonstrated in Figure 2.8. The angular intra-prediction

method is considered the first category, where it is modeled to predict the different directional structures present in the pictures. It consists of 33 directional modes provides better accuracy codec prediction than its previous standard, H.264/MPEG-4 AVC. The predicted pixel value is estimated as in Eq. (2.3) by projecting its location to a reference row of pixels and performing interpolating the two closest reference samples.

$$p_{x,y} = \left(\left(32 - w_y\right) \cdot R_{i,0} + w_y \cdot R_{i+1,0} + 16\right) >> 5 \tag{2.3}$$

, where w_y represents the weighting between the two reference samples, $R_{i,0}$ and $R_{i+1,0}$, corresponding to the projected subpixel location. The index i and w_y are computed based on the displacement d of projection associated with the selected prediction direction as Eq. (2.4),

$$\begin{aligned} c_y &= \left(y \cdot d\right) >> 5 \\ w_y &= \left(y \cdot d\right) \& 31 \\ i &= x + c_y \end{aligned} \tag{2.4}$$

, where & signifies a bitwise AND operation. The parameters c_y and w_y are determined depending on only the coordinate y and the displacement d.

Another category is a group of DC prediction and planar prediction modes. The DC prediction mode provides an approximation average predicted sample value in the luminance blocks of size 16x16 and smaller. This average of the reference samples may introduce the discontinuities along the block boundaries. So the planar prediction mode is modeled to preserve the continuities along the block boundaries by applying an average of two linear predictions as Eq. (2.5),

$$\begin{aligned} p^V_{x,y} &= \left(N - y\right) \cdot R_{x,0} + y \cdot R_{0,N+1} \\ p^H_{x,y} &= \left(N - x\right) \cdot R_{0,y} + x \cdot R_{N+1,0} \\ p_{x,y} &= \left(N + p^V_{x,y} + p^H_{x,y}\right) >> \left(\log_2\left(N\right) + 1\right) \end{aligned} \tag{2.5}$$

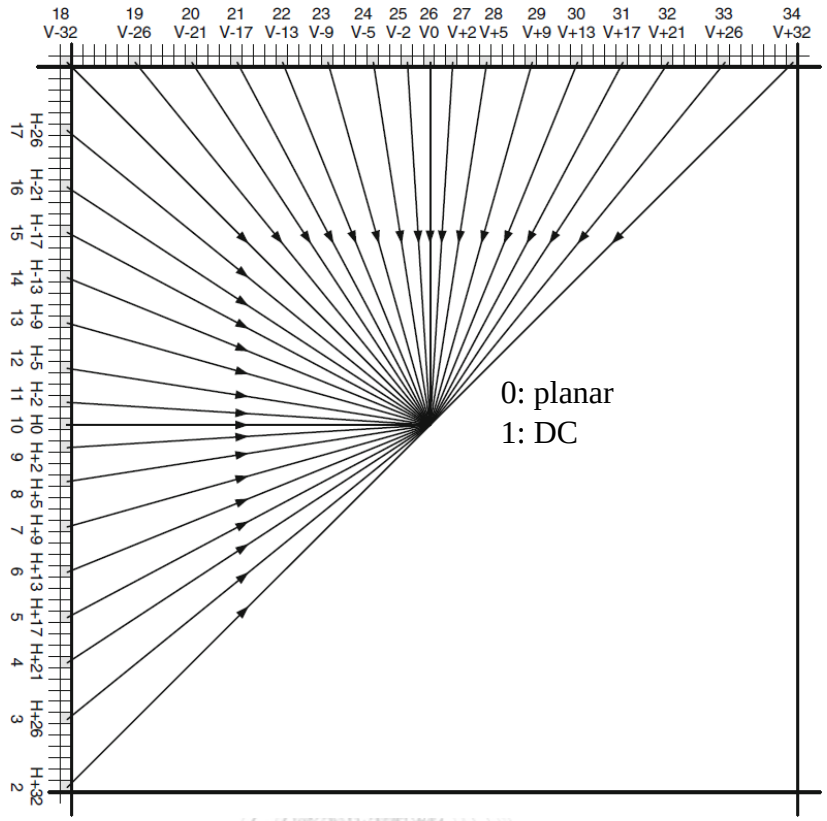


Figure 2.8 Intra-Picture Prediction Modes

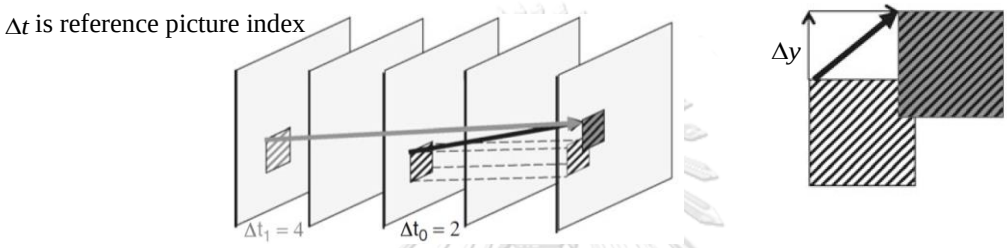
The last step is post-processing for the predicted samples. The function of post-processing is to filter the prediction value to achieve better continuities such that a three-tap [1 2 1]/4 smoothing filter is applied on DC prediction mode.

b. Inter-Picture Prediction

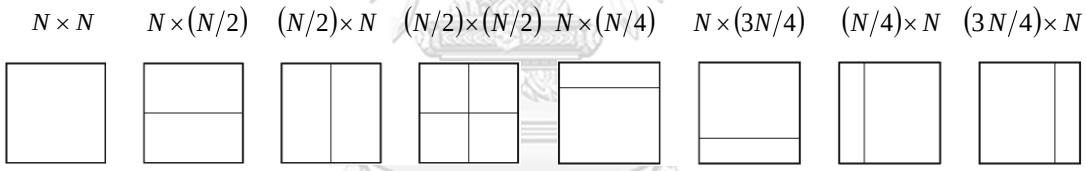
Inter-picture prediction is another redundancy removal method that makes use of the temporal correlation between consecutive frames to observe a motion-compensated prediction (MCP), (Δx and Δy), for a based unit of image samples as illustrated in Figure 2.9 (a). Figure 2.9 (b) indicates all subdivision modes of prediction unit from a CU with size $N \times N$, including those symmetric and asymmetric blocks for inter-picture prediction mode. In total, there are eight different modes of PUs design in HEVC to achieve better prediction on both the DC component and the AC component comparing to H.264/MPEG-4 AVC.

In addition, the motion vector is enhanced in HEVC by applying the advanced motion vector prediction (AMVP) to adopt the motion vector with the

flexible block structure. The motion vector can be observed in the fractional position of the underlying object by applying the interpolation technique to achieve more accurately capturing the continuous motion. Hence, the interpolation filter has been re-designed. The tap-lengths are increased in HEVC by using 7/8 tap filter kernels for the luma channel and 4-tap filter kernels for the chroma channel of each PUs to get high precision interpolation filtering, especially in the high-frequency range. After interpolation, the final prediction stage is performed, known as weighted sample prediction, by averaging two motion-compensated predictions.



(a) Motion Vector Example



(b) Prediction Unit of Inter-Picture Prediction

Figure 2.9 Motion Vector and Prediction Unit of Inter-Picture Prediction

2.2.4. Transform and Quantization

Before the transform and quantization procedure, the residual signal is computed by subtracting the original picture block with the prediction block. Then, the flexible two-dimensional transforms of various sizes from 32x32 to 4x4 are proposed in HEVC, where those values are a finite approximation to the discrete cosine transform (DCT). Furthermore, if the prediction block is a 4x4 luma intra-prediction residual block, the 4x4 integer discrete sine transform (DST) is implemented. The result of the transform provides the transform coefficients (*Coeff*), then subject to the quantization to obtain the quantized transform coefficients (*levels*) by dividing *Coeff* with the quantization step size (*Qstep*). In the end, the entropy

coding encodes those *levels* to generate the *bitstream* representing the input block. The whole procedure can be illustrated in Figure 2.10.

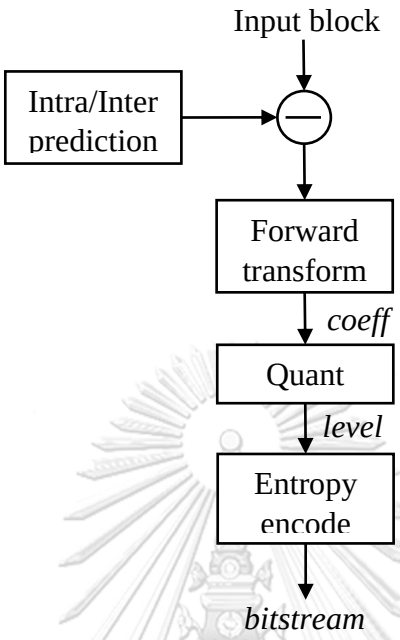


Figure 2.10 Transform and Quantization Procedure in Encoder

2.2.5. In-Loop Filters

In general, the codec is designed to reduce the redundancies as much as possible to achieve high-efficiency coding. The in-loop filter is one of the necessary modules to enhance the quality of the reconstructed picture in both encoding and decoding to minimize the residual error, where it is applied after the inverse process of quantization. There are two sub-algorithms in the in-loop filter, including a deblocking filter and a sample adaptive offset (SAO). HEVC introduces a deblocking filter to attenuate the discontinuities at block boundaries and SAO to reduce the ringing artifacts of the reconstructed picture. These two filters can better redundancy removal in both spatial and temporal context compared to H.264/MPEG-4 AVC.

2.2.6. Entropy Coding

The entropy coding is the last part of the video codec, which uses statistical properties to compress data. An earlier video coding standard, H.264/AVC, introduced CABAC and context-adaptive variable length coding (CAVLC). Typically, CAVLC provides the reducing implementation price cost of lower compression efficiency and its

bitrate overhead relative to CABAC. Thus, CABAC became the single entropy coding algorithm that is used in HEVC. In conclusion, the CABAC method can challenge parallel processing architectures and provide high coding efficiency.

Table 2.2 Limit of HEVC Profile and Level Definitions

Level	Max Luma Picture Size (samples)	Max Luma Sample Rate (samples/s)	Main Tier Max Bit Rate (1000 bits/s)	High Tier Max Bit Rate (1000 bits/s)	Min Compression Ratio
1	36 864	552 960	128	–	2
2	122 880	3 686 400	1500	–	2
2.1	245 760	7 372 800	3000	–	2
3	552 960	16 588 800	6000	–	2
3.1	983 040	33 177 600	10 000	–	2
4	2 228 224	66 846 720	12 000	30 000	4
4.1	2 228 224	133 693 440	20 000	50 000	4
5	8 912 896	267 386 880	25 000	100 000	6
5.1	8 912 896	534 773 760	40 000	160 000	8
5.2	8 912 896	1 069 547 520	60 000	240 000	8
6	35 651 584	1 069 547 520	60 000	240 000	8
6.1	35 651 584	2 139 095 040	120 000	480 000	8
6.2	35 651 584	4 278 190 080	240 000	800 000	6

2.2.7. The HEVC Profile and Level Definitions

HEVC is designed to fulfill advanced multimedia applications, especially for high video definition. Three profiles are targeting different application requirements represented the Main, Main 10, and Main Still Picture profiles. Only two configurations, Main and Main Still Picture profiles, supports 8 bits per sample in a video, and the other profile supports 10 bits per sample. Additionally, HEVC supports 13 video definitions of 176x144 pixels to 7680x4320 (8kx4k) pixels, and it can reach the minimum compression ratio by 2 to 8 following the sample size. Table 2.2 shows

the total video definitions with the maximum luma sample rate and the minimum compression ratio. For example, if a video definition size is 176x144 pixels, then the total sample is 25344, less than 36864 samples. So, in this case, the video definition is in level 1, and the user can use the maximum bitrate is 128 kbit/second.

Table 2.3 The Comparison of HEVC and H.264/MPEG-4 AVC

Tool	H.264/MPEG-4 AVC	HEVC
Coding Unit Size	Fix 16x16 block size	64x64 to 8x8 block sizes
Partitioning	• Intra: 3 partitioning (16x16, 8x8, and 4x4) • Inter: 4 partitioning (16x16, 16x8, 8x16, 8x8)	• Intra: Current CU size down into to 4x4 (symmetric) • Inter: Current CU size down into four symmetric and four asymmetric
Intra Prediction	Nine directional modes	35 directional modes
Motion Prediction	Spatial Median (3 blocks)	Advanced Motion Vector Prediction Spatial + Temporal
Transform	Integer DCT 8x8, 4x4	Square Integer DCT from 32x32 to 4x4 + Integer DST Luma Intra 4x4
Interpolation	• ½ Pixel 6-tap • ¼ Pixel bi-linear	• ¼ Pixel 7 or 8 tap Luma • ⅛ Pixel 4-tap Chroma
Entropy Coding	CABAC or CAVLC	CABAC with parallel operations
In-Loop Filter	Deblocking Filter	Deblocking Filter and SAO

2.2.8. The Comparison of HEVC and H.264/MPEG-4 AVC

In summary, HEVC is the current video coding standard that was published in 2013. It introduces several advanced tool techniques, including the coding unit size selection, partitioning, intra-picture prediction, inter-picture prediction or motion prediction, flexible transform size, deeper fractional interpolation, and an in-loop filter. Those advanced tools can lead the codec to improve the quality of the reconstructed picture. It can improve the performance of redundancy removal as high as possible compared to the previous standard. It can notify 50 percent of bit deductions with the same quality visual picture versus H.264/MPEG-4 AVC. Table 2.3 indicates the main tool comparisons of HEVC and H.264/MPEG-4 AVC that can aid the codec improvement in the HEVC.

2.3. Rate Control Algorithm

Rate control is a necessary module to control bit allocation to achieve the given bit budget after the encoding process and minimize distortion rate to get higher quality performance after the decoding process. In general, there are two main objectives to discuss in rate control; they are bit allocation and quantization parameter (QP) computation. In the bit allocation part, the bit budgets must be generated carefully to assign to each coding level, such as group of pictures (GOP) level, picture level, and basic unit level to control bits overflow. In addition, to achieving the target bitrate, QP is taken into account because it has a higher correlation of assigning bits. If QP is large, bit allocation will be less. Rate-Distortion (R-D) performance has been considered prior knowledge to generate a function related to QP. Several rate control algorithms are designed to adapt to the standards. The following subsections are briefly described three intuitive rate controls [16], [17], and [18].

2.3.1. Q-Domain Rate Control

Q-domain rate control [16] is proposed for the MPEG video coding standard, where it modeled the correlation between bit rate and quantization parameter (QP) as Eq. (2.6). This model is also called a quadratic rate-quantizer. The *R-Q* curves plots were constructed as in Figure 2.11, indicating the bit consumption comparison curves

of the spatial frame and the temporal frames. This model can perform QP adaptively as an indicator to reach the target bit.

$$R = \alpha \cdot Q^{-1} + \beta \cdot Q^{-2} \tag{2.6}$$

, where R is the target bitrate, Q is the quantization parameter, α and β are the coefficients related to video content.

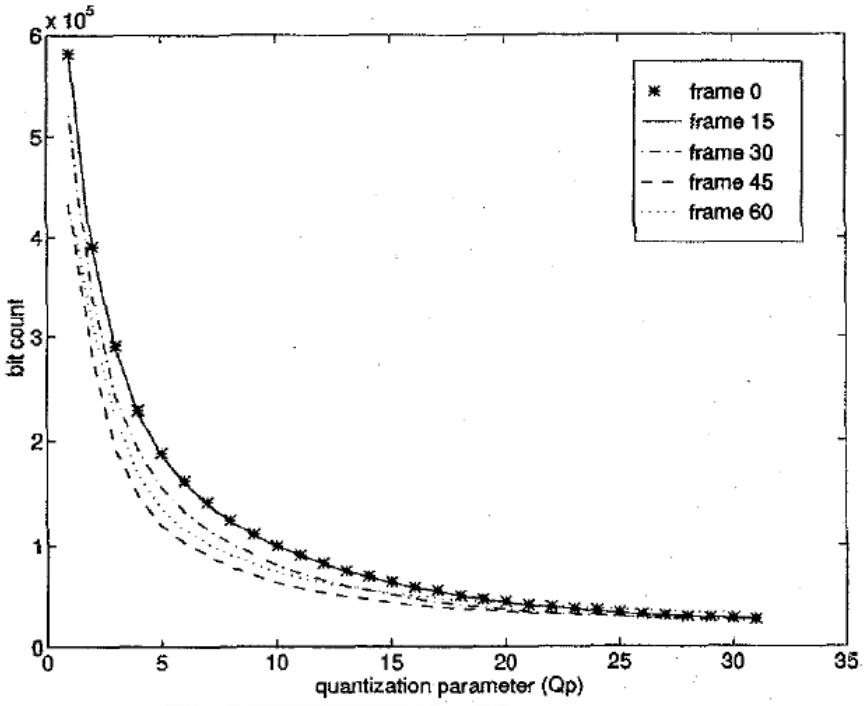


Figure 2.11 R-Quantization Model Plot

2.3.2. Rho-Domain Rate Control

In the encoder control, the *R-Q* model is not yet enough to adapt the bit allocation *R* with the distortion *D* behavior of a transform coding system. The rho-domain rate control [17] introduces the new concepts of the characteristic *R-D* curve based on DCT video coding. This novel rate control observed that the zeros play a crucial role in transforming the images to allow the model to discriminate the zero and non-zero coefficients. The model is determined as the Eq. (2.7).

$$R = \theta \cdot (1 - \rho) \tag{2.7}$$

, where ρ is the percentage of the zeros among the quantized transform coefficient, and θ is a coefficient related to the video source.

2.3.3. Lambda-Domain Rate Control

As mentioned above, QP is the detracting factor considered in the rate control model, Q-domain rate control, and rho-domain rate control. QP is the only parameter with higher effectiveness in picture quality performance when other parameters are fixed. New rate control is publicized with the latest video coding standard, HEVC, to have high flexibility in various video contents in various applications. This new rate-control is called R-lambda rate control [18]. There are two flexible steps, computing a model λ of the relationship between picture qualities with bitrate and analyzing QP by using λ. For the first step, the Hyperbolic R-D model is defined to compute λ related to bitrate *R*, as in Eq. (2.9).

$$D(R) = C \cdot R^{-K} \tag{2.8}$$

$$\lambda = -\frac{\partial D}{\partial R} = C \cdot K \cdot R^{-K-1} = \alpha \cdot R^{\beta} \tag{2.9}$$

, where *C* and *K* are coefficients related to the source characteristics. From Eq. (2.9), the λ can be simplified to compute correlated to bit per pixel (*bpp*) as Eq. (2.10).

$$\lambda = \alpha \cdot bpp^{\beta} \tag{2.10}$$

Afterward, λ is defined, the QP can be calculated as Eq. (2.11).

$$QP = 4.2005 \cdot \ln(\lambda) + 13.7122 \tag{2.11}$$

Subsequently, the encoding procedure in each frame or a CTU, all coefficients need to be updated. α and β values are updated following actual generated bits, QP value, and λ value using Eq. (2.11) to (2.14).

$$\lambda_{comp} = \alpha_{old} \cdot bpp_{real}^{\beta_{old}} \tag{2.12}$$

$$\alpha_{new} = \alpha_{old} + \delta_{\alpha} \cdot \left(\ln(\lambda_{real}) - \ln(\lambda_{comp})\right) \cdot \alpha_{old} \tag{2.13}$$

$$\beta_{new} = \beta_{old} + \delta_{\beta} \cdot \left(\ln(\lambda_{real}) - \ln(\lambda_{comp})\right) \cdot \ln(bpp_{real}) \tag{2.14}$$

, where bpp_{real} is calculated from actual generated bits, α_{old} and β_{old} are α and β values used in the coded frame, $\delta_{\alpha} = 0.1$ and $\delta_{\beta} = 0.05$.

The bit allocation proceeding, including the GOP level, picture level, and the LCU level, is assigned. In GOP level bit allocation, the target bits in a GOP can be computed by Eq. (2.15) and (2.16).

$$T_{AvgPic} = \frac{R_{PicAvg} \cdot (N_{coded} + SW) - R_{coded}}{SW} \tag{2.15}$$

$$T_{GOP} = T_{AvgPic} \cdot N_{GOP} \tag{2.16}$$

, where T_{AvgPic} is the average target bit per picture, R_{PicAvg} is the average target bit per picture $(R_{PicAvg} = Target\_Bitrate / framerate)$, N_{coded} is the number of pictures already been code, R_{coded} is the bit cost on the picture already been coded, N_{GOP} is the number of pictures in the current GOP, SW is the other number ($SW = 40$), and T_{GOP} is the target bits for current GOP. For picture level, a bit budget can be assigned in Eq. (2.17).

$$T_{CurrPic} = \frac{T_{GOP} - Coded_{GOP}}{\sum_{NotCodedPictures} wp_i} \cdot wp_{CurrPic} \tag{2.17}$$

, where $T_{CurrPic}$ is the target bit budget for the current picture, $Coded_{GOP}$ is the bits budgets for coded frames in the current GOP, and $wp_{CurrPic}$ is the weight of each picture. The weight value depends on the position of the picture in the coding structure. In the LCU level, suppose Bit_{header} is the estimated bits of all headers, $wp_{CurrLCU}$ is the weight of each LCU, and $Coded_{Pic}$ is the generated bits for coded LCUs in the current picture. Hence, the target bit of each LCU is calculated as Eq. (2.18).

$$T_{CurrLCU} = \frac{T_{CurrPic} - Bit_{header} - Coded_{Pic}}{\sum_{NotCodedLCUs} wp_i} \cdot wp_{CurrLCU} \tag{2.18}$$

2.4. Constrained Optimization

In mathematical optimization, constrained optimization maximizes or minimizes the objective function to a set of constraints for obtaining certain variables in the presence. If $f(x,y)$ is a nonlinear function, then the optimum values can be determined at the boundaries or between the constraints, as shown in Figure 2.12. If $f(x,y)$ is a linear function, then the optimum values can be determined at only the boundaries, as shown in Figure 2.13.

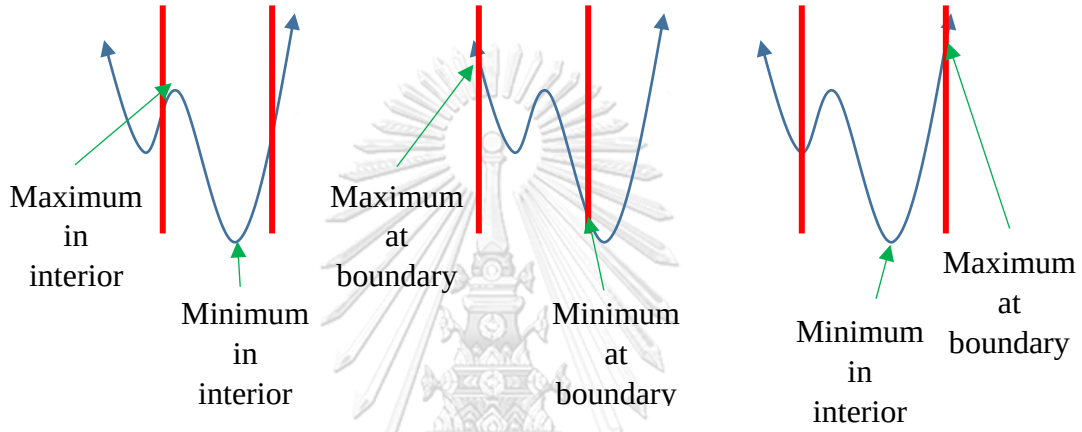


Figure 2.12 Optimum Values of a Nonlinear Function

Figure 2.13 Optimum Values of a Linear Function

The constrained optimization techniques can be grouped into two categories, including gradient-based approach [19], [20], [21], [22], and non-gradient-based approach. The sub-section below describes each approach in detail.

2.4.1. Gradient-Based Approach

In common, the constrained optimization problems are converted into unconstrained optimization problems to define the relationship between the objective constraint and the desired parameters. The Lagrange multiplier technique [21] is a

popular gradient-based optimization used to solve a constrained optimization. The technique has been proposed to determine the maximum or minimum of a multivariable function *f*. Suppose g is the constraint function, *x* and *y* represent the variable of function *f* and *g*, and λ is the Lagrange multiplier. The Lagrangian $\mathcal{L}$ can be defined as expressed in Eq. (2.19) to translate the constrained optimization to unconstrained optimization in order to determine the optimum value of the function *f*.

$$\mathcal{L}(x, y, \lambda) = f(x, y) - \lambda \cdot g(x, y) \tag{2.19}$$

Then, the Lagrange multiplier can be extracted by setting the gradient of $\mathcal{L}$ equal to the zero vector:

$$\begin{aligned} &\Delta\mathcal{L} = 0 \\ &\equiv \begin{bmatrix} \frac{\partial}{\partial x}\mathcal{L}(x, y, \lambda) \\ \frac{\partial}{\partial y}\mathcal{L}(x, y, \lambda) \\ \frac{\partial}{\partial \lambda}\mathcal{L}(x, y, \lambda) \end{bmatrix} = 0 \end{aligned} \tag{2.20}$$

In summary, there are three main steps to optimize the constrained optimization problems using the Lagrange multiplier:

- Step 1: Introduce the unconstrained function $\mathcal{L}$
- Step 2: Set the gradient of equal to the zero vector
- Step 3: Choose the solution that observes *f* as the smallest or the highest value according to the target.

Although the Lagrange multiplier technique can observe the solution, it is most likely stuck at the local optima.

Figure 2.14 indicates an example of the visualization of local optima and optimum global point of a function *f* in a boundary constrained function *g*. In common, the gradient-based approaches are not able to handle discrete, discontinuous, multi-model, and mixed discrete-continuous problems, as shown in Figure 2.15. The best way to solve the above problems is to define a gradient-free optimization algorithm. Consequently, the gradient-free techniques are designed to

find the optimal global solutions in many ways. The subsection below describes the common gradient-free or non-gradient-based approaches in detail.

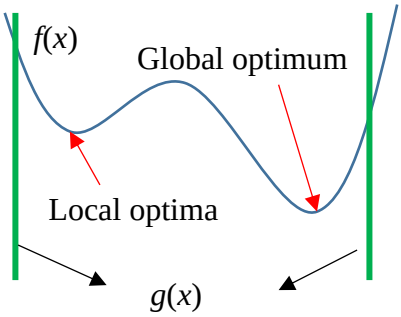


Figure 2.14 Local Optima and Global Optimum Point

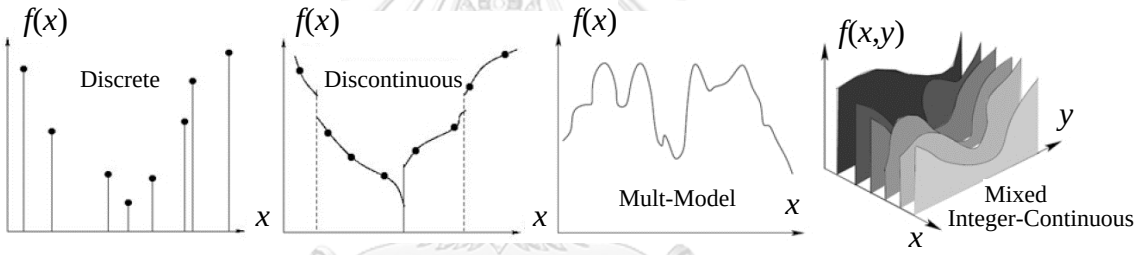


Figure 2.15 Gradient-Based Optimization Problems

2.4.2. Non-Gradient-Based Approach

Many non-gradient-based methods are proposed to characterize the imitation mechanism that discovers the nature of the problem. The most commonly used methods are evolution strategies (ES) [23], simulated annealing (SA) [24], genetic algorithm [25], and particle swarm optimization (PSO) [26] - [27]. These non-gradient-based methods can be called Evolution Algorithms (EAs). Additionally, PSO is the most powerful compared to all EAs because of its simplicity and convergence speed characteristics [28]. Besides, PSO has been successfully implemented to solve various constrained optimization problems in [29], [30], [31], and [32]. Precisely, PSO is known as a stochastic or population-based algorithm, which applies the position and velocity of particles to update the state to achieve the global solution. Initially, let us suppose $X = \{x_1, x_2, x_3, \ldots, x_n\}$ represents as points in n-dimensional

space. r_1 and r_2 are the random uniform distribution in the range [0,1], c_1 is the cognitive constant, c_2 is the social constant, P_k^i represents as the best local position of particle i, P_k^g represents as a swarm or the best global position of particles at iteration k, w is the inertia constant. The velocity update of the particle is computed as in Eq. (2.21)

$$v_{k+1}^i = wv_k^i + c_1 r_1 \left(p_k^i - x_k^i\right) + c_2 r_2 \left(p_k^g - x_k^i\right) \tag{2.21}$$

The position update of the particle is then computed as in Eq. (2.22).

$$x_{k+1}^i = x_k^i + v_{k+1}^i \tag{2.22}$$

The entire procedure of PSO, as shown in Figure 2.16, can be summarized into five steps:

- Step 1: Define the objective functions or fitness functions $f(x_k^i)$
- Step 2: Initialize a set of particles positions and velocities
- Step 3: Evaluate the fitness functions $f(x_k^i)$
- Step 4: Update the position and velocity in Eq. (2.21) and Eq. (2.22)
- Step 5: Repeat steps 3-4 until converge to the stopping criteria

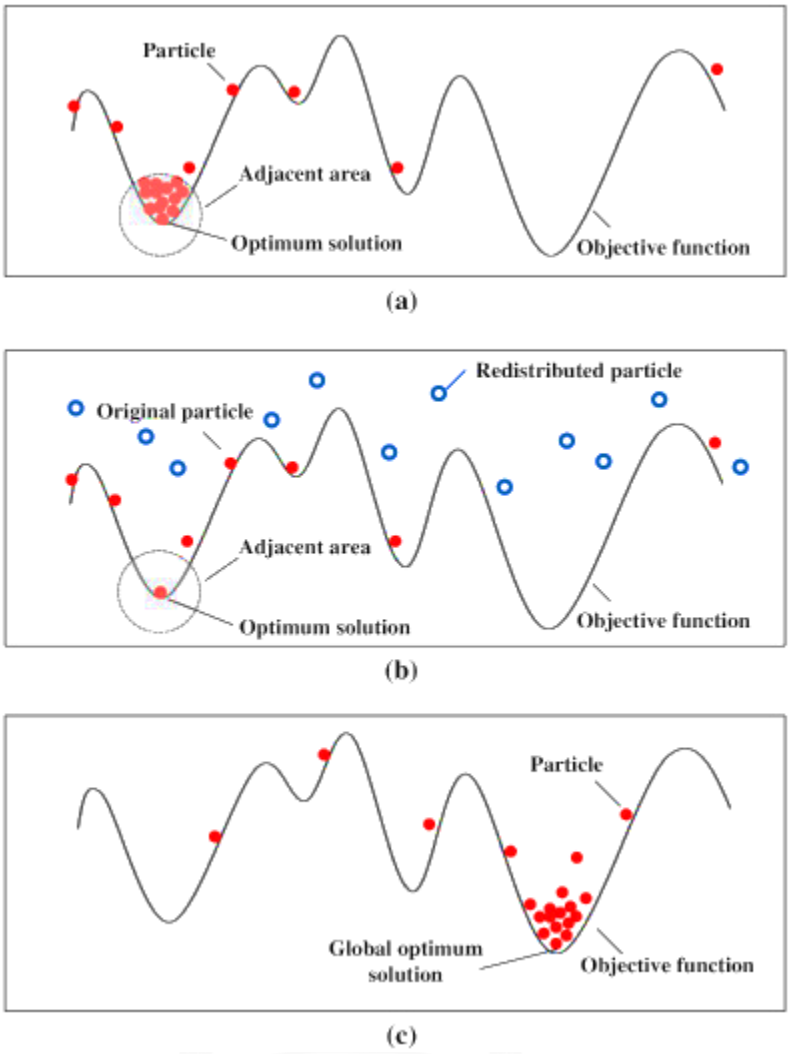


Figure 2.16 Particle Swarm Optimization Example: (a) Initial particle and determine the objective function, (b) Redistributed particle, (c) Solution that meet the criteria of an objective function

2.5. Deep Learning Algorithm

2.5.1. Neural Network

The neural network (NN) or artificial neuron [33] is inspired by the biological neuron concept, as shown in Figure 2.17, which contains neurons, dendrites (information coming from other neurons), and synapses (information output to other neurons). The input of neurons represents dendrites, and the output of neurons represents synapses.

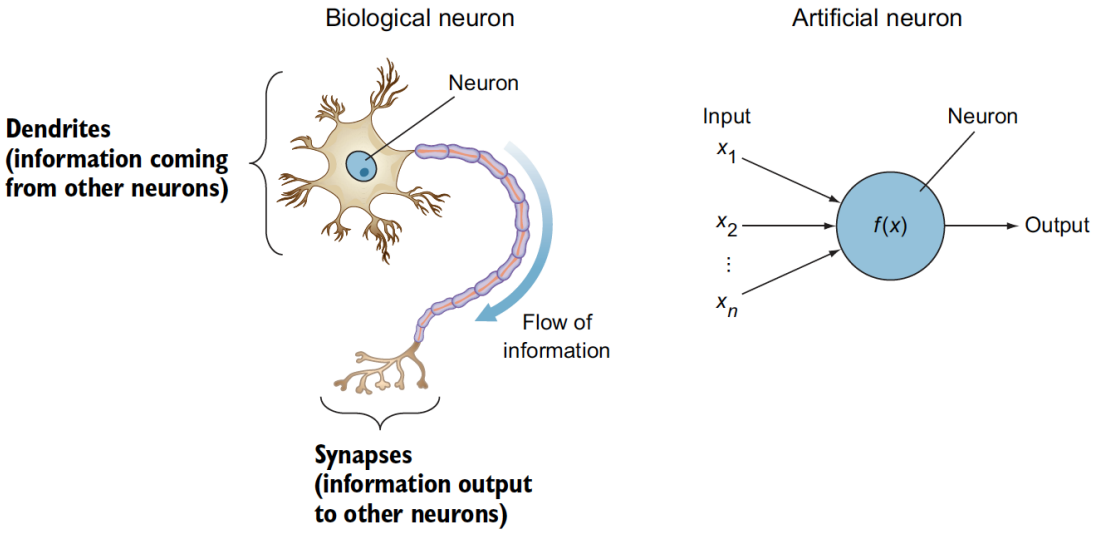


Figure 2.17 Artificial Neurons and Biological Neurons

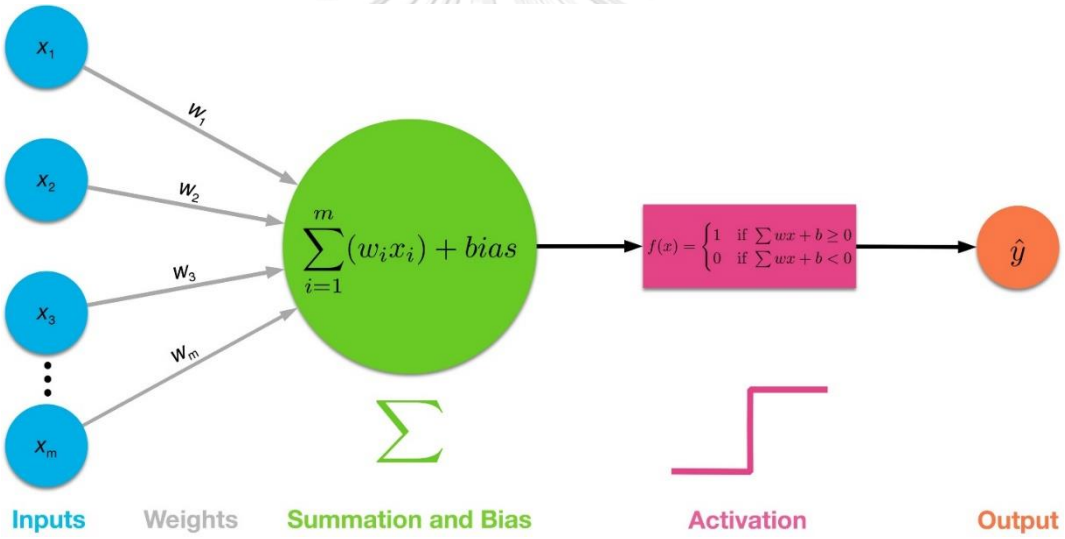


Figure 2.18 Forward Pass of A Neural Network

Additionally, multiple NN architectures have three main layers: the input layer, hidden layers (containing the NN layers [34]), and the output layer, as illustrated in Figure 2.19. NN is learned using the backpropagation technique, known as gradient-based optimization, presented in Section 2.4.1. The parameters of NN, called weight, are updated if the objective function does not meet the stopping criteria.

Let define X as the input vector, W is weight parameters, and Y is the target output. Then, the forward pass of a NN or perceptron is performed by applying a

weighted sum and passing through the activation function *f* to produce the estimated output $\hat{Y}$, as indicated in Figure 2.18. The forward pass can be formulated as in Eq. (2.23).

$$\hat{Y} = f\left(\sum_i (w_i x_i) + bias\right) \tag{2.23}$$

Furthermore, the various activation functions have been used in the NN [35], such as:

- A linear transfer function: $f(x) = x$
- A Heaviside step function: $f(x) = \begin{cases} 0, & if\ w \cdot x + b \le 0 \\ 1, & if\ w \cdot x + b \ge 0 \end{cases}$
- Sigmoid/logistic function: $f(x) = \dfrac{1}{1+e^{-x}}$
- Softmax function: $f(x) = \dfrac{e^x}{\sum_i e^{x_i}}$
- Hyperbolic tangent function: $f(x) = \dfrac{\sinh(x)}{\cosh(x)} = \dfrac{e^x - e^{-x}}{e^x + e^{-x}}$
- Rectified linear unit (ReLU) function: $f(x) = \max(0, x)$

leaky ReLU function: $f(x) = \max(0.01x, x)$, etc.

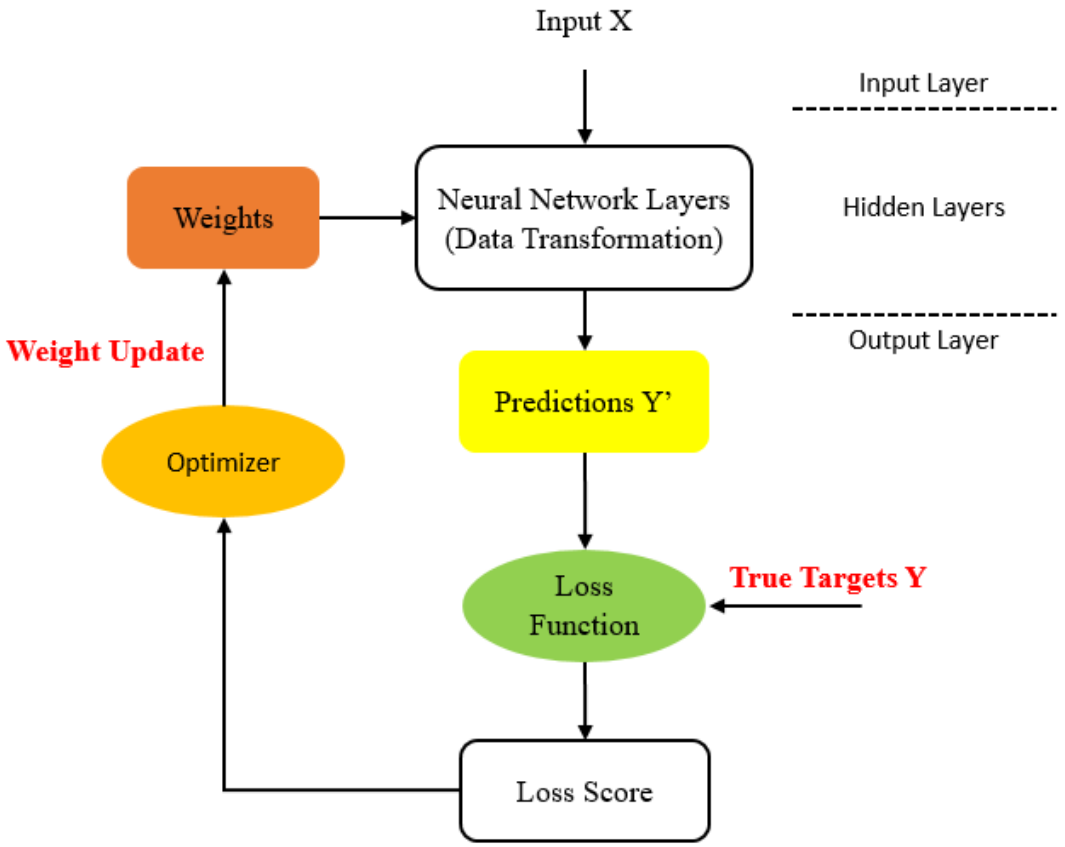


Figure 2.19 Neural Network Learning Procedure

2.5.2. Convolutional Neural Network

Convolutional Neural Networks, known as CNNs, are based on the neural network (NN) architecture, made up of neurons with learnable weights and biases by learning the local patterns of inputs. Additionally, the convolution layer is the core building block of CNN that does the most computational heavy lifting. Figure 2.21 shows the examples of feature extraction using CNNs, consisting of a set of learnable filters to extract the representation of the input color image size 32x32. Each filter is applied across the width and height of the input volume, and the destination pixel is computed using dot products operation between the entries of the filter and the input image. The convolution operation and output size of convolution can be determined as Eq. (2.24) and Eq. (2.25), respectively.

$$y_c = \sum_{i=1}^{k\times k}\left(w_i \times I_{i,c}\right) + b_c \tag{2.24}$$

$$O = \frac{\left(n-k+2p\right)}{s} + 1 \tag{2.25}$$

, where y_c is output, w is a window with size $k \times k$, I represents an image with size $n \times n$, convolution function on c cell output of image I, and O is output size of convolution, p is the padding, and s is the stride.

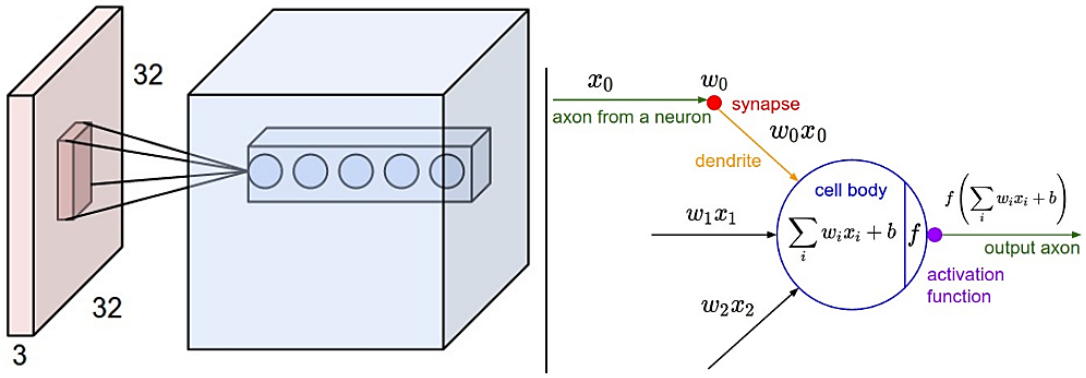


Figure 2.20 An example input 32x32x3 pass through the neurons in the Convolution layer

For example, Figure 2.21 shows the convolution operation on the local patch of the image with a size 8x8. Besides, the total parameters of one convolution layer can be defined as in Eq. (2.26),

$$T_P = k \times k \times n_c + 1 \tag{2.26}$$

, where T_P represents the total number of parameters and n_c is the total size of kernel size. For this example, the filter kernel size is 3x3 with one filter. The convolution layer will have weights to a 3x3x1 region in the input volume for a total of 10 parameters, including nine weights w_i and one bias b.

Furthermore, the first LeNet CNN architecture [36] is proposed to implement a feature extraction for digit classification, as shown in Figure 2.22. It contains convolutional layers, pooling layers, fully connected layers (FCs), and the activation

function layer using rectified linear units (ReLU). The first layer has always been the convolutional layer in which consists of filters implemented on the input image sequence and its size. The numbers of convolutional layers determine the complexity of the network. The more it has, the more it is complex.

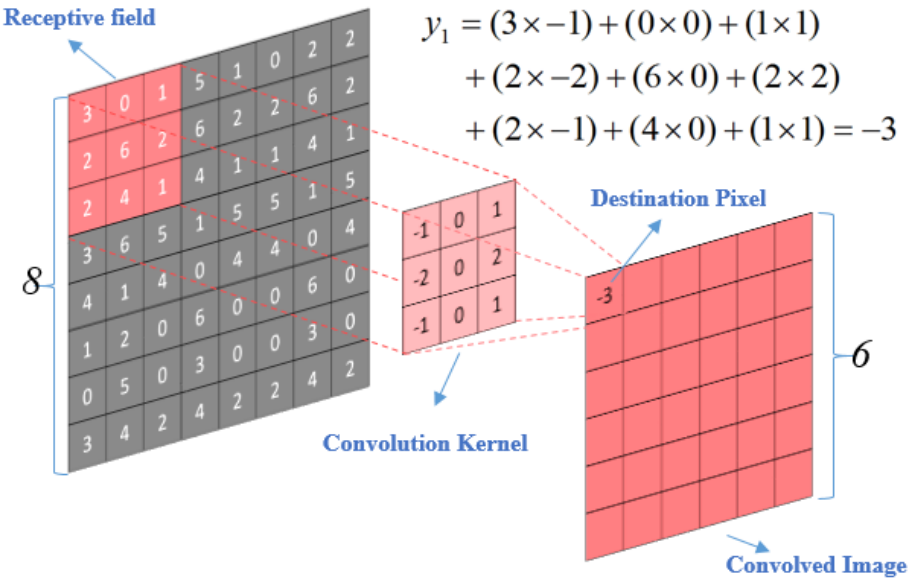


Figure 2.21 Convolution Operation

In most cases, there are multiple convolutional layers applied in one network for generating feature maps. Apart from the convolutional layers, a pooling layer is essential for subsampling the spatial dimension of a feature map. A max-pooling selects only the maximum value and helps to reduce noises from the input. After obtaining the features from the convolutional layers, the FCs are applied to get the classification result by flattening the output from the previous layer and then connecting with the FCs. There is the activation function, ReLU, placed at the hidden layers in the CNNs and used for transforming the batch data to obtain good gradient descent for fast learning. And then, the loss function has computed the penalty between a predicted class and a ground truth label. The standard approach for the loss function is softmax with cross-entropy loss. Since the CNNs are sparsity, share weights, and are not fully connected, it does not connect for every neuron but only a

few from the previous one. Hence, the CNNs have been widely applied instead of the NNs.

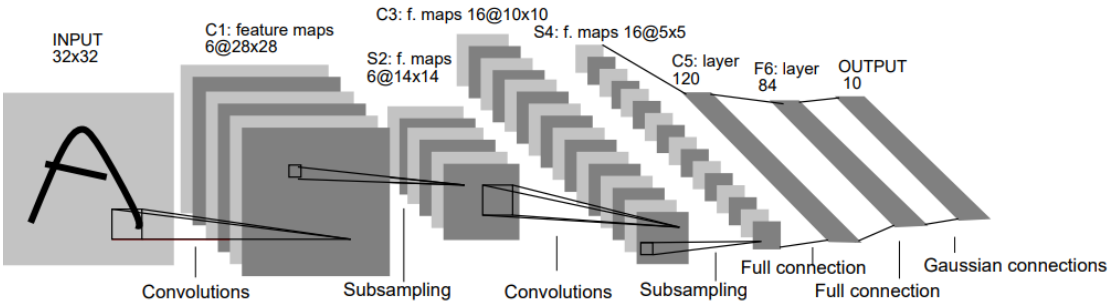


Figure 2.22 LeNet-5 architecture [36]

2.6. Literature Review

Based on our literal study, the visual quality enhancement in HEVC can group into three main approaches, including the encoder rate control based [37], [38], [39], and [40], the frame rate up-conversion [41], [42], and [43], and decoder convolution neural network (CNN) [44], [45], and [46].

2.6.1. Encoder Rate Control Approach

The encoder rate control approach considers an algorithm defined as the effective parameters to update or change the current rate control method in the standard. The low-delay rate control for consistent quality using distortion-based Lagrange multiplier is proposed in [38]. The main algorithm of this paper is to replace the relationship between the Lagrange multiplier λ and the bit rate R (R-λ) into the relationship between the Lagrange multiplier λ and the distortion D (D-λ). This new relationship can be derived from the introduced hyperbolic R-D model in HEVC, as the prove below.

$$D(R) = K \cdot R^{-C} \tag{2.27}$$

$$\Rightarrow \lambda = -\frac{\partial D}{\partial R} = C \cdot K \cdot R^{-C-1}$$

$$\Leftrightarrow \lambda = C \cdot K^{-\frac{1}{C}} \cdot D^{\frac{C+1}{C}}$$

$$\Leftrightarrow \lambda = \gamma \cdot D^{\tau} \tag{2.28}$$

, where *C* and *K* are the parameters related to the characteristic of the video source γ and τ are both new coding constants, and D is the distortion measure by calculating the mean squared error (MSE) between the original coding unit and the reconstructed coding unit in frames. As a result, this technique can get a more accurate rate regulation with lower video quality fluctuation, and it has been designed for the non-hierarchical structure. They can improve by an average of 0.23 dB compared with non-hierarchical in the low-delay P configuration of HEVC reference software. Generally, the performance of original rate control in HEVC using hierarchical structure is better than non-hierarchical [18], about 0.26 dB on average. Consequently, the hierarchical structure is set as the default HEVC general test condition in [47].

Another encoder rate control based is proposed in [39] by modifying the bit allocation of the GOP level. The modified GOP bit allocation can be formulated as Eq. (2.29).

$$T_{GOP}(i) = \left(R_{PicAvg} - \frac{V(i)}{N_{PicRem\_IP}} \right) \times N_{GOP} \tag{2.29}$$

, where *i* represents the *i*-th GOP in the current Intra period, *V*(*i*) is the encoder buffer occupancy before encoding the *i*-th GOP, N_{GOP} is the number of frames in one GOP, R_{PicAvg} is the average target bit per frame, and N_{PicRem_IP} is the number of remaining pictures in current Intra period. The author claims the proposed algorithm is slightly better rate-distortion than the original rate control average is 0.05% rate control accuracy.

2.6.2. Frame Rate Up-Conversion Approach

Besides the rate control approach, the frame rate up-conversion approach is also proposed to picture quality than the reference better [41]. A novel integration of frame rate up-conversion and HEVC coding based on rate-distortion optimization is proposed. The author uses the IBBBP coding structure in GOP, which is different from those encoding rate control. The core idea in this framework is to interpolate the frame into the original frame following joint motion estimation algorithms. The whole

framework can be illustrated in Figure 2.23. The algorithm can achieve about 0.48 dB picture quality improvement by analyzing the performance based on the QP is fixed into four values (22, 27, 32, and 37). However, increasing the frame rate is not a better solution for some applications. It can lead to a bit over-head occurred.

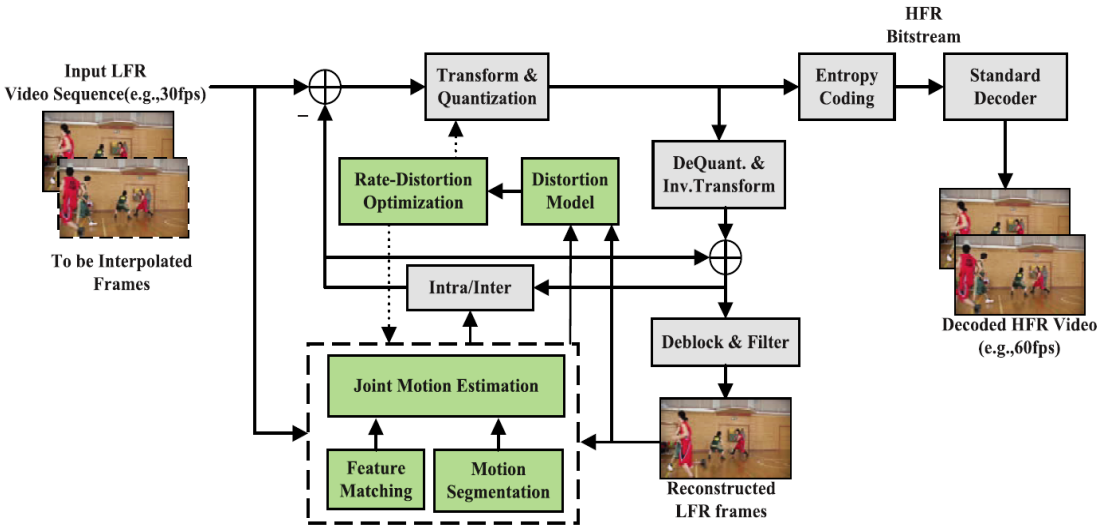


Figure 2.23 Frame Rate Up-Conversion Approach

2.6.3. Decoder Convolution Neural Network Approach

The last approach is the decoder convolution neural network, representing the CNN approach applying in the decoder side of the video coding standard. The CNN-based in-loop filtering for coding efficiency improvement is proposed in [44]. Figure 2.24 shows the entire framework of the proposed framework. The author replaced the SAO filtering in in-loop filtering with the learnable CNN network to enhance the reconstructed picture in both encoder and decoder. The proposed framework CNN has used the architecture of VDSR as the pre-trained network in reference software HEVC. The proposed algorithm can get slightly better picture quality than reference software on average 0.05 dB.

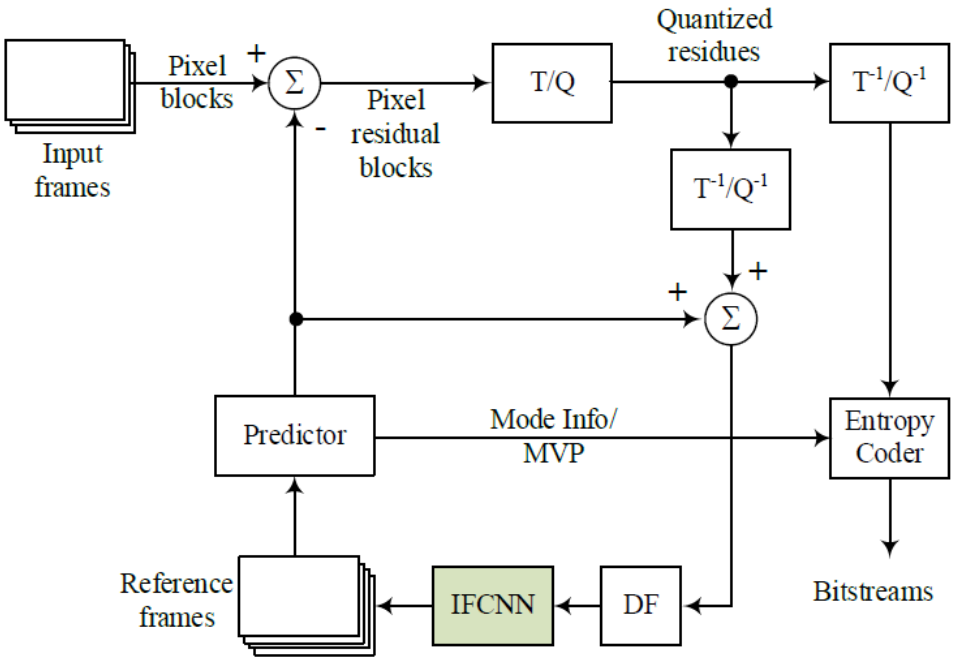


Figure 2.24 IFCNN Framework in In-Loop Filtering [44]

Another CNN-based approach for visual quality improvement on HEVC is proposed in [45]. The author proposed deep CNN in only the decoder side, as shown in Figure 2.25. The CNN model is applied after finishing the adaptive loop filter to reduce the blocking artifacts and also the discontinuities in the frame. The proposed can achieve about 0.07 to 0.24 dB picture quality improvement than the original reference software HEVC.

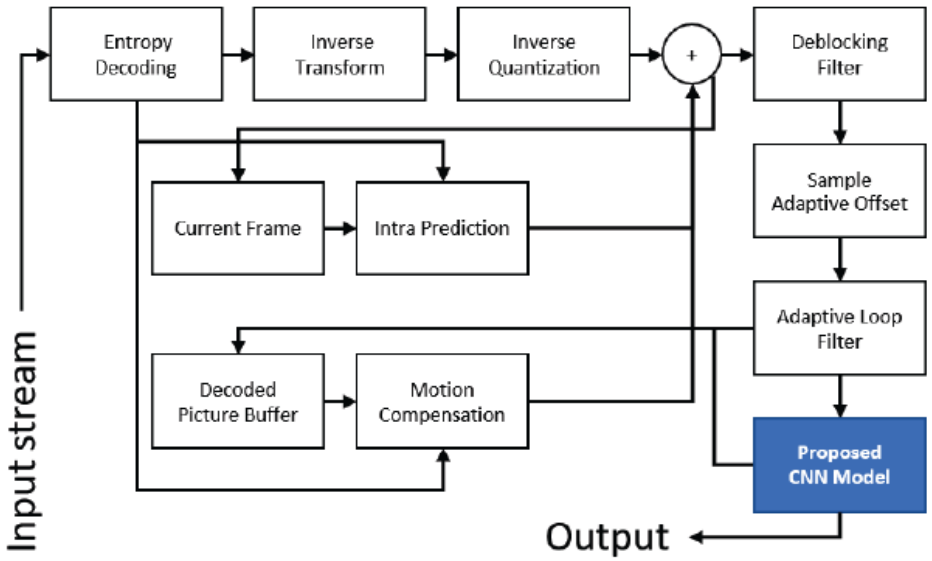


Figure 2.25 Deep CNN-based Approach on HEVC decoder side [45]

Ren Yang proposes a learnable CNN on compressed video in [46]. The author named the algorithm as a multi-frame quality enhancement (MFQE) illustrated in Figure 2.26. The first procedure of the proposed framework is to search the frame which has the highest picture quality in total compressed videos. Then, multi-frame CNN is assigned to enhance the non-peak quality frame or low-quality frame to adapt to the high-quality frame. As a result, the framework can increase picture quality by about 0.45 dB on average comparing to reference software.

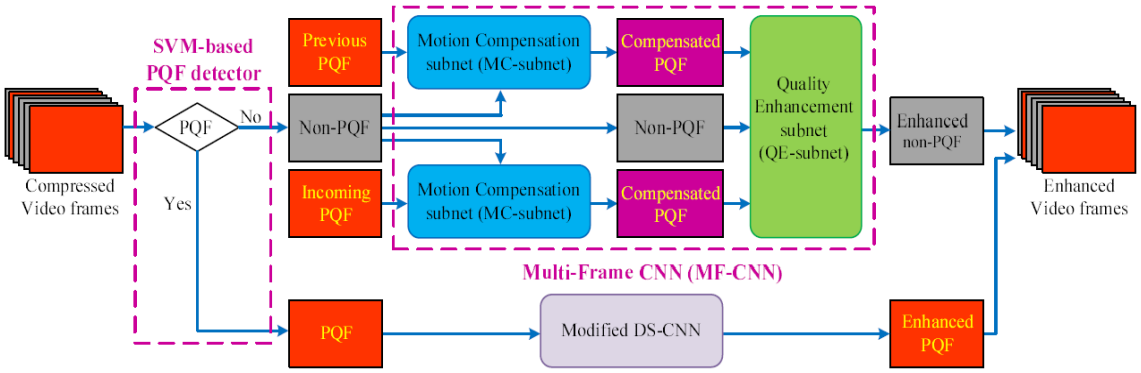


Figure 2.26 Multi-Frame Quality Enhancement for Compressed Video Framework [46]

CNN on the decoder side can help improve picture quality, but it is not a compactable standard. In this work, the learning-based approach is proposed in only the encoder side for visual quality enhancement on HEVC.

CHAPTER 3
METHODOLOGY

This chapter is separated into three main parts. Firstly, the overall block digram of the proposed framework is described. Then, the correlation between rate control and neuron network is explained. The last part presents the detail of the proposed method.

3.1.System Overview

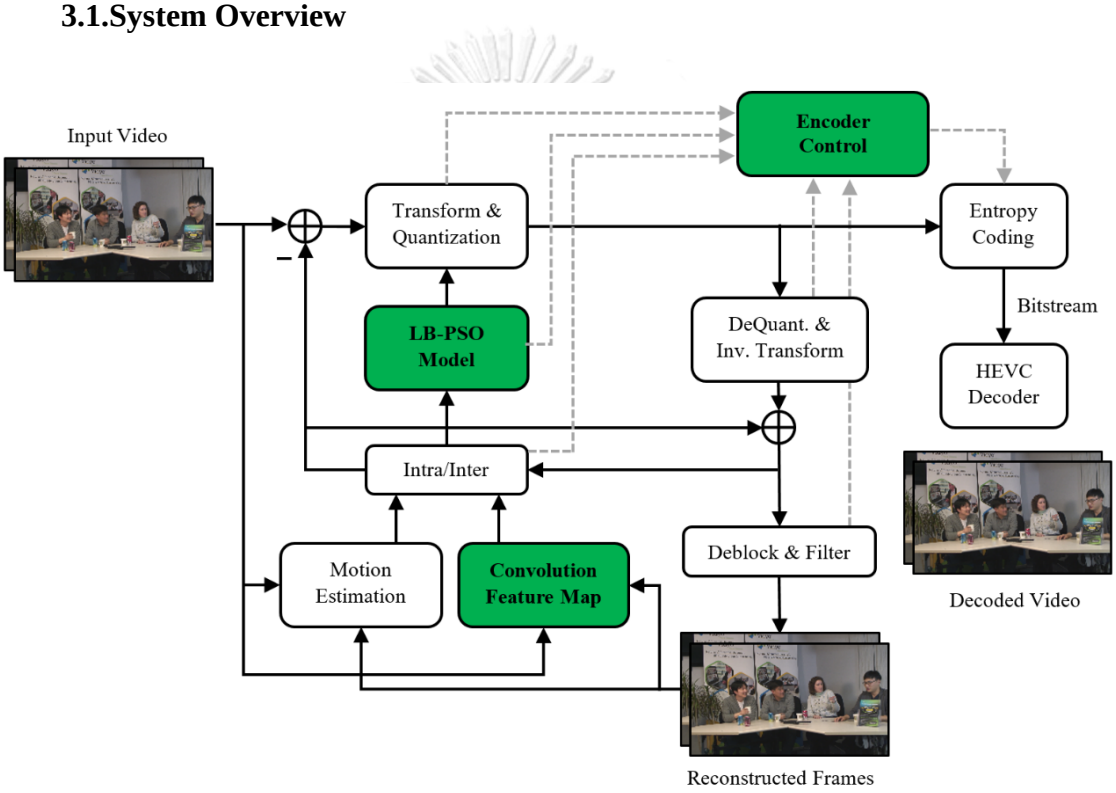


Figure 3.1 Learning-Based Rate Control Diagram for High Efficiency Video Coding

The proposed framework mainly focuses on the adaptive rate control associated with the video content to improve the compressed video quality and maintain the bit budgets at the encoder side only, as shown in Figure 3.1. Precisely, the green boxes represent the modification rate control model using the feature translation technique. First, the input video is fed into the convolution feature map to extract the high dimensional feature space, which contains essential features representing the object in the scene. Then, the proposed model is learned to translate

the input feature space to rate control parameters to get the optimal trade between target bit rate and distortion rate. The following section presents the correlation between rate control and neuron network.

3.2.Rate Control and Neuron Network Correlation

The hyperbolic R-D model is performed in HEVC, where the computation of λ related to bit rate R can re-formulate as the neural network function. Generally, the neural architecture is constructed by applying weight sum with a bias and then pass through the activation function to activate or deactivate the neurons. Figure *3.2* shows the general architecture of the neural network, where $x_0, x_1, x_2, \ldots, x_m$ are the inputs, $w_0, w_1, w_2, \ldots, w_m$ are the learnable weights, and b represents as a bias.

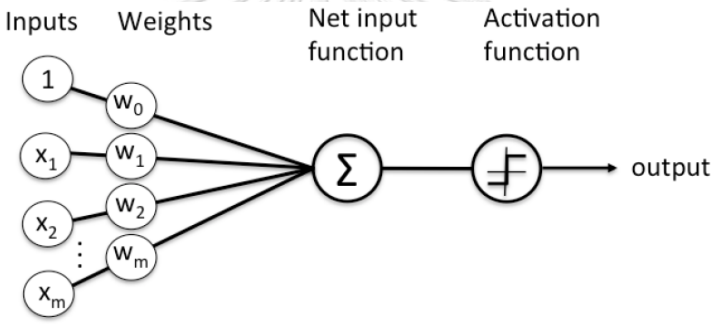


Figure 3.2 Neural Network Architecture

The score function is defined as in Eq. (3.1),

$$f(x_i, W, b) = W \cdot x_i + b \tag{3.1}$$

Or

$$f(x_i, W) = W \cdot x_i$$

And the loss function L can be calculated as in Eq. (3.2),

$$L = \underbrace{\frac{1}{m}\sum_i L_i}_{\text{Data Loss}} + \underbrace{\lambda R(W)}_{\text{Regularization Loss}} \tag{3.2}$$

where $R(W)$ represents the regularization loss, it uses to prevent the overfit data training.

In HEVC, the Lagrange multiplier λ is computed by knowing the input bit rate R as in Eq. (3.3),

$$\lambda = \alpha \cdot R^{\beta} \tag{3.3}$$

$$\ln(\lambda) = \ln\left(\alpha \cdot R^{\beta}\right)$$

$$\ln(\lambda) = \ln\left(R^{\beta}\right) + \ln(\alpha)$$

$$\ln(\lambda) = \beta \cdot \ln(R) + \ln(\alpha)$$

$$\equiv f(x, W, b) = W \cdot x + b$$

Hence, the neural network can solidify the Hyperbolic R-D model as a learnable weight to adapt to the video content.

3.3. Learning-Based Rate Control

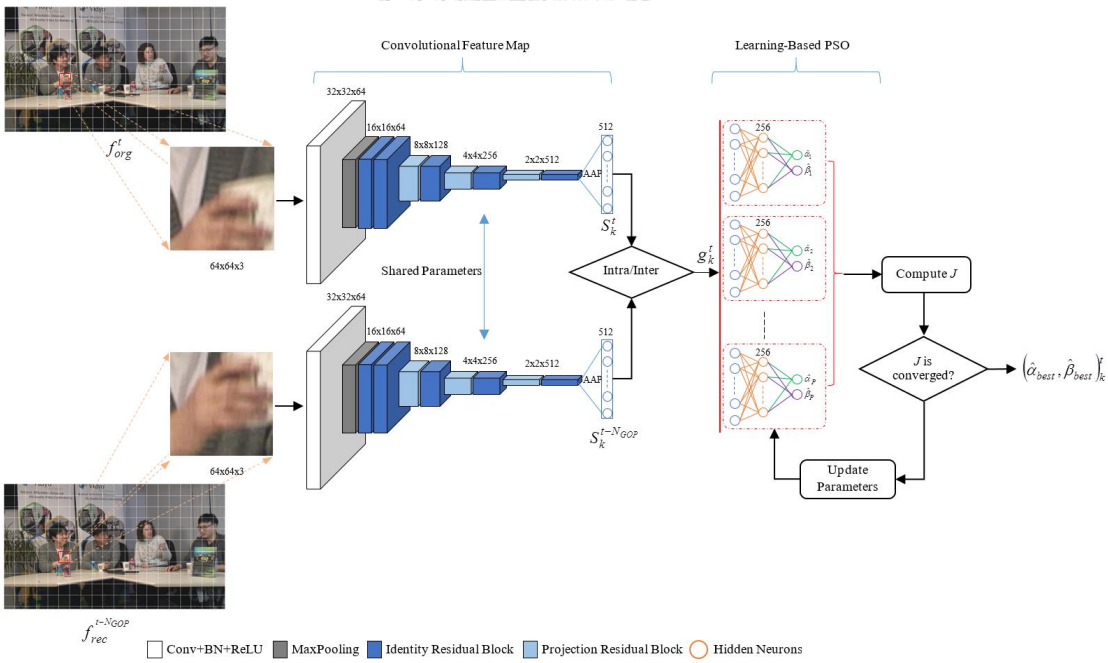


Figure 3.3 Overview of proposed learning-based particle swarm optimization

This section introduces a learning-based rate control algorithm, which creates a regression map for the R-λ parameter. The proposed architecture consists of two main modules, including the convolution feature map and the regression map representations for R-λ parameters, as shown in Figure 3.3. The regression map is designed as learning-based particle swarm optimization (LB-PSO). Besides, the

parameter updating for Inter-coding is performed by taking residue information into account. The details of each part are presented in the following subsections.

3.3.1. Convolutional Feature Map

The convolutional feature map (Fully Convolutional Networks - FCNs) is introduced at the first stage to obtain the meaningful spatial representation of CTUs pictures for the input of our LB-PSO model. In general, the early layers of convolutions in the deep convolutional networks demonstrate the local or low-level feature information of the input image. In contrast, the deeper layers of convolutions indicate the high-level feature information that provides more global image information [48]. Additionally, the last fully connected (FC) layer of deep nets is designed to define the high-level feature information into object classes. Since FCNs do not include the FC layer, a relationship between the input image and the final feature output layer is preserved and expressed as data compression, which encodes the raw-pixel representation of the input image to high-level information. This information provides the global feature *G* representing the input image characteristic. *G* is fed into our LB-PSO model to generate the *R*-λ parameters. Precisely, a pre-trained residual networks (ResNets) model [49] on the ImageNet dataset [50] is used in this work without the FC layer to extract the powerful convolutional feature. However, the original input size of ResNets is incompatible with the maximum size of CTUs. The adaptive average pooling (AAP) is then applied to the last convolution layers to ensure the compatibility of input and output dimensions. Figure 3.3 demonstrates the overall layout of our convolutional feature map architecture.

Suppose a t^{th} frame contains a total *K* CTUs, then $G^t = \{g_0, g_1, \cdots, g_K\}_t$. In order to obtain *G* for re-feedback coding of each coding structure in HEVC, i.e., Intra or Inter pictures, we define *G* as in Eq.(3.4),

$$g_k^t = \begin{cases} S_k^t & \text{if } IntraPicture, \\ \left| S_k^t - S_k^{t-N_{GOP}} \right|_{(t \bmod c)} & otherwise \end{cases} \tag{3.4}$$

, where $k \in K$, and $c\,(c > 0)$ is a constant to determine the frame index for re-feedback coding on $(t \bmod c)$. N_{GOP} is total number of pictures in a GOP. S_k^t and $S_k^{t-N_{GOP}}$ represent the convolutional feature information (spatial representation) of k^{th} CTU getting from the original frame f_{org} at *t* position and reconstruction frame f_{rec} at $t - N_{GOP}$ position, respectively.

Specifically, if the encoding mode is Intra-coding, the spatial representation is directly inputted to the LB-PSO model. Otherwise, we compute the semantic residue information by applying the absolute difference between the current spatial representation S_k^t of the original CTU and the previous spatial representation $S_k^{t-N_{GOP}}$ of the reconstructed CTU before feeding it to the LB-PSO model.

3.3.2. Learning-Based Particle Swarm Optimization Network

a) LB-PSO Estimator

Our LB-PSO is proposed to define the optimal mapping ϕ from the spatial-temporal representation of CTU g_k to rate control parameters y_k, $y_k = \{\alpha, \beta\}_k$. We introduce a feedforward network with one hidden layer to determine y_k. This feedforward network can be computed as in Eq. (3.5).

$$y_k = \phi(h_k; W_\phi, b_\phi) = W_\phi^T h_k + b_\phi \tag{3.5}$$

, where W_ϕ provides the weights of a mapping function ϕ, b_ϕ is a bias, and h_k represents the output of the hidden layer. Precisely, h_k is designed by applying a rectified linear activation function to the output of a linear transformation composed of the weights W_h and bias b_h parameters to trigger a non-linear transformation. Thus, h_k can be derived as in Eq. (3.6).

$$h_k = \max\{0, W_h^T g_k + b_h\} \tag{3.6}$$

From Eq. (3.5) and Eq. (3.6), our complete mapping model can be re-formulated as in Eq. (3.37).

$$y_k = W_\phi^T \max\left\{0, W_h^T g_k + b_h\right\} + b_\phi \tag{3.7}$$

The model parameters $M = \left\{W_\phi, W_h, b_\phi, b_h\right\}$ are optimized by utilizing swarm intelligence in order to exchange the information between particles with regard to R-D cost function, *J*. On the other hand, the model parameters regulate its trajectory concerning its best previous position and the best previous position reached by any member of its neighborhood. The cost function *J* is determined by two objective functions, including a reconstruction error (MSE) of visual quality and smooth$_{L1}$ error of bit allocation, to target the swarm intelligence rule. The cost function *J* can be defined as in Eq. (3.8) and Eq. (3.9).

$$J = \frac{1}{N}\sum_{j=0}^{N-1}\left(f_{org_j} - f_{rec_j}\right)^2 + \eta \text{smooth}_{L1}\left(R_T - R_A\right) \tag{3.8}$$

$$\text{smooth}_{L1}(U) = \begin{cases} \dfrac{U^2}{2} & if\ |U| < 1, \\ |U| - \dfrac{1}{2} & otherwise \end{cases} \tag{3.9}$$

, where *N* is the total number of pixels in a picture and η is a penalty coefficient. R_T and R_A are the target bit and actual bit on picture level, respectively.

According to the cost function design, the model parameters are updated after all CTUs are fully encoded. This cost function aims to model learning to achieve the trade-off between distortion and bit allocation. The following section introduces the complete process of parameters update.

b) Parameters Updating

In this subsection, we present the parameter update of the encoder controller corresponding to the Intra/Inter coding mode. In addition, the Inter coding mode is classified into two sets of coding frames, such as a core frame and a common frame.

A core frame is encoded by activating the re-feedback coding to adjust the bit budget at the CTU coding level. In contrast, the common frame is coded by applying the default Lagrangian multiplier to determine the bit budget at the CTU coding level. For both Intra coding and core frame of Inter coding, the bit budget at the CTU coding level is computed by using Eq. (2.10) and Eq. (3.7). Additionally, the model parameters M in equation Eq. (3.7) individually parameterize its value according to its movement in a search space.

Let P is the total size of the population, V_i is the velocity (position change) of i^{th} particle, B_i is the best previous model parameters of i^{th} particle, and B_g is the best model parameters in the swarm. Then the swarm is manipulated on each iteration n according to the following two equations,

$$V_i^{n+1} = aV_i^n + c_1 r_{i1}^n \left(B_i^n - M_i^n\right) + c_2 r_{i2}^n \left(B_g^n - M_i^n\right), \tag{3.10}$$

$$M_i^{n+1} = M_i^n + V_i^{n+1} \tag{3.11}$$

, where $i = 1, 2, ..., P$ and a is the inertia weight of velocity V, which is used to control the trade-off between the global and the local exploration capabilities of the swarm. c_1 and c_2 are two positive acceleration constants, named the cognitive and social parameters of PSO, respectively. r_{i1} and r_{i2} are random numbers generated from a uniform distribution within the range [0, 1]. The performance of each model parameters M_i in the swarm is measured according to the cost function J. The lower cost function indicates a better M_i. After finalizing the best M_i to preserve the minimal cost function J at CTU coding level, the CTU is encoded.

For the picture level of Inter coding, the rate control parameters are adjusted by considering the residue score of the semantic residue information. The probability of residue score Q^t on a picture at time t can be computed as in Eq. (3.12) and Eq. (3.14).

$$Q^t = \sum_{k \in K} \sum_{j \in S_k} \frac{A_k^t(j)}{S_k^t(j)} \tag{3.12}$$

$$A_k^t(j) = \begin{cases} 0 & if\ t - N_{GOP} < 0, \\ \left| S_k^t(j) - S_k^{t \times \left\lfloor \frac{t}{N_{GOP}} \right\rfloor}(j) \right| & otherwise \end{cases} \quad (3.13)$$

, where $\lfloor . \rfloor$ represents the rounded result. Additionally, in the *GOP* regarding the Spatio-temporal information of the video sequence, the picture levels generally have different pairs of encoder controller coefficients α_p and β_p. Therefore, the rate control parameters can be updated by Eq. (3.14) to Eq. (3.17).

If the GOP_{id} equals 0, a pair of rate control parameters can be formulated in Eq. (3.14) to Eq. (3.15).

$$\alpha_{pnew} = \alpha_{pold} + \delta_\alpha \cdot (\ln(\lambda_r - \lambda_c)) \cdot \alpha_{pold} + \zeta Q^t \quad (3.14)$$

$$\beta_{pnew} = \beta_{pold} + \delta_\beta \cdot (\ln(\lambda_r - \lambda_c)) \cdot \ln(bpp_r) + \frac{\zeta}{2} Q^t \quad (3.15)$$

Otherwise, a pair of rate control parameters can be computed as Eq. (3.16) to Eq. (3.17).

$$\alpha_{pnew} = \alpha_{pold} + \zeta Q^t \quad (3.16)$$

$$\beta_{pnew} = \beta_{pold} + \frac{\zeta}{2} Q^t \quad (3.17)$$

, where δ_α and δ_β are the default constant in HEVC reference software. λ_r represents as real λ value, λ_c is a computed λ value from real cost bpp_r with the previous rate control parameters α_{pold} and β_{pold} at picture level and ζ is residue penalty constant.

CHAPTER 4
EXPERIMENTAL RESULTS

To evaluate the performance of the proposed learning-based particle swarm optimization, the experiments are conducted on various videos, including static and dynamic scenes. The experiment setting is presented in Section 4.1, and the experimental results and analysis are described in Section 4.2

4.1. Experiment Setting

4.1.1. Test Sequences and Parameter Setting

In the experiment, the proposed algorithm is implemented on HEVC reference software [51] and is compared with the PS-GOP [40] and the state-of-the-art *R-λ* rate control (RC-HEVC) [18]. The proposed algorithm and baseline methods are simulated in the same reference software HM-16.10. Precisely, the experiments are conducted under the low-delay P main profile configurations, and the encoder parameters are set according to the standard-setting in [47] by enabling the Rate Control as *True*. There are thirteen test video sequences with four video resolutions, as shown in Figure 4.1. They are two videos of 240p (Wide Quarter Video Graphics Array - WQVGA), three videos of 480p (Wide Video Graphics Array - WVGA), five videos of 720p (HD), and three videos of 1080p (Full HD). Table 4.1 briefly summarizes the characteristics of the test video sequence. In addition, the test video sequence is encoded at four different target bit rates corresponding to the video resolution.

Since the goal of rate control is not only to improve the visual quality of the video for a given bit rate but also to achieve the bit rate closest to the target bit rate, so both Peak Signal-to-Noise Ratio (PSNR) and bit rate error (BRE) are used as the criteria for determining the performance of rate control algorithm.

(a) ParkScene

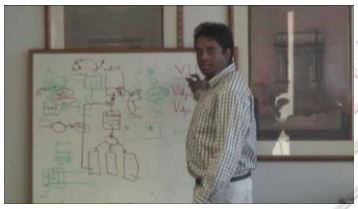
(b) Cactus

(c) BQTerrace

(d) FourPeople

(e) KristenAndSara

(f) Vidyo1



(g) Vidyo3

(h) Vidyo4

(i) BasketballDrillText

(j) PartyScene

(l) BQMall

(m) BlowingBubbles

(n) BQSquare

Figure 4.1 Test Sequence Videos Dataset

Table 4.1 Video Sequence Detail

Resolution	Name of Video Sequence	Frame Rate (fps)	Bit Rate (kbps)
1920 x 1080	ParkScene Cactus BQTerrace	24 50 60	1000, 2000, 3000, 4000
1280 x 720	FourPeople KristenAndSara Vidyo1 Vidyo3 Vidyo4	60 60 60 60 60	384, 512, 850, 1200
832 x 480	BasketballDrillText PartyScene BQMall	50 50 60	384, 512, 768, 1200
416 x 240	BlowingBubbles BQSquare	50 60	256, 384, 512, 1200

4.1.2. Peak Signal to Noise Ratio

The quality of the reconstructed image or video comparing with raw image or video is computed based on Peak signal-to-noise ratio (PSNR) measurement. Defining PSNR has a close relationship between mean square errors (MSE) where PSNR can be computed as Eq. (4.1).

$$PSNR = 10\log\left(\frac{\left(2^n - 1\right)^2}{\sqrt{MSE}}\right) \tag{4.1}$$

, where

$$MSE = \frac{1}{N}\sum_{j=0}^{N-1}\left(f_{org_j} - f_{rec_j}\right)^2 .$$

4.1.3. Bit Rate Error

BRE is used to determine the accurate bit consumption of the proposed method to what the target bit is assigned. BRE can be computed as Eq. (4.2).

$$BRE = \left(\frac{R_T - R_A}{R_T} \right) \times 100\% \tag{4.2}$$

4.2. Experimental Results and Analysis

4.2.1. Rate-Distortion Performance and Bit Rate Accuracy

The first experiment is conducted on the low video resolution (WQVGA), which contains two video sequences with different frame rates, including BlowingBubbles and BQSquare. These two videos have various dynamic characteristics, such as a moving camera, moving objects, and illumination changes. Table 4.2 describes the PSNR and BRE performance of the proposed method compared with the baseline methods. It is clearly shown that our learning-based method outperforms all the baseline methods as we achieve the highest PSNR value with the same bit rate. Specifically, the average PSNR enhancement of our method is 0.23 dB and 0.12 dB compared with RC-HEVC and PS-GOP, respectively. Our approach also performs the maximum PSNR improvement (max) of 0.30 dB and 0.20 dB compared to RC-HEVC and PS-GOP. Figure 4.2(a) illustrates the R-D curve performance of the BQSquare test sequence. The learning-based approach obtains better R-D performance than that of the baselines method. In addition, the average BRE of RC-HEVC, PS-GOP, and our methods are 0.01%, indicating that all approaches can effectively achieve the target bit rate. However, the proposed method has the lowest BRE at a lower target bit rate (256kbps). It is noticed that the RC-HEVC has a poor visual quality on these WQVGA with dynamic scenes compared to all approaches. As a result, even if the scene has dynamic properties, our algorithm can constructively achieve the target bit rate with the good visual quality of the WQVGA sequence.

Table 4.2 The Performance of PSNR and BRE of Video Sequence with Resolution of 416x240

Name of Video Sequence	Target Bit Rate	RC-HEVC			PS-GOP			Proposed Method		
		Bit Rate	PSNR	BRE	Bit Rate	PSNR	BRE	Bit Rate	PSNR	BRE
BlowingBubbles	256	256.06	29.69	-0.02	256.08	29.79	-0.03	256.02	29.99	-0.01
	384	384.05	31.14	-0.01	384.00	31.26	0.00	384.02	31.44	-0.01
	512	512.06	32.26	-0.01	512.05	32.38	-0.01	512.04	32.51	-0.01
	1200	1200.18	35.64	-0.02	1200.05	35.71	0.00	1200.15	35.73	-0.01
BQSquare	256	256.04	30.31	-0.02	256.01	30.42	-0.01	256.02	30.60	-0.01
	384	384.03	31.53	-0.01	384.03	31.67	-0.01	384.03	31.78	-0.01
	512	512.03	32.45	-0.01	512.03	32.56	-0.01	512.02	32.64	0.00
	1200	1200.06	35.20	0.00	1200.04	35.33	0.00	1200.04	35.37	0.00
Average			32.28	-0.01		32.39	-0.01		**32.51**	-0.01

Next, the WVGA sequences are tested, such as BasketballDrillText, PartyScene, and BQMall. The scene properties are similar to the above experiments, but these WVGA sequences are more challenging than WQVGA because they involve multi-object movement, camera movement, and higher resolution. The outcomes of PSNR and BRE are summarized in Table 3, where the proposed learning-based method works much better. It reaches 0.41 dB and 0.33 dB of visual quality better than RC-HEVC and PS-GOP, respectively. Concisely, our approach has no error bit consumption on average and performs 0.23 dB and 0.16 dB on average higher than RC-HEVC and PS-GOP, respectively. Our proposed method is significantly higher on one side of the R-D curve than the competitive methods, as shown in Figure 4.2(b). Based on the outcomes of all approaches in Table 4.2 and Table 3, the R-λ rate control and PS-GOP are not suitable for such dynamic scenes and cameras. Consequently, it can indicate that the λ adjustment and quality control are not correctly estimated.

Table 4.3 The Performance of PSNR and BRE of Video Sequence with Resolution of 832x480

Name of Video Sequence	Target Bit Rate	RC-HEVC			PS-GOP			Proposed Method		
		Bit Rate	PSNR	BRE	Bit Rate	PSNR	BRE	Bit Rate	PSNR	BRE
BasketballDrill Text	384	384.03	30.82	-0.01	383.99	30.93	0.00	384.02	30.99	-0.01
	512	512.05	31.94	-0.01	512.00	32.01	0.00	511.99	32.08	0.00
	768	768.04	33.46	-0.01	768.04	33.52	-0.01	768.05	33.60	-0.01
	1200	1200.10	35.15	-0.01	1200.07	35.20	-0.01	1200.07	35.32	-0.01
PartyScene	384	384.01	26.40	0.00	384.00	26.49	0.00	383.97	26.80	0.01
	512	512.02	27.27	0.00	512.01	27.37	0.00	511.96	27.68	0.01
	768	768.09	28.61	-0.01	768.02	28.68	0.00	768.02	29.01	0.00
	1200	1200.06	30.15	-0.01	1200.02	30.20	0.00	1200.03	30.53	0.00
BQMall	384	384.01	30.68	0.00	384.13	30.77	-0.03	384.00	30.85	0.00
	512	512.01	31.86	0.00	512.05	31.92	-0.01	512.03	32.00	-0.01
	768	768.01	33.50	0.00	768.01	33.59	0.00	768.01	33.66	0.00
	1200	1200.04	35.28	0.00	1200.03	35.33	0.00	1200.01	35.39	0.00
Average			31.26	-0.01		31.33	-0.01		**31.49**	0.00

After testing the WVGA sequences, the HD videos containing video conferencing and online teaching test sequences are simulated. The HD videos are FourPeople, KristenAndSara, Vidyo1, Vidyo3, and Vidyo4. These videos have the characteristics of a static camera with multiple objects moving. Figure 4.2 shows an overall outgrowth of the R-D curve of FourPeople from the low bit rate to the high bit rate. Although the scene is used with a static camera, the R-D performance of the proposed method is noticeably more significant than the competitive methods. Additionally, the PSNR and BRE evaluations of these HD video sequences are recorded in Table 4.4. The average PSNR enhancement value of our method is

approximately 0.17 dB (max = 0.30 dB) and 0.08 dB (max = 0.21 dB) in comparison with the RC-HEVC and PS-GOP.

Table 4.4 The Performance of PSNR and BRE of Video Sequence with Resolution of 1280x720

Name of Video Sequence	**Target Bit Rate**	**RC-HEVC**			**PS-GOP**			**Proposed Method**		
		Bit Rate	**PSNR**	**BRE**	**Bit Rate**	**PSNR**	**BRE**	**Bit Rate**	**PSNR**	**BRE**
FourPeople	384	383.97	37.02	0.01	383.99	37.12	0.00	383.99	37.32	0.00
	512	511.97	38.10	0.01	512.00	38.24	0.00	511.99	38.38	0.00
	850	849.98	39.84	0.00	849.99	39.94	0.00	849.98	40.06	0.00
	1200	1200.08	40.81	-0.01	1199.96	40.87	0.00	1200.05	40.97	0.00
KristenAndSara	384	384.06	39.17	-0.02	384.08	39.32	-0.02	384.12	39.37	-0.03
	512	512.07	40.03	-0.01	512.09	40.17	-0.02	512.11	40.20	-0.02
	850	850.12	41.31	-0.01	850.09	41.43	-0.01	850.12	41.47	-0.01
	1200	1200.18	42.04	-0.01	1200.16	42.12	-0.01	1200.16	42.16	-0.01
Vidyo1	384	384.00	38.95	0.00	383.98	39.06	0.01	384.00	39.11	0.00
	512	512.01	39.86	0.00	511.93	39.95	0.01	511.99	40.01	0.00
	850	849.96	41.19	0.00	849.88	41.26	0.01	850.01	41.32	0.00
	1200	1200.00	41.93	0.00	1199.96	42.00	0.00	1200.01	42.07	0.00
Vidyo3	384	384.01	37.85	0.00	384.00	38.00	0.00	384.02	38.01	-0.01
	512	512.02	38.82	0.00	512.01	38.95	0.00	512.01	38.97	0.00
	850	850.01	40.22	0.00	850.01	40.33	0.00	850.01	40.37	0.00
	1200	1200.02	41.00	0.00	1200.03	41.08	0.00	1200.00	41.12	0.00
Vidyo4	384	384.01	38.68	0.00	384.01	38.73	0.00	384.01	38.86	0.00
	512	512.02	39.47	0.00	512.01	39.53	0.00	512.02	39.67	0.00
	850	850.02	40.67	0.00	850.01	40.74	0.00	850.02	40.86	0.00
	1200	1200.02	41.39	0.00	1200.05	41.45	0.00	1200.02	41.54	0.00
Average			39.92	0.00		40.02	0.00		**40.09**	0.00

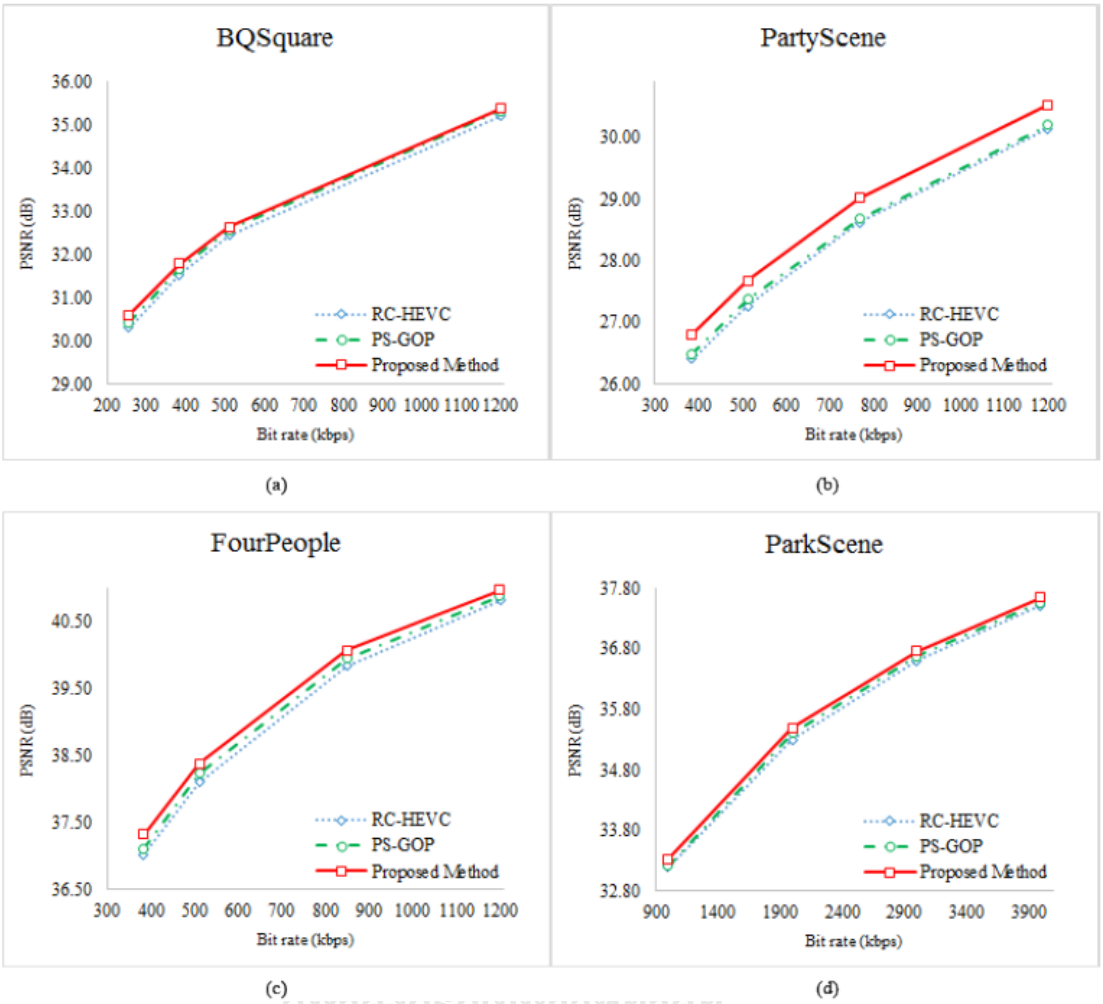


Figure 4.2 Rate-Distortion curves: (a) BQSquare, (b) PartyScene, (c) FourPeople, (d) ParkScene

The last experiment is applied on full HD video test sequences, including, ParkScene, Cactus, and BQTerrace. This last test contains all types of scenarios. ParkScene video has a moving camera and multiple object motions, while BQTerrace video stacks the camera motion with a static camera. Besides, Cactus video consists of a static camera and the rotation of the objects. According to Table 4.5, the overall PSNR evaluation of the proposed method on the BQTerrace sequence at a low bit rate is the highest compared to the others sequences. In contrast, the ParkScene sequence has the highest PSNR at a high bit rate. The reason is that the scenes containing a dynamic camera have large movement changes; thus, the state-of-the-art R-λ rate

control cannot update the encoding controller correctly. In addition, PS-GOP uses parameter sharing in GOP, which is not enough to adapt to encoder parameters following frame characteristics. Reasoning from this fact, our method establishes a novel mapping between frame features and R-λ coefficient parameters. We provide a computationally feasible solution using LB-PSO to optimal R-D for good visual quality and maintain the target bit rate. Figure 4.2 shows the overall R-D curve on different video resolutions. Consequently, our method has achieved the highest outcomes of all competitive methods. From Table 4.2 to Table 4.5, the average PSNR improvement is 0.19 dB (max = 0.41 dB) and 0.10 dB (max = 0.33 dB) compared with RC-HEVC and PS-GOP, respectively.

Table 4.5 The Performance of PSNR and BRE of Video Sequence with Resolution of 1920x1080

Name of Video Sequence	**Target Bit Rate**	**RC-HEVC**			**PS-GOP**			**Proposed Method**		
		Bit Rate	**PSNR**	**BRE**	**Bit Rate**	**PSNR**	**BRE**	**Bit Rate**	**PSNR**	**BRE**
ParkScene	1000	999.96	33.20	0.00	999.84	33.21	0.02	999.86	33.32	0.01
	2000	2000.01	35.30	0.00	1999.89	35.41	0.01	2000.10	35.49	0.00
	3000	2999.95	36.60	0.00	2999.91	36.68	0.00	2999.98	36.76	0.00
	4000	4000.11	37.52	0.00	4000.09	37.57	0.00	4000.11	37.66	0.00
Cactus	1000	1000.01	31.62	0.00	1000.02	31.74	0.00	1000.02	31.75	0.00
	2000	2000.04	33.77	0.00	2000.03	33.85	0.00	2000.03	33.87	0.00
	3000	3000.09	34.96	0.00	3000.03	35.01	0.00	3000.03	35.04	0.00
	4000	4000.06	35.70	0.00	3999.95	35.77	0.00	4000.07	35.81	0.00
BQTerrace	1000	1000.05	31.62	-0.01	1000.01	31.73	0.00	1000.17	31.97	-0.02
	2000	2000.13	33.03	-0.01	2000.02	33.11	0.00	2000.04	33.25	0.00
	3000	3000.15	33.67	0.00	3000.01	33.78	0.00	3000.08	33.82	0.00
	4000	4000.53	34.10	-0.01	4000.05	34.20	0.00	4000.11	34.15	0.00
Average			34.26	0.00		34.34	0.00		**34.41**	0.00

4.2.2. Bit Heatmaps and Visual Quality

To indicate the performance of bit allocation at the CTU level, the heatmap visualization and the subjective of the reconstructed frame are illustrated in Figure 4.3 and Figure 4.4. Since there is no modification on intra coding of PS-GOP, Figure 4.3 shows only the comparison between state-of-the-art RC-HEVC with our proposed learning-based approach. The bit consumption is highlighted by red color intensity on each CTU, while the blue act as a mask to cover the frame. If the red intensity is low, it means that the allocated bits are consumed less. The patch image is extracted from the frame to clearly illustrate the most different bit consumption at the CTU level of RC-HEVC and our proposed method. Figure 4.3(b) and Figure 4.3(c) reveal that the bit allocation performance of RC-HEVC on the plane space CTU is slightly high, which leads to less bit budget for necessary spatial CTU.

On the contrary, our proposed method is to obtain smoother bit allocation on non-important spatial images (low-frequency components), providing more budget to important CTU features. Additionally, the visualization of the human face of the proposed learning-based approach on the intra-picture shows more details with a smoother look than that of RC-HEVC, as shown in the green box of Figure 4.3(b) and Figure 4.3(c). According to these results, our LB-PSO can obtain better bit allocation by using the information from the mapping encoder control parameters with the input convolution feature map of each spatial CTU instead of the fixed initialization of R-λ rate control.

For inter coding, the PS-GOP is added in comparison. Similarly, the color representation is defined the same as the intra coding. Figure 4.4(b) shows that RC-HEVC has a problem with bit allocation on the essential features in terms of bitmaps. Due to hand movement, RC-HEVC should provide higher bit allocation in these necessary parts; on the contrary, it allocates fewer bits to these blocks. Besides, PS-GOP attempts to allocate the amount of bit budget to the hand movement area to keep the visual quality of the action consistent. However, the bit budget on large hand motion blocks is still small, as shown in Figure 4.4(c). With regard to residual

semantic information, our proposed method can regulate the bit budget correctly responding to the motion in the scene, as illustrated in Figure 4.4(d).

On the other hand, our proposed method obtains the accurate bit allocation of each CTU corresponding to its spatial-temporal characteristics. Furthermore, the visual quality visualization of this hand movement is shown in Figure 4.4(e) to Figure 4.4(g). In particular, RC-HEVC has a considerable distortion in this hand movement area, while PS-GOP is slightly better than RC-HEVC. Although PS-GOP is better than RC-HEVC, PS-GOP still has higher distortion compared with our proposed method. As a result, the proposed method achieves better hand and cup shapes compared to the competitive methods. According to our experimental results, we can conclude that the proposed learning-based R-λ parameter outperforms other competing methods by achieving the highest PSNR with maintaining the target bit rate.

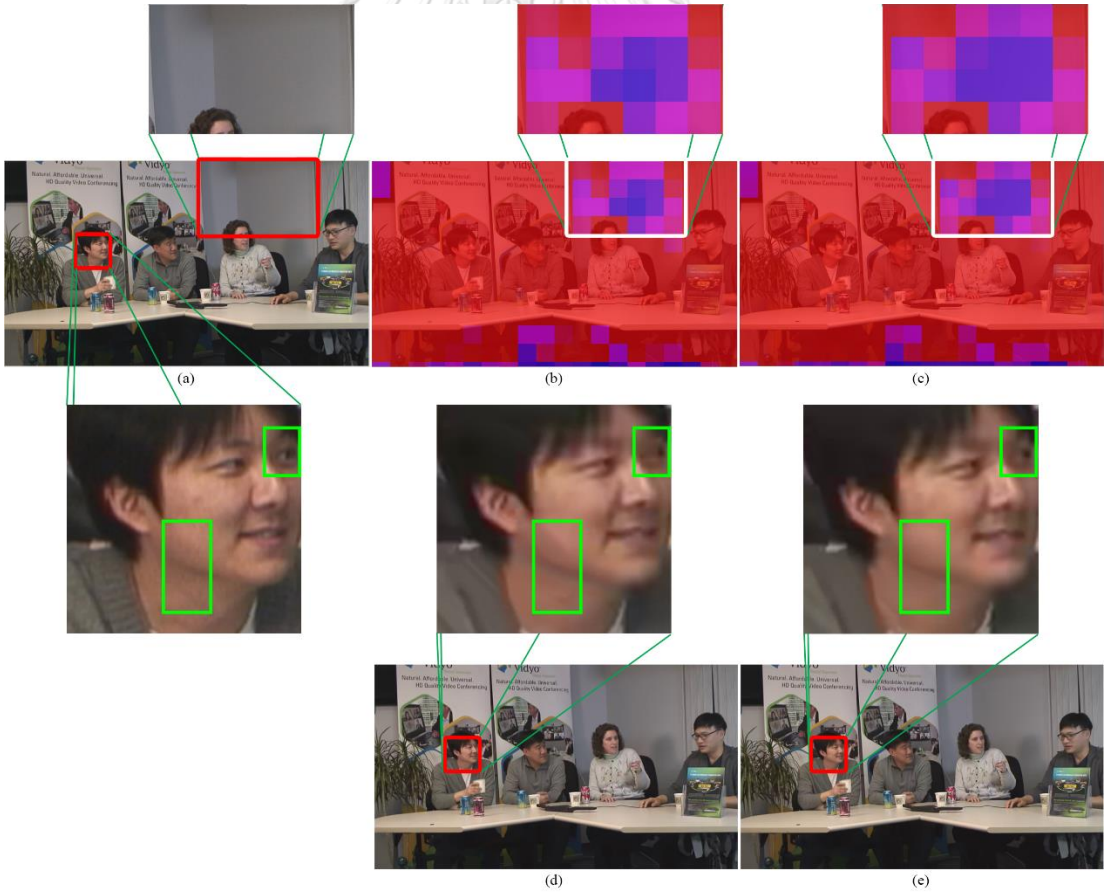


Figure 4.3 Bit Heatmaps and Reconstructed Frame of Intra Coding at 384 kbps: (a) Original Frame, (b)&(d) RC-HEVC, (c)&(e) Proposed Method

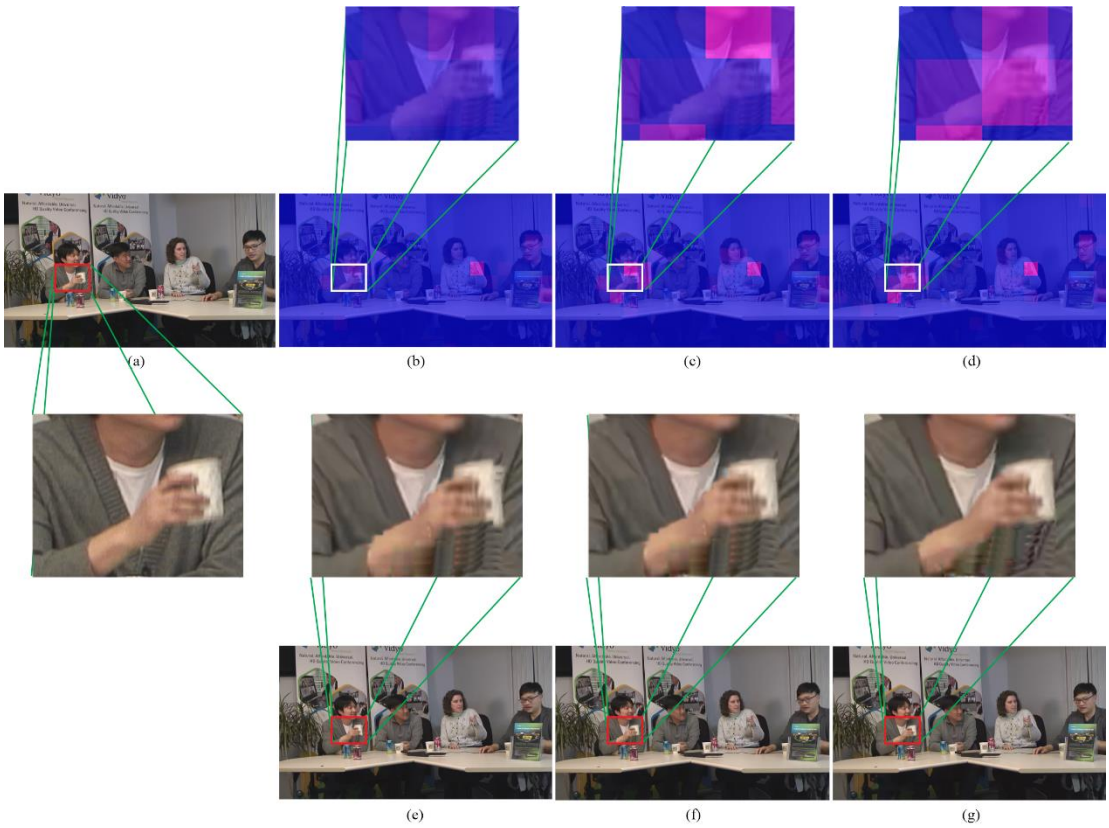


Figure 4.4 Bit Heatmaps and Reconstructed Frame of Inter Coding at 384 kbps: (a) Original Frame, (b)&(e) RC-HEVC, (c)&(f) PS-GOP, (d)&(g) Proposed Method

CHAPTER 5
CONCLUSION

In this work, we proposed novel learning-based R-lambda parameters for HEVC. The proposed framework is embedded with a deep convolution neural network feature map and LB-PSO, which brings advantages to rate control parameters estimation corresponding to spatial-temporal CTU. LB-PSO is designed to obtain the feasible solution of rate control coefficient parameters to optimize the *R-D* relationship. Experimental results clearly show that our proposed learning-based approach obtains an accurate target bit rate with the 0.19 dB on average to 0.41 dB and 0.10 dB on average to 0.33 dB maximum PSNR improvement than the state-of-the-art RC-HEVC and PS-GOP, accordingly. Due to the bit allocation, our algorithm can achieve an operational bit distribution to each CTU on both Intra and inter coding. In other words, our method is effective and robust for determining the bit budget for the CTU of the frame. For future work, CTU partitioning will be considered together with R-lambda parameters to increase coding efficiency.

REFERENCES

[1] Cisco, "Cisco Annual Internet Report (2018–2023) White Paper," 9 March 2020. [Online]. Available: https://www.cisco.com/c/en/us/solutions/collateral/executive-perspectives/annual-internet-report/white-paper-c11-741490.html. [Accessed 13 July 2021].

[2] M. Liou, "Overview of the p× 64 kbit/s video coding standard," *Communications of the ACM*, vol. 34, no. 4, pp. 59-63, 1991.

[3] K. Brandenburg and G. Stoll, "ISO/MPEG-1 audio: A generic standard for coding of high-quality digital audio," *Journal of the Audio Engineering Society*, vol. 42, no. 10, pp. 780-792, 1994.

[4] I. Recommendation, "Generic coding of moving pictures and associated audio information: Video," 1995.

[5] K. Rijkse, "H. 263: Video coding for low-bit-rate communication," *IEEE Communications magazine*, vol. 34, no. 12, pp. 42-45, 1996.

[6] T. Sikora, "The MPEG-4 video standard verification model," *IEEE Transactions on circuits and systems for video technology*, vol. 7, no. 1, pp. 19-31, 1997.

[7] T. Wiegand, G.J. Sullivan, G. Bjontegaard, and A. Luthra, "Overview of the H. 264/AVC video coding standard," *IEEE Transactions on circuits and systems for video technology*, vol. 13, no. 7, pp. 560-576, 2003.

[8] G.J. Sullivan, J.-R. Ohm, W.-J Han, and T. Wiegand, "Overview of the high efficiency video coding (HEVC) standard," *IEEE Transactions on circuits and systems for video technology*, vol. 22, no. 12, pp. 1649-1668, 2012.

[9] G. Correa, P. Assuncao, L. Agostini, and L. A. da Silva Cruz, "Performance and computational complexity assessment of high-efficiency video encoders," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 22, no. 11, pp. 1899-1909, 2012.

[10] D. Grois, D. Marpe, A. Mulayoff, B. Itzhaky, and O. Hadar, "Performance comparison of h. 265/mpeg-hevc, vp9, and h. 264/mpeg-avc encoders," in *2013 Picture Coding Symposium (PCS)*, 2013.

[11] V. Sze, M. Budagavi, and G. J. Sullivan, "High efficiency video coding (HEVC)," in *Integrated circuit and systems, algorithms and architectures*, Springer, 2014, p. 40.

[12] P. A. Chou, T. Lookabaugh, and R. M. Gray, "Optimal pruning with applications to tree-structured source coding and modeling," *IEEE transactions on*

information theory, vol. 35, no. 2, pp. 299-315, 1989.

[13] J. Lainema, F. Bossen, W.-J. Han, J. Min, and K. Ugur, "Intra coding of the HEVC standard," *IEEE transactions on circuits and systems for video technology,* vol. 22, no. 12, pp. 1792-1801, 2012.

[14] H. Lv, R. Wang, X. Xie, H. Jia, and W. Gao, "A comparison of fractional-pel interpolation filters in HEVC and H. 264/AVC," in *2012 Visual Communications and Image Processing*, 2012.

[15] C.-M. F. e. al., "Sample adaptive offset in the HEVC standard," *IEEE Transactions on Circuits and Systems for Video technology,* vol. 22, no. 12, pp. 1755-1764, 2012.

[16] W. Ding and B. Liu, "Rate control of MPEG video coding and recording by rate-quantization modeling," *IEEE transactions on circuits and systems for video technology,* vol. 6, no. 1, pp. 12-20, 1996.

[17] Z. He, Y. K. Kim, and S. K. Mitra, "Low-delay rate control for DCT video coding via Rho-domain source modeling," *IEEE transactions on Circuits and Systems for Video Technology,* vol. 11, no. 8, pp. 928-940, 2001.

[18] B. Li, H. Li, L. Li, and J. Zhang, "Lambda-domain rate control algorithm for High Efficiency Video Coding," *IEEE transactions on Image Processing,* vol. 23, no. 9, pp. 3841-3854, 2014.

[19] T. Takahama and S. Sakai, "Constrained optimization by the ε constrained differential evolution with gradient-based mutation and feasible elites," in *2006 IEEE international conference on evolutionary computation*, 2016.

[20] A. W. Mohamed and H. Z. Sabry, "Constrained optimization based on modified differential evolution algorithm," *Information Sciences,* vol. 194, pp. 171-208, 2012.

[21] D. P. Bertsekas, Constrained optimization and Lagrange multiplier methods, Academic press, 2014.

[22] Y. Choi, G. Boncoraglio, S. Anderson, D. Amsallem, and C. Farhat, "Gradient-based constrained optimization using a database of linear reduced-order models," *Journal of Computational Physics,* vol. 423, p. 109787, 2020.

[23] H.-G. Beyer and H.-P. Schwefel, "Evolution strategies–a comprehensive introduction," *Natural computing,* vol. 1, no. 1, pp. 3-52, 2002.

[24] K. A. Dowsland and J. Thompson, "Simulated annealing," *Handbook of natural computing,* pp. 1623-1655, 2012.

[25] S. Mirjalili, "Genetic algorithm," in *Evolutionary algorithms and neural networks*, Springer, 2019, pp. 43-55.

[26] K. E. Parsopoulos and M. N. Vrahatis, "Recent approaches to global optimization problems through particle swarm optimization," *Natural computing,* vol. 1, no. 2, pp. 235-306, 2002.

[27] K.-L. Du and M. Swamy, "Particle swarm optimization," in *Search and optimization by metaheuristics*, Springer, 2016, pp. 153-173.

[28] B. Tang, Z. Zhu, and J. Luo, "A framework for constrained optimization problems based on a modified particle swarm optimization," *Mathematical Problems in Engineering,* 2016.

[29] K. Khalili-Damghani, A.-R. Abtahi, and M. Tavana, "A new multi-objective particle swarm optimization method for solving reliability redundancy allocation problems," *Reliability Engineering & System Safety,* vol. 111, pp. 58-75, 2013.

[30] Y. Zhang, L. Wu, and S. Wang, "UCAV path planning by fitness-scaling adaptive chaotic particle swarm optimization," *Mathematical Problems in Engineering,* 2013.

[31] L. Xu, J. Wang, Y.-p. Li, Q. Li, and X. Zhang, "Resource allocation algorithm based on hybrid particle swarm optimization for multiuser cognitive OFDM network," *Expert Systems with Applications,* vol. 42, no. 20, pp. 7186-7194, 2015.

[32] A. Darwish, D. Ezzat, and A. E. Hassanien, "An optimized model based on convolutional neural networks and orthogonal learning particle swarm optimization algorithm for plant diseases diagnosis," *Swarm and Evolutionary Computation,* vol. 50, p. 100616, 2020.

[33] F. Rosenblatt, "The perceptron: a probabilistic model for information storage and organization in the brain," *Psychological review,* vol. 65, no. 6, p. 386, 1958.

[34] M. W. Gardner and S. Dorling, "Artificial neural networks (the multilayer perceptron)—a review of applications in the atmospheric sciences," *Atmospheric environment,* vol. 32, no. 14-15, pp. 2627-2636, 1998.

[35] A. Apicella, F. Donnarumma, F. Isgrò, and R. Prevete, "A survey on modern trainable activation functions," *Neural Networks,* 2021.

[36] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE,* vol. 86, no. 11, pp. 2278-2324, 1998.

[37] S. Wang, S. Ma, S. Wang, D. Zhao, and W. Gao, "Rate-GOP based rate control for high efficiency video coding," *IEEE Journal of selected topics in signal processing*, vol. 7, no. 6, pp. 1101-1111, 2013.

[38] M. Wang, K. N. Ngan, and H. Li, "Low-delay rate control for consistent quality using distortion-based Lagrange multiplier," *IEEE Transactions on Image Processing*, vol. 25, no. 7, pp. 2943-2955, 2016.

[39] F. Song, C. Zhu, Y. Liu, and Y. Zhou, "A new GOP level bit allocation method for HEVC rate control," in *2017 IEEE International Symposium on Broadband Multimedia Systems and Broadcasting (BMSB)*, 2017.

[40] S. Chen, S. Aramvith, and Y. Miyanaga, "Encoder Control Enhancement in HEVC Based on R-Lambda Coefficient Distribution," in *2019 International Symposium on Multimedia and Communication Technology (ISMAC)*, 2019.

[41] G. Lu, X. Zhang, L. Chen, and Z. Gao, "Novel integration of frame rate up conversion and HEVC coding based on rate-distortion optimization," *IEEE Transactions on Image Processing*, vol. 27, no. 2, pp. 678-691, 2017.

[42] U. S. Kim and M. H. Sunwoo, "New frame rate up-conversion algorithms with low computational complexity," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 24, no. 3, pp. 384-393, 2013.

[43] J.-H. Kim, Y.-H. Ko, H.-S. Kang, S.-W. Lee, and J. W. Kwon, "Frame rate up-conversion method based on texture adaptive bilateral motion estimation," *IEEE Transactions on Consumer Electronics*, vol. 60, no. 3, pp. 445-452, 2014.

[44] W.-S. Park and M. Kim, "CNN-based in-loop filtering for coding efficiency improvement," in *2016 IEEE 12th Image, Video, and Multidimensional Signal Processing Workshop (IVMSP)*, 2016.

[45] Y.-w. Lee, J.-h. Kim, Y.-j. Choi, and B.-g. Kim, "CNN-based approach for visual quality improvement on HEVC," in *2018 IEEE International Conference on Consumer Electronics (ICCE)*, 2018.

[46] R. Yang, M. Xu, Z. Wang, and T. Li, "Multi-frame quality enhancement for compressed video," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

[47] F. Bossen, "Common test conditions and software reference configurations," *JCTVC-L1100*, vol. 12, no. 7, 2013.

[48] Y. Guo, Y. Liu, A. Oerlemans, S. Lao, S. Wu, and M. S. Lew, "Deep learning for visual understanding: A review," *Neurocomputing*, vol. 187, pp. 27-48, 2016.

[49] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016.

[50] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *2009 IEEE conference on computer vision and pattern recognition*, 2009.

[51] "HEVC Reference Software," 2014. [Online]. Available: http://hevc.kw.bbc.co.uk/trac/browser/jctvc-hm/tags.

REFERENCES

VITA

NAME	Sovann Chen
DATE OF BIRTH	9 March 1992
PLACE OF BIRTH	Cambodia
INSTITUTIONS ATTENDED	Chulalongkorn University (2017-present) Chulalongkorn University (2015-2017)
PUBLICATION	Chen, S., Aramvith, S., & Miyanaga, Y. (2019, August). Encoder Control Enhancement in HEVC Based on R-Lambda Coefficient Distribution. In 2019 International Symposium on Multimedia and Communication Technology (ISMAC) (pp. 1-4). IEEE.