SCALABLE VIDEO ENCODING, ADAPTATION, AND RATE MODELING

DISSERTATION

Submitted in Partial Fulfillment of

the Requirements for

the Degree of

DOCTOR OF PHYLOSOPHY (Electrical Engineering)

at the

POLYTECHNIC INSTITUTE OF NEW YORK UNIVERSITY

by

Meng Xu

January 2014

SCALABLE VIDEO ENCODING, ADAPTATION, AND RATE MODELING

DISSERTATION

Submitted in Partial Fulfillment of

the Requirements for

the Degree of

DOCTOR OF PHYLOSOPHY (Electrical Engineering)

at the

POLYTECHNIC INSTITUTE OF NEW YORK UNIVERSITY

by

Meng Xu

January 2014

Approved:

_____

Department Head Signature

_____

Date

Copy No. #

Student ID# 0350422

Approved by the Guidance Committee:

Major: Electrical Engineering

Yao Wang
Professor of Electrical and Computer Engineering

Date

Shivendra Panwar
Professor of Electrical and Computer Engineering

Date

Yong Liu
Associate Professor of Electrical and Computer Engineering

Date

Zhan Ma
Senior Engineer, FutureWei Technologies, Inc.

Date

Minor: Computer Science

Edward Wong
Associate Professor of Computer Science and Engineering

Date

Vita

Meng Xu was born in China on June 25th, 1984. He received the B.S. degree in Physics from Nanjing University, Nanjing, China, in 2006, and M.S. degree in Electrical Engineering from Polytechnic Institute of New York University, Brooklyn, NY, in 2007. Since 2009, he has been a Ph.D. student at Electrical and Computer Engineering Department in Polytechnic Institute of New York University, Brooklyn, NY under the supervision of Professor Yao Wang. From June 2010 to August 2010, February 2013 to May 2013, and June 2013 to August 2013, he interned at Dialogic Media Labs, NJ, Samsung Telecommunications America, L.L.C., TX, and Futurewei Technologies, Inc., CA, respectively. He has conducted researches in the field of video coding and its applications, primarily focused on scalable video coding.

Acknowledgment

First and foremost, I would like to take this opportunity to express the deepest appreciation to my advisor Prof. Yao Wang. Without her endless patience, advice, inspiration, guidance and support, this thesis would not have been realized.

Furthermore, I would also like to thank the rest members of my thesis committee: Prof. Shivendra Pawar, Prof. Yong Liu, Prof. Edward Wong, and Dr. Zhan Ma, for their warm encouragement, insightful comments, and hard questions.

I am especially grateful to Dr. Zhan Ma, who provided persistent help in many projects during my Ph.D study, and assisted me in gaining experience for my future career. I would also like to express the gratitude to my supervisors and colleges during my internship: Dr. Kyeong Yang, Dr. Felix Fernandes, Dr. Imed Bouazizi, and Dr. Haoping Yu, for their invaluable experience and insightful technical discussions.

I appreciate the assistance given by Prof. Jose Ulerio, Dr. Xiao-Kang Chen, and Valerie Davis, for their warm help during my study at NYU-Poly. My sincere thanks also go to my lab mates of the Video Lab: Dr. Zhengye Liu, Dr. Yen-Fu Ou, Dr. Hao Hu, Xiaozhong Xu, Xuan Zhao, Yuanyi Xue, Zhili Guo, Eymen Kurdoglu, Andy Chiang, Jen-Wei Kuo, and Shervin Minaee, for their collaborations and valuable discussions.

Lastly, but by no means the least, I need to thank my family for all their love, encouragement, and support throughout my life.

ABSTRACT

SCALABLE VIDEO ENCODING, ADAPTATION, AND RATE MODELING

by

Meng Xu

Advisor: Prof. Yao Wang

Submitted in Partial Fulfillment of the Requirements for the Degree of Doctor of Philosophy (Electrical Engineering)

December 2013

With the widespread high-speed Internet and mobile wireless networks, multimedia content, especially video content, is dominating the consumer network traffic. The users enjoying online video services such as video broadcasting or instant video communication usually have different connection bandwidth, different screen size on the device, and different demand or tolerance on video quality. To adapt to the heterogeneous connection conditions and different user demands, the scalable video coding extension of the H.264 standard (SVC) is a promising approach, where multiple bitstreams containing different temporal, quality, or spatial resolutions can be encoded into a single bitstream, and extracted later based on the demand.

SVC adopts the layered coding technique, where the base layer carries only fundamental information while the enhancement layers carry the refinement information to

produce the enhanced quality. Compared with traditional single-layered video, SVC requires about 20% more bits to maintain the same reconstructed video quality. Moreover, the complexity grows linearly as the number of layers increases.

This thesis consists of two components. First we present solutions to reduce the SVC coding complexity without much loss in coding efficiency. Then we propose a rate model that can predict the bits needed for coding a block from its prediction error and the quantization stepsize. We further consider how to use this model to predict the total rate at different temporal layers. In the first part, we attack the SVC coding efficiency and complexity jointly. By analyzing the conventional encoding algorithm for SVC, we design a novel coding scheme by exploiting the correlation between the layers. In our approach, different quality layers of the same coding unit are forced to use the same mode and the same motion vector(s) if an Inter-mode is chosen. The mode and motion vectors are determined at the base layer only but using the information from the highest layer as well. By forgoing motion estimation and mode decision at higher layers, the complexity of enhancement layers is reduced to a negligible level, without much sacrifice in the coding efficiency. For some test sequence, the proposed scheme even achieves better coding efficiency, due to the fact that no mode and motion information need to be specified at higher layers.

To further reduce the coding complexity, we investigate the existing early Skip technique, and extend it with our unified Direct mode. The proposed early Skip/Direct (ESD) mode decision allows the computationally intensive mode decision to be bypassed if the ESD condition is satisfied. By exploring the quantization process in the video coding, we choose to use the averaged quantization error as the threshold. The ESD mode decision is further integrated with our multilayer mode decision for SVC, resulting significant complexity reduction with only slight coding efficiency degradation.

In the second part, we investigate the conventional rate model that relates the video bitrate with the prediction error and the quantization error. The conventional model

relates the video rate linearly with the logarithm of the prediction error, which fails when the bitrate is low. We propose a non-linear model that is fitted from the collected data. We further show that the quantization error can be modeled by a power function with respect to the quantization stepsize. The complete model can predict the number of bits required for coding a block from its prediction error and the quantization stepsize. The model requires four parameters, which can be predicted by content features via light-weighted prepossessing.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Challenge and Motivation

We are now witnessing a revolution in the Information Age, which sometimes is also known as the New Media Age, where the information sharing, especially the multimedia content delivery, keeps growing in a dramatic fast pace. With the widespread high-speed Internet and mobile wireless network such as 4G, the emerging video services, and the growing demand on the multimedia contents, it is expected that in the near future, video traffic in the network will continue to up-trend in a faster pace. Cisco has predicted the Internet traffic for the next a few years [1], as visualized in Fig. 1.1. Clearly the video traffic will dominate the consumer Internet traffic in the near future.

Among the rapidly increasing video traffic, video broadcasting and instant video communication are two major categories of the on-line video applications. The users watching video on different types of hardware generate different demand on level of quality. For example, a user watching on a smart phone via mobile network may wish to receive a standard-definition (SD) video while another user watching on a large screen may request the high-definition (HD) quality. Besides, all the users have different

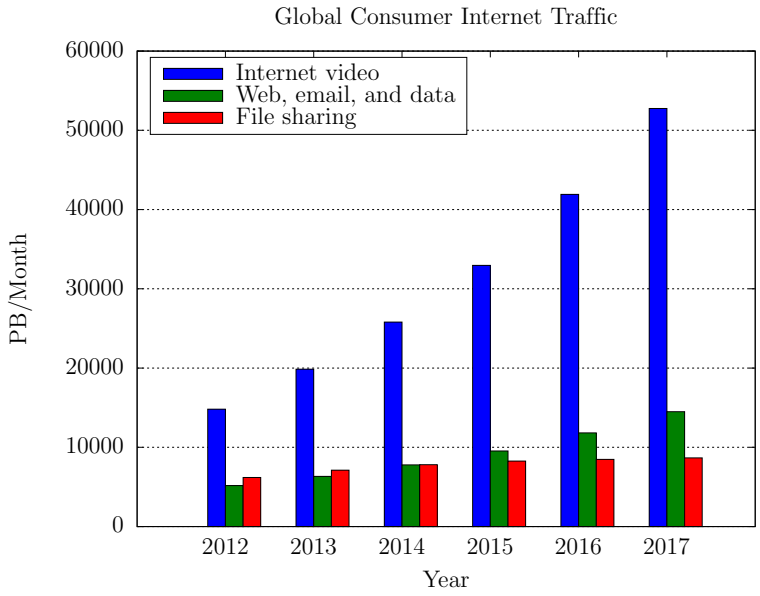


Figure 1.1: Video traffic forecast 2012–2017. The source data is from Cisco VNI [1].

bandwidth allocation from their Internet service providers.

Under this heterogeneous demand on video quality and network condition, it is challenging to deliver video of different level of quality to different users. Several solutions have been designed, among which the scalable video coding (SVC) promises a great potential, where a single coded video bit stream can be adapted at different levels of quality. However, there are two obstacles that cause the SVC not being widely adopted: the coding efficiency and the complexity.

The ability to adapt to multiple sub streams does not come free. Compared with a non-scalable video, SVC bit streams usually are 20% larger under similar quality, and the encoding complexity is at least doubled. We propose to attack the coding efficiency and complexity together. After exploring the encoder structure, a multilayer encoding scheme is designed to reduce the cross-layer redundancy. Our simulation results show more than 50% encoding time saving on a three-layer structure, with marginally worse or even better coding efficiency.

In this chapter, we provide a brief summary of some of the relevant concepts of

video coding in H.264 for AVC and SVC. The proposed methods will be detailed in the following chapters.

1.2 Brief review of video coding in H.264

1.2.1 Video coding standards

The uncompressed video (also referred as raw video) has extremely huge data rate and is not affordable for large scale distribution. To achieve high compression ratio, various video coding algorithms are designed to reduce the redundancies in the raw video signals.

Those video coding techniques have been standardized by two organizations: International Telecommunication Union (ITU) and International Organization for Standardization (ISO). A series of video coding standards have been published separately or through their joint work, such as H.261 [2], MPEG-1 Video [3], H.262/MPEG-2 Video [4], H.263 [5], MPEG-4 Visual [6], H.264/MPEG-4 AVC [7] (also referred as H.264/AVC, H.264, or AVC), and the very recent H.265/MPEG-H HEVC [8].

Among these standards, H.264 is the most successful one and gradually dominates the market since it was published in 2003. For its higher coding efficiency compared with its predecessors, it has brought vast interest in the industry. Its successor, H.265, published in 2013, although can achieve twice compression ratio than H.264 does while maintaining the same video quality, its huge complexity prevents it from being widely used in the near future. Therefore our work is based on the H.264.

1.2.2 Working flow of video codec

In the encoder, the source video is process by three modules, namely prediction, transform and quantization, and entropy coding, as shown in Fig. 1.2.

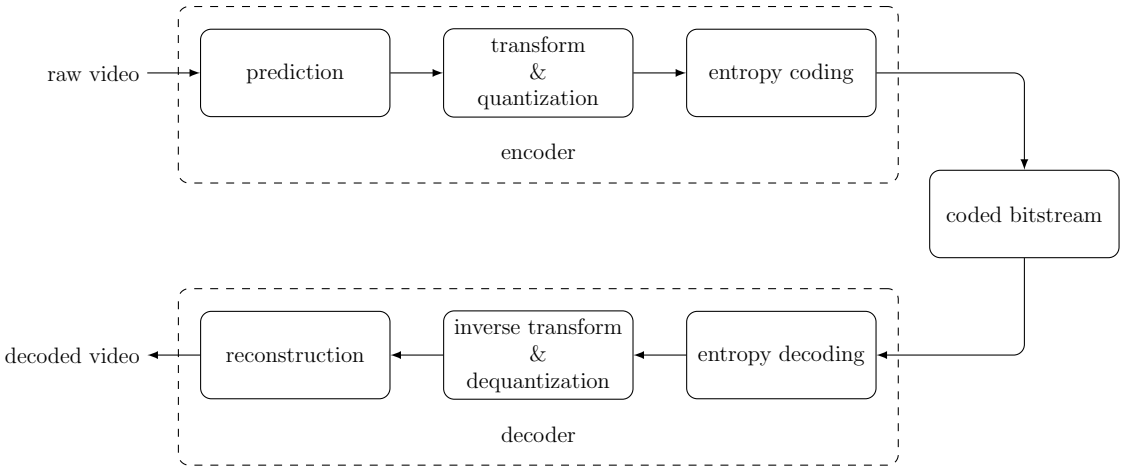


Figure 1.2: Diagram of conceptual video encoding and decoding process.

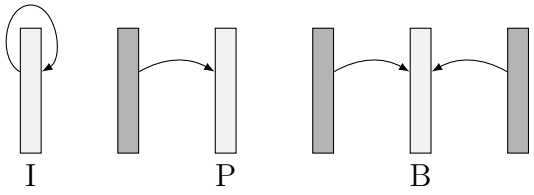


Figure 1.3: Frame prediction types.

Since video frames have high temporal and spatial correlations, predictive coding is designed to exploit these correlations. For each frame, other than code it directly, it is predicted from a reference frame, and the the prediction error (also referred as residual) is coded. H.264 allows multiple frames to be selected as the reference; however we choose to use only one reference frame in this work to targeting at low-complexity encoding.

There are three prediction types for video frames, i.e., Intra-frame, Predictive-frame, and Bi-predictive-frame, as illustrated in Fig. 1.3. Intra-frame (I-Frame) exploits the spatial correlation, where the same frame is used as the reference. In a Predictive-frame (P-frame), addition the tools available in I-frame, the temporal correlation is also utilized, with the previous frames used as reference. Bi-predictive-frame (B-frame) is similar to P-frame, except that future frames can also be used as the reference.

Once the prediction errors are obtained, they are then transformed and quantized in

the next stage. The goal of transform is to eliminate the low energy components in the error signal that is not sensitive to human eyes, while the aim of quantization is to represent the signal by limited number of bits (and yielding the corresponding quantization error). In H.264, the quantization is controlled by the quantization parameter (QP), which ranges from 0 to 51 (QP of 0 means lossless coding). The quantization stepsize q is determined by

$$q = 2^{\frac{\mathrm{QP}-4}{6}}. \tag{1.1}$$

At the final stage in the encoder, the quantized transformed coefficients, together with the syntax elements, are encoded by the entropy coder to produce the bitstream.

For video delivery, the video bitstreams are usually multiplexed with coded audio streams and/or subtitles, using various container formats such as AVI, MP4, MKV, etc. When such video file (static file or dynamic stream) is opened by the player, the player extracts the video bitstrem and forwards it to the video decoder.

In the decoder, the inverse operations, including entropy decoding, quantization and inverse transform, as well as reconstruction, are performed to obtain the decoded video signal. Since video coding is generally lossy, the reconstructed signal is not exactly the same as the source signal. The video bitrate and the quality are used to evaluate the coding efficiency of an encoder. The Peak Signal-to-Noise-Ratio (PSNR) is widely used as an objective quality metric in video process. For a video frame with typical 8-bit depth (i.e., the peak pixel value is 255), the PSNR is defined as

$$\mathrm{PSNR} = 10 \log_{10} \frac{255^2}{\mathrm{MSE}}, \tag{1.2}$$

where MSE is the prediction error measured in terms of the mean square error (between the reconstruction and the source) of the entire frame. For a video sequences, the PSNR is usually averaged over all the frames.

1.2.3 Color space conversion and down-sampling

For the raw video captured by the image sensors, color space down-sampling is designed to eliminate the less important data in the color components. Although the image sensors produce native RGB (red, green, and blue) components for each pixel, it is usually converted to YCbCr[1] color space before coded. The commonly used conversion defined in the ITU-R BT.601 standard [9] is as follows:

$$\begin{bmatrix} Y \\ Cb \\ Cr \end{bmatrix} = \begin{bmatrix} 65.481 & 128.553 & 24.966 \\ -37.797 & -74.203 & 112.0 \\ 112.0 & -93.786 & -18.214 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} + \begin{bmatrix} 16 \\ 128 \\ 128 \end{bmatrix}. \tag{1.3}$$

The Y component carries the luminance information and referred as the luma component, while the Cb and Cr components carry the chrominance information and refereed as the chroma components. Since human eyes are less sensitive for the chroma components than in the luma, the chroma components are usually down-sampled. In the widely used YUV 4:2:0 format, Cb and Cr components are down-sampled by half in both horizontal and vertical directions.

1.2.4 Macroblock types in H.264

After the pre-processing for the video source, each frame is divided into small processing units, where various coding algorithms are applied. In H.264, the basic coding unit is called macroblock (MB) that contains 16×16 pixels [10], which can be further split into smaller sub-blocks, as illustrated in Fig. 1.4.

In the prediction stage, a prediction signal is generated for each MB, and then the prediction error (also known as residual) is calculated by subtracting the original signal. There are a number of MB types defined in the H.264 standard, each specifying an

[1]When referring to digital video signals, it is often interchangeably used with the term YUV.

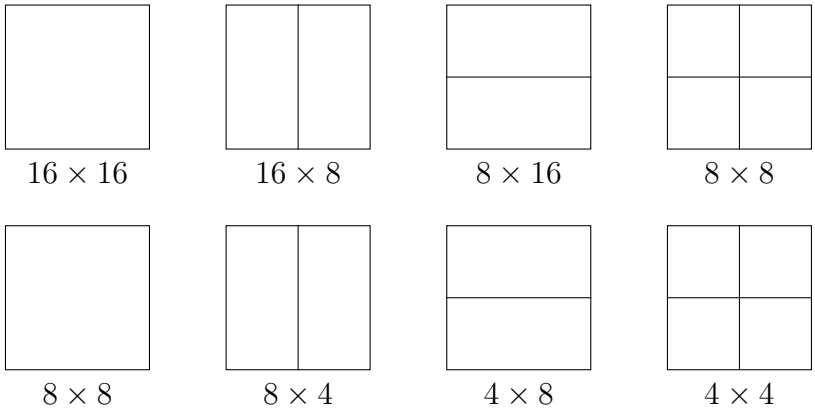


Figure 1.4: Macroblock partitions in H.264.

algorithm to code that MB. These types fall in two categories, i.e., Intra-modes and Inter-modes, exploring the spatial and temporal correlations, respectively.

Intra-picture prediction modes, also referred as Intra-modes, are designed to reduce the spatial redundancy within the same frame. In the Intra-modes, the spatial neighboring blocks (if available) are used to predict the current MB. The predicted pixels are generated by copying or interpolating from the boundary pixels using different directions. A DC mode is also available to yield homogeneous samples using averaged boundary pixels. There are 9 prediction modes defined in H.264 for 4×4 Intra-blocks, as shown in Fig. 1.5. Similarly, 4 prediction modes are defined for the 16×16 Intra-blocks.

Inter-picture prediction modes, also referred as Inter-modes, exploit the temporal correlation among different frames. The encoder finds a best-matching block in the reference frame, and uses it as the prediction for the current block. The location displacement is called the motion vector (MV), as illustrated in Fig. 1.6. The MV and the best-matching block are obtained by the motion estimation process, also known as motion search. For the MB coded using Inter-mode, a predicted motion vector (PMV) is derived from the spatial neighboring blocks. The PMV is used as the start point for the motion search. The difference between the MV obtained from motion search and

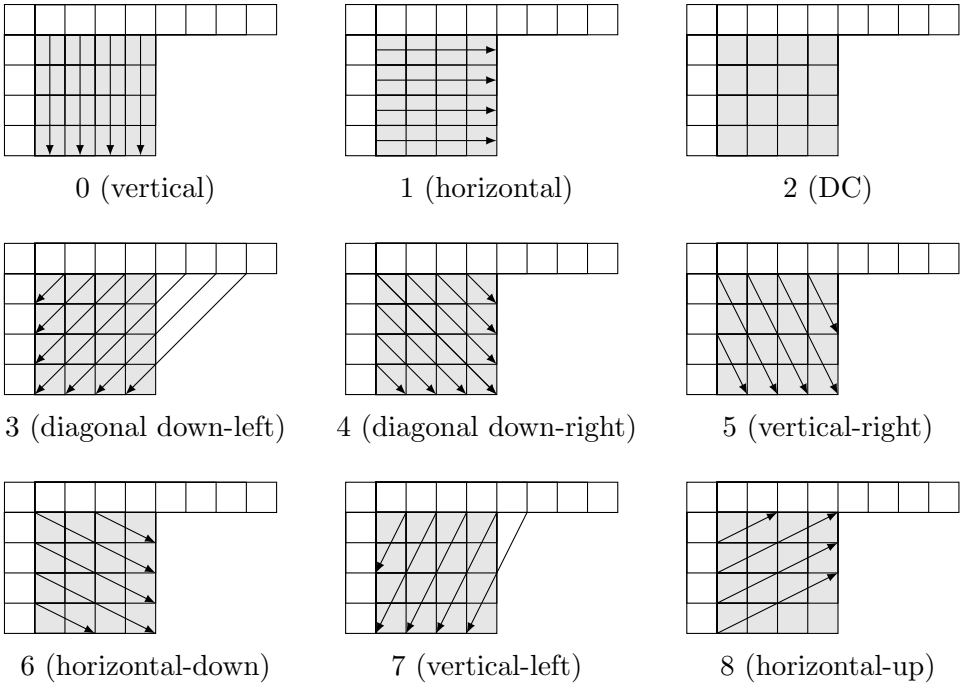


Figure 1.5: 4×4 Intra-prediction modes in H.264.

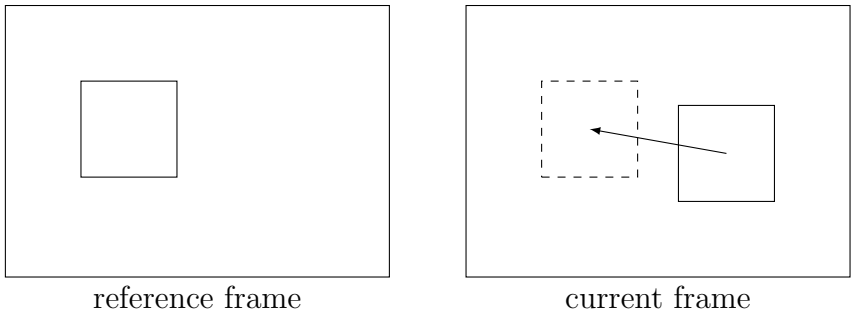


Figure 1.6: Illustration of Inter-mode in H.264.

the PMV is called motion vector difference (MVD), and coded into the bitstream.

Although various fast motion search algorithms have been developed, it is still the major complexity bottleneck in the encoder. A so-called early skip technique is developed to reduce the complexity, and is detailed in Chapter 3.

1.3 Introduction to scalable video coding in H.264/SVC

The idea of adapting a single video bit stream for various scenarios has been introduced in the video coding standards developed in the early years, such as H.262/MPEG-2 Video [4], H.263 [5], and MPEG-4 Visual [6]. However, the scalable profiles in these standards are not widely used in the market, due to the poor coding efficiency and high coding complexity. The most recent published scalable video coding standard is developed as an extension of the H.264/AVC, and is denoted as H.264/SVC, or simply SVC. H.264/SVC inherits the coding tools from H.264/AVC that has significantly higher coding efficiency than the prior standards, thus the SVC has the potential to be used in the market.

The scalability in SVC is achieved through layered video coding, where the video content is coded into multiple layers. The base layer (BL) carries fundamental information that can only produce limited video quality, while the enhancement layer (EL) carries refinement over the BL, providing enhanced video quality. As the BL inherits all the mode candidates from AVC, the EL enjoys additional inter-layer modes to utilize the inter-layer correlation.

To utilize the inter-layer correlation, several coding tools are designed at the EL inherit the information form the BL. The inter-layer mode prediction, enables the EL to reuse the lower layer mode with little addition cost. (More details of inter-layer mode decision will be presented in Chapter 2 together with the proposed mode decision algorithm.) The inter-layer motion prediction, allows the EL to reuse the MV from the BL, even if they are coded with different modes. In addition, the residue prediction, exploits the correlation of the transform coefficients between two adjacent layers.

There are three types of scalability supported in the H.264/SVC standard, namely temporal, quality, and spatial scalability [11].

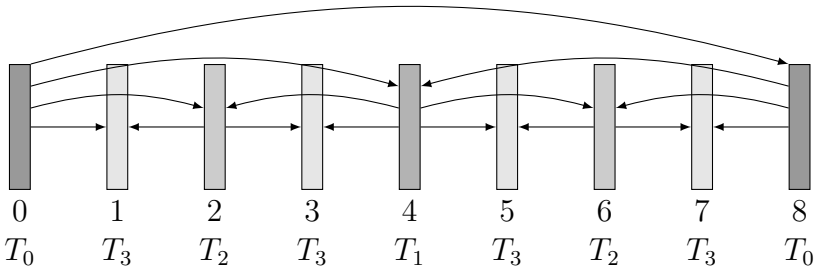


Figure 1.7: Illustration of temporal layers in SVC with GOP length of 8. Frames belong to the temporal layer #0, #1, #2, and #3 are labeled as T_0, T_1, T_2, and T_3, respectively.

Temporal scalability enables a scalable bitstream to be extracted at different frame rates. In H.264/SVC it is supported via the hierarchical coding structure. Fig. 1.7 illustrates a dyadic temporal scalability coding structure with GOP length 8. In this example, the frames at temporal layer #1, #2, and #3 are coded as B-frames with a hierarchical structure.

The quality scalability allows each layer to be coded using different quantization parameters to tweak the video quality at each layer. There are two types of quality scalability in SVC: coarse grain scalability (CGS) and medium grain scalability (MGS). More details of the encoder design for CGS and MGS will be discussed together with the proposed algorithm in Chapter 2.

The spatial scalability in SVC allows each layer to be coded with different frame resolutions. The study of the spatial scalability is deferred as our future work.

1.4 Dissertation layout

This dissertation is organized as follows. In Chapter 2, we first analyze the mode decision scheme used in the conventional SVC encoder, then propose a low-complexity multilayer mode decision algorithm that determines the mode jointly for all layers, to reduce the coding complexity at the enhancement layers while maintaining the coding efficiency.

The joint mode is determined at the base layer using the higher layer information. Once the best mode is selected, it is then reused at higher layers.

In Chapter 3, we design a low complexity mode decision algorithm targeting at single-layered video and the base layer of SVC, using the early Skip/Direct mode decision technique. To integrate it with the multilayer mode decision scheme presented in Chapter 2, slight algorithm modifications are made to the original multilayer mode decision scheme. In the combined algorithm, the motion estimation is conducted at most once among all layers.

In Chapter 4, we present a rate model that directly relates the rate with the quantization stepsize and the prediction error variance.. While the conventional model only works at the high bitrate, our proposed model is designed for both low and high rate range.

In Appendix **??**, we discuss how the conventional rate model can be derived from a first-order Gaussian-Markov source and why it behaves differently at low rate and high rate.

The frequently used abbreviations and notations are listed on page 101.

Chapter 2

Cross-layer mode decision for quality scalability

In this chapter we first briefly review the mode decision algorithm in the conventional SVC encoder, then present a novel multilayer mode decision algorithm for the quality scalable video coding. While the conventional mode decision method performs exhaustive search at each layer, the proposed scheme determines the mode only at the lowest layer, but using the information from the highest layer. Once the best mode has been determined at the BL, the higher layers simply reuse this mode without computationally expensive mode decision. As shown by the simulation results, the proposed scheme achieves an overall coding efficiency very close to the original SVC, but significantly low complexity. The complexity reduction for the BL will be discussed in Chapter 3.

2.1 Related works

With numerous possible modes defined in the H.264/SVC standard and each generates a different rate (R) and distortion (D) pair, it is the encoder's responsibility to choose a wise mode for each block. The mode decision algorithm is therefore the essence of a

video encoder. To achieve the highest coding efficiency, rate-distortion optimized mode decision is usually used. With this scheme, all possible mode candidates for the current MB are exhaustively evaluated, and the mode is determined by choosing the one with best R-D trade-off. In a conventional implementation of an SVC encoder, including in the H.264/SVC reference software JSVM [12], this approach is applied in all layers. With an R-D optimized mode decision algorithm, the mode chosen at each layer is optimal only for the current layer, and the global optimality is not guaranteed. Li *et. al* proposed to tweak the Lagrangian parameter for the ELs to boost the coding performance [13]. Although this method has later been adopted as an option in the JSVM software, the inter-layer dependency has still yet to be explored.

The cross-layer rate-distortion optimization (RDO) based mode decision was studied by Schwarz *et. al* in [14], where the mode is jointly determined for all layers, yielding the best performance of 10% overhead in bitrate for two-layer structure compared with AVC. However, since the possible EL modes are taken into account when deciding current layer mode, it requires multiple motion search even within the same layer. The significant amount of encoding time makes it impracticable for structures with more than two layers. Li *et. al* proposed an improved scheme [15] that requires only single motion search at each layer. However, it still has higher complexity than the conventional JSVM encoder. These works emphasize on boosting the encoding efficiency, without too much consideration of the complexity reduction.

With the independent mode decision at each layer, under an L-layer structure, the encoder complexity is usually more than L times of that as in AVC. This huge complexity of the current SVC encoder (using the JSVM implementation [12]) was verified by Alfonso *et. al* [16]. There are a number of research works targeting at SVC encoder complexity reduction (with slight coding efficiency sacrifice). In [17, 18], the MB mode correlation between different layers are studied. Then the number of candidate modes at EL can be reduced by pruning off the ones with low correlation with the lower

layer. In addition, the mode decision does not need to be R-D optimized, as long as the selected mode is near optimal. In [19] the EL is completely RDO off, where multiple comparisons using current or lower layer samples are performed to determine the mode. The low complexity mode decision for Intra-modes are studied in [20]. Although quite noticeable complexity reduction can be achieved with these methods, the modes at each layer are still separately determined, hence are not globally optimal.

Different from all prior works, we attack the coding efficiency and complexity reduction jointly in multilayer quality scalable video coding. We propose a cross-layer mode decision, where the motion search is performed only once among all layers. Once the mode is decided at the lower layer (using the information from the higher layers), it is then directly reused by the higher layers.

2.2 Brief review of conventional mode decision algorithm for SVC

For each coding unit, there are a number of possible modes can be used for coding. It is the encoder's responsibility to choose a mode with good R-D trade-off. The RDO-based mode decision method incorporated in the modern encoders was introduced since H.263 by Wiegand *et. al* [21, 22] and also applied in H.264 encoder design [23], where for each mode candidate, a R-D cost function J is computed, and the best mode m^* is determined to have the lowest J, i.e.,

$$m^* = \arg\min_m J\left(m; f, \hat{f}, \lambda\right), \tag{2.1}$$

with the cost function J defined as

$$J\left(m; f, \hat{f}, \lambda\right) = D\left(f, \hat{f}\left(m\right)\right) + \lambda\left(\mathrm{QP}\right) R\left(m, f - \hat{f}\left(m\right)\right), \tag{2.2}$$

where f is the original signal, $\hat{f}$ is the reconstructed signal (from previously encoded frame) used as the reference frame for the Inter-modes, $\hat{f}\left(m\right)$ is the reconstructed signal

coded using $\hat{f}$ as the reference frame with mode candidate m, λ is the Lagrangian multiplier depending on the quantization parameter (QP), and R is the rate to code the mode m and the corresponding distortion $f - \hat{f}$.

In the implementation of JSVM encoder [24], the distortion between f and $\hat{f}(m)$, i.e., $D\left(f, \hat{f}(m)\right)$, is measured in terms of the sum of squared error (SSE) of all pixels in the block, and λ is determined by

$$\lambda(\mathrm{QP}) = 0.85 \times 2^{\frac{\mathrm{QP}-12}{3}}. \tag{2.3}$$

In H.264/AVC, each mode is associated with a macroblock partition. Even using the same partition, the encoder may still have multiple prediction options and need to choose the best one from the available candidates. For the Inter-mode with a given partition, one has to determine the best motion vector (MV) for each sub-block from a list of candidates. Similar to the mode decision process, a cost function J_{Inter} is defined to select the best MV for each sub-block in the RDO-based motion estimation. Each MV candidate can be treated as a prediction method. The best MV v^* is determined by

$$v^* = \arg\min_{v} J_{\mathrm{Inter}}\left(v; f, \hat{f}, \lambda_{\mathrm{MV}}\right), \tag{2.4}$$

and the cost function J_{Inter} is defined as

$$J_{\mathrm{Inter}}\left(v; f, \hat{f}, \lambda_{\mathrm{MV}}\right) = D_{\mathrm{MV}}\left(f, \hat{f}(v)\right) + \lambda_{\mathrm{MV}}(\mathrm{QP})\, R(v), \tag{2.5}$$

where $\hat{f}(v)$ is the compensated signal using the motion vector candidate v with $\hat{f}$ as the reference frame. In H.264/AVC, the reference frame is the previously decoded frame (Note that we only consider the case of using a single reference frame in this paper). $D_{\mathrm{MV}}\left(f, \hat{f}(v)\right)$ and $R(v)$ are the distortion between f and $\hat{f}(v)$, and the rate to encode v, respectively. The Lagrangian multiplier λ_{MV} depends on the distortion criterion used by D_{MV}. In the case that D_{MV} is measured by sum of absolute difference (SAD), λ_{MV} is given by

$$\lambda_{\mathrm{MV}}(\mathrm{QP}) = \sqrt{\lambda(\mathrm{QP})}. \tag{2.6}$$

Although the H.264 standard allows multiple frames to be used as the reference for prediction, it multiplies the time for motion estimation and therefore is impractical for low-complexity encoder. Thus we only consider the case of using a single reference frame in this work.

For the Intra-mode, there are also multiple predictions from the spatial neighbors with various angular and non-angular directions, associated with 16×16 and 4×4 macroblock partition sizes. For a given partition, the best Intra-prediction mode $\tilde{m}$ is determined similarly, by minimizing the cost function J_{Intra}, i.e.,

$$\tilde{m}^* = \arg\min_{\tilde{m}} J_{\text{Intra}}\left(\tilde{m}; f, \tilde{f}, \lambda_{\text{Intra}}\right), \tag{2.7}$$

with the cost function for Intra-prediction defined as

$$J_{\text{Intra}}\left(\tilde{m}; f, \tilde{f}, \lambda_{\text{Intra}}\right) = D_{\text{Intra}}\left(f, \tilde{f}\left(\tilde{m}\right)\right) + \lambda_{\text{Intra}}\left(\text{QP}\right) R\left(\tilde{m}, f - \tilde{f}\left(\tilde{m}\right)\right), \tag{2.8}$$

where $\tilde{f}$ stands for the previously reconstructed signal in the same frame, $\tilde{f}\left(\tilde{m}\right)$ is the predicted block using $\tilde{f}$ and mode m. In JSVM implementation, SAD is also used for measuring D_{Intra}, and λ_{Intra} is determined in the same fashion as λ_{MV}.

The RDO-based mode decision algorithm in JSVM implementation is summarized as follows. For each possible Inter-partition, the encoder first determines the best MV for each sub-block using (2.4). The total distortion and corresponding rate for using this partition is then determined by summing the distortions and rates for all sub-blocks. Similarly, for each Intra-partition, the encoder first determines the best Intra-prediction mode for each sub-block using (2.7), then the total distortion and rate for using this partition is determined by summing the distortions and rates for all sub=blocks. Finally, the encoder uses the mode decision approach described in (2.1), (2.2), and (2.3) to compare all possible Inter-partitions and all possible Intra-partitions, and choose the one that minimizes (2.1).

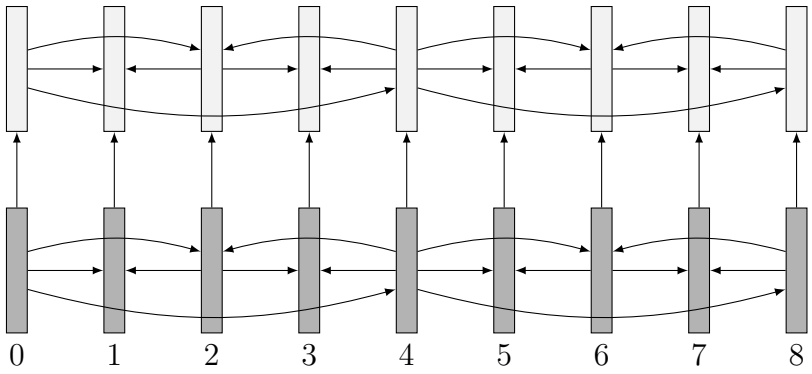


Figure 2.1: Illustration of CGS coding structure with two layers and GOP length of 4.

2.2.1 Conventional RDO-based mode decision method for CGS

Two types of quality scalability are supported in H.264/SVC: Coarse grain scalability (CGS) and Medium grain scalability (MGS). The coding structure for CGS is illustrated in Fig. 2.1. In a bitstrem encoded under CGS structure, the bitsteam switching between layers can occur only at IDR frames[1]. CGS applies the so-called two-loop encoding control, where the mode decision at the BL and the EL are carried out separately, each using the current layer as reference, and the EL is featured with additional inter-layer prediction tools.

In the conventional SVC encoder such as JSVM, the RDO-based motion estimation and mode decision algorithm is applied in all layers [24], using the previously decoded frame for the current layer as the reference frame. It also uses the Lagrangian multiplier determined from the QP used in the current layer. Specifically, at i-th layer, the best mode the m_i^* and motion vector v_i^* at i-th layer are determined by (2.1) and (2.4), with

$$\hat{f} = \hat{f}_i, \quad \lambda = \lambda\left(\mathrm{QP}_i\right), \quad \lambda_{\mathrm{MV}} = \lambda_{\mathrm{MV}}\left(\mathrm{QP}_i\right), \tag{2.9}$$

[1]IDR frame is a special I-frame used for fast seeking within the video stream. In typical video applications, IDR frames are inserted in the bitstream at a period of 2 – 3 seconds.

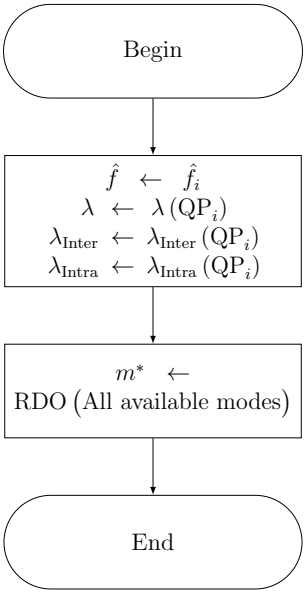


Figure 2.2: Conventional mode decision algorithm for CGS coding structure.

where QP_i denotes the QP used in i-th layer, i.e.,

$$\begin{aligned} v_i^* &= \arg\min_{v_i} J_{\mathrm{Inter}}\left(v_i; f, \hat{f}_i, \lambda_{\mathrm{Inter}}(\mathrm{QP}_i)\right) \\ &= \arg\min v_i \left(D_{\mathrm{MV}}\left(f, \hat{f}_i(v_i)\right) + \lambda_{\mathrm{Inter}}(\mathrm{QP}_i) R(v_i)\right), \end{aligned} \tag{2.10}$$

and

$$\begin{aligned} m_i^* &= \arg\min_{m_i} J\left(m_i; f, \hat{f}_i, \lambda(\mathrm{QP}_i)\right) \\ &= \arg\min_{m_i} \left(D\left(f, \hat{f}_i(m_i)\right) + \lambda(\mathrm{QP}_i) R\left(m_i, f - \hat{f}_i(m_i)\right)\right). \end{aligned} \tag{2.11}$$

Similarly, the best Intra-prediction mode $\tilde{m}_i$ at i-th layer is determined by

$$\begin{aligned} \tilde{m}_i^* &= \arg\min_{\tilde{m}_i} J_{\mathrm{Intra}}\left(\tilde{m}_i; \hat{f}_i, \lambda_{\mathrm{Intra}}(\mathrm{QP}_i)\right) \\ &= \arg\min_{\tilde{m}_i} \left(D_{\mathrm{Intra}}\left(f, \hat{f}_i(\tilde{m}_i)\right) + \lambda_{\mathrm{Intra}}(\mathrm{QP}_i) R\left(\tilde{m}_i, f - \hat{f}_i(\tilde{m}_i)\right)\right), \end{aligned} \tag{2.12}$$

using the previously reconstructed signal $\hat{f}_i$ from current layer as the reference.

The flowchart of this method is demonstrated in Fig. 2.2, where the operation $\mathrm{RDO}(\mathbf{X})$ is defined to choose the best mode (including the MV derivation) from the list of mode candidates $\mathbf{X}$, using (2.1), (2.4) and (2.7).

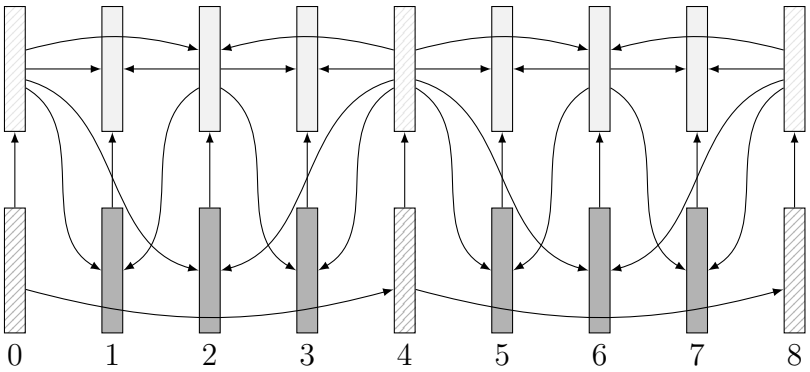


Figure 2.3: Illustration of MGS coding structure with two layers and GOP length of 4. Frames #0, #4, and #8 are encoded as the key frames.

2.2.2 Conventional RDO-based mode decision method for MGS

Compared with CGS, MGS has a quite different coding structure, where the highest layer is used as reference for both prediction and reconstruction for all layers, as shown in Fig. 2.3. A bitstream coded with MGS structure can be switched between layers at the group-of-pictures (GOP) boundary (with a finer granularity) instead of at the IDR frames. With this scheme, an MGS bitstream can be extracted at the frame granularity.

Since the MGS uses the reference frame from the highest layer that has the best reconstructed quality, MGS can noticeably improve the coding efficiency. However, potential error drift might be introduced and propagated due to the packet/slice loss at higher enhancement layer. To control the error drift due to the possible data missing in the highest layer, key frames are introduced, and usually the I- or P-frames at the GOP boundaries are coded as key frames (frames #0, #4, and #8 in the example shown in Fig. 2.3). With referencing only from the BL, key frames are immune to the loss in higher layers, thus the error drift is confined within the erroneous GOP.

For key frames, the best mode (including the MV if the block is Inter-coded) at layer

i is determined using (2.1) and (2.4), with

$$\hat{f} = \hat{f}_0, \quad \lambda = \lambda\left(\mathrm{QP}_i\right), \quad \lambda_{\mathrm{MV}} = \lambda_{\mathrm{MV}}\left(\mathrm{QP}_i\right). \tag{2.13}$$

Note that key frames use the BL as the reference for both prediction and reconstruction. With this approach, the best mode m_i^* and MV v_i^* at i-th layer are derived from

$$\begin{aligned} m_i^* &= \arg\min_{m_i} J\left(m_i; f, \hat{f}_0, \lambda\left(\mathrm{QP}_i\right)\right) \\ &= \arg\min_{m_i} \left(D\left(f, \hat{f}_0\left(m_i\right)\right) + \lambda\left(\mathrm{QP}_i\right) R\left(m_i, f - \hat{f}_0\left(m_i\right)\right)\right), \end{aligned} \tag{2.14}$$

and the best motion vector v_i^* is derived by

$$\begin{aligned} v_i^* &= \arg\min_{v_i} J_{\mathrm{Inter}}\left(v_i; f, \hat{f}_0, \lambda_{\mathrm{MV}}\left(\mathrm{QP}_i\right)\right) \\ &= \arg\min_{v_i} \left(D_{\mathrm{Inter}}\left(f, \hat{f}_0\left(v_i\right)\right) + \lambda_{\mathrm{MV}}\left(\mathrm{QP}_i\right) R\left(v_i\right)\right). \end{aligned} \tag{2.15}$$

For non-key frames, the highest layer is used as reference for both prediction and reconstruction. In an L-layer structure, the best mode m_i^* at i-th layer is determined by

$$\hat{f} = \hat{f}_L, \quad \lambda = \lambda\left(\mathrm{QP}_i\right), \quad \lambda_{\mathrm{MV}} = \lambda_{\mathrm{MV}}\left(\mathrm{QP}_i\right), \tag{2.16}$$

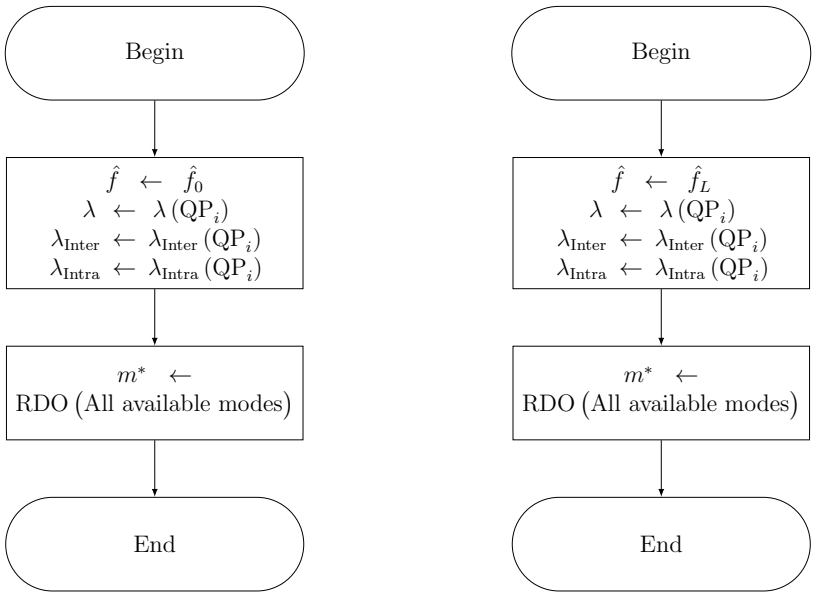
i.e., the best mode m_i^* is derived from

$$\begin{aligned} m_i^* &= \arg\min_{m_i} J\left(m_i; f, \hat{f}_L, \lambda\left(\mathrm{QP}_i\right)\right) \\ &= \arg\min_{m_i} \left(D\left(f, f, \hat{f}_L\left(m_i\right)\right) + \lambda\left(\mathrm{QP}_i\right) R\left(m_i, f - \hat{f}_L\left(m_i\right)\right)\right), \end{aligned} \tag{2.17}$$

and the best motion vector v_i^* is selected using

$$\begin{aligned} v_i^* &= \arg\min_{v_i} J_{\mathrm{Inter}}\left(v_i; f, \hat{f}_L, \lambda_{\mathrm{MV}}\left(\mathrm{QP}_i\right)\right) \\ &= \arg\min_{v_i} \left(D_{\mathrm{MV}}\left(f, \hat{f}_L\left(v_i\right)\right) + \lambda_{\mathrm{MV}}\left(\mathrm{QP}_i\right) R\left(v_i\right)\right). \end{aligned} \tag{2.18}$$

The Intra-prediction modes in MGS (for both key and non-key frames) are determined using the same method as in CGS, using (2.12). Note that for the Intra-mode, the reference signal f always comes from the reconstructed blocks within the same frame



(a) Key frame (b) Non-key frame

Figure 2.4: Conventional mode decision algorithm for key frame and non-key frame in MGS coding structure.

in the same layer (except for the inter-layer Intra-mode (IntraBL), where the reference signal comes from the lower layer, which will be discussed in detail in in Section 2.3.2). Unlike the Inter-modes, it is extremely difficult to obtain the higher layer reconstruction of previously coded blocks in this frame before finish coding the current layer under the JSVM implementation.

Fig. 2.4 illustrates the flowchart of the mode decision algorithm for key and non-key frames in the MGS coding structure. The only difference between the key and non-key frame is the reference frame selection.

2.3 Constrained multilayer mode decision

2.3.1 Motivations and related works

With the conventional mode decision approach in CGS and MGS, at each layer, the mode decision including the motion search is optimized only for the current layer, using the reconstructed signal (from the same layer for CGS and MGS key frames, and from the highest layer for MGS non-key frames) in the previous frame as the reference, without considering whether the chosen mode could benefit the higher layers. Besides, the mode decision is conducted multiple times among all layer, thus the encoding complexity grows at least linearly as the number of layers.

To further improve the RD performance by exploiting the cross-layer correlation, a multilayer mode decision was proposed in [14]. Although it only considered the scenario containing two layers, it can be easily extended to an L-layered structure, where the best mode at all layers are determined jointly, i.e.

$$m_0^*, m_1^*, \ldots, m_L^* = \arg\min \sum_{i=0}^{L} w_i J_i, \tag{2.19}$$

where w_i is the weight for i-th layer, satisfying $\sum_{i=0}^{L} w_i = 1$. However, with this approach, the motion search is conducted multiple times even with the same layer. Due to its enormous complexity, it is impractical for multilayer coding with more than two layers. Even with the simplified approach presented in [15], the motion estimation is still required in every layer.

In quality scalability, since the all the layers are coded using the same frame at the same resolution, high correlation is expected between adjacent layers. As reported in [17], in a conventional SVC encoder, most of the collocated MBs are coded with the same mode (between the BL and 1st EL, over 70% of the collocated modes are determined to be the same, and between 1st EL and 2nd EL, 48%–63% are the same). This means that for most of the MBs at the EL, the motion search and the mode decision become

redundant. Moreover, if the mode decision at the BL is tuned toward the higher layers, more MBs will share the same optimal mode at different layers.

The massive amount of motion search in SVC and the correlation between adjacent layers inspire us to exploit the inter-layer mode and motion vector correlation as well as reduce the complexity. In the following sections, we will present a multilayer mode decision algorithm, where the MB mode is be jointly decided across the layers, with the motion estimation performed only once for all layers. The mode decision process does not need to be fully RD optimized, but should be near optimal.

2.3.2 Enforced inter-layer prediction

To force the EL to use the same mode as the lower layer, we make use of the macroblock type called *MB_Inferred*, which is defined in the SVC standard for inter-layer mode derivation (with the syntax *base mode flag* set to 1), where only the residual is coded, and other information is derived from the lower layer. This mode enables the EL to inherit the lower layer mode (including the MB partition, MV, etc.).

For different modes at the lower layer, the inferred mode is resolved to different prediction types, noted as the IntraBL and BLSkip mode receptively, as illustrated in Fig. 2.5.

If the lower layer collocated block is Inter-coded, then the current MB is also coded in an Inter-mode (noted as the BLSkip mode). In the BLSkip mode, the MB partition as well as the MV are derived from the lower layer, thus the computational expensive iteration for all the possible candidates is bypassed.

If the collocated MB in the lower layer is Intra-coded, the the Inferred mode is also coded in an Intra-mode (noted as the Intra-BL mode). Other than the conventional Intra-modes defined in the AVC (where the prediction is generated from neighboring block in the current layer using one of the 13 Intra-prediction types described in Sec. 1.2.4), the Intra-BL mode directly uses the lower layer reconstruction as the prediction. Note

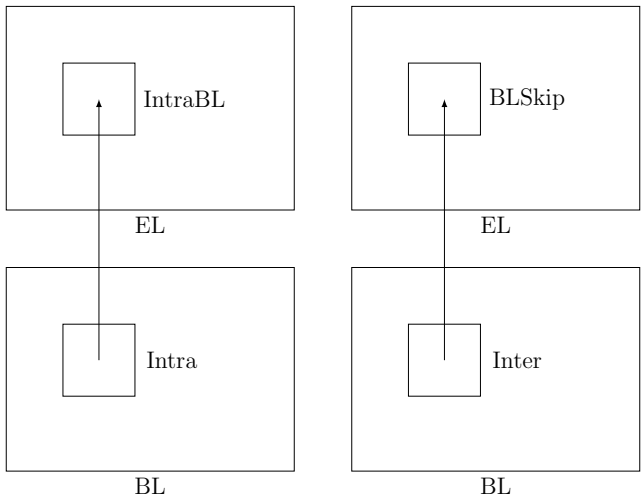


Figure 2.5: Illustration of inter-layer mode prediction for Intra-mode and Inter-mode of a two-layer structure.

that when the BL uses an Intra-mode, we do not force the EL to also use the same Intra-prediction, as there is no existing syntax in the H.264/SVC standard to support this. We choose to use the IntraBL mode, because it is simple and gives very good overall coding efficiency, whereas the EL is forced to use the same Inter-prediction when the BL is coded in the Inter-mode. The rationale for this choice is explained in Sec. 2.3.5.

2.3.3 Proposed multilayer mode decision for CGS

In the SVC encoder design, all blocks at the BL are encoded first, followed by the next higher layer, until all layers are encoded. Because of this bottom-up coding order, the joint mode decision must take place at the BL. During the mode decision, once the best MV or the Intra-prediction is determined for each sub-block, it will compete with other modes using (2.1). In this section, we discuss how to determine the reference frame and the Lagrangian parameters in (2.1) at the BL under CGS coding structures. The mode decision for the MGS structure will be presented in Sec. 2.3.4.

To tune the BL mode toward higher layers, we would like to use the reference frame

and the Lagrangian parameter applied at the highest layer as much as possible. In the conventional encoder, at the highest layer L, the best mode m_L^* is determined using the least quantized reference frame from L-th layer. Had this frame together with its corresponding Lagrangian multiplier λ_L been used at the BL, then the BL would choose a mode m_0' that is close to m_L^*, i.e., m_0' is near-optimal for L-th layer. For the motion search in the Inter-modes, we would like to obtain an as accurate MV as possible, thus we use L-th layer as reference, and L-th layer's QP is also used to derived the Lagrangian parameter λ_{Inter}. Similarly, for the Intra-prediction mode decision in the conventional encoder, we use the Lagrangian multiplier derived from QP_L guides the Intra-mode selection toward higher layers. The proposed motion search and Intra-prediction mode decision algorithms for the BL are

$$\hat{f} = \hat{f}_L, \quad \lambda_{\mathrm{MV}} = \lambda_{\mathrm{MV}}\left(\mathrm{QP}_L\right), \quad \lambda_{\mathrm{Intra}} = \lambda_{\mathrm{Intra}}\left(\mathrm{QP}_L\right). \tag{2.20}$$

i.e., the best MV at the BL is determined by

$$\begin{aligned} v_0^* &= \arg\min_{v_0} J_{\mathrm{Inter}}\left(v_0; f, \hat{f}_L, \lambda_{\mathrm{MV}}\left(\mathrm{QP}_L\right)\right) \\ &= \arg\min_{v_0}\left(D_{\mathrm{sub}}\left(f, \hat{f}_L\left(v_0\right)\right) + \lambda_{\mathrm{MV}}\left(\mathrm{QP}_L\right) R\left(v_0\right)\right), \end{aligned} \tag{2.21}$$

whereas the Intra-prediction is derived using (2.7) and (2.20).

However, our simulation shows a quite noticeable coding efficiency degradation at the BL if $\lambda\left(\mathrm{QP}_L\right)$ is use in mode decision. This is because in the RDO-based encoder, the Lagrangian multiplier at the BL is derived using QP_0. Had QP_L been used instead of QP_0, the chosen mode would be far from the optimal mode chosen by the RDO-based encoder. We have experimented with the Lagrangian multiplier coming from different layers, and found out that using the lowest layers QP to derive the Lagrangian multiplier provides good trade-off for coding efficiency at all layers, i.e.,

$$\lambda = \lambda\left(\mathrm{QP}_0\right), \tag{2.22}$$

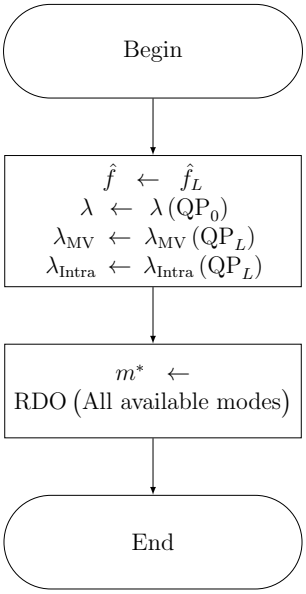


Figure 2.6: Proposed mode decision algorithm for the BL of CGS coding structure.

with the best mode derived using

$$\begin{aligned} m_0^* &= \arg\min_{m_0} J\left(m_0; f, \hat{f}_L, \lambda(\mathrm{QP}_0)\right) \\ &= \arg\min_{m_0} D\left(f, \hat{f}_L(m_0)\right) + \lambda(\mathrm{QP}_0)\, R\left(m_0, f - \hat{f}_L(m_0)\right), \end{aligned} \tag{2.23}$$

as illustrated in Fig. 2.6.

Note that to prevent the decoding mismatch, L-th layer is used only for prediction during the mode decision stage, while the reference still comes from the current layer during the coding stage.

After the mode is determined at the BL, it will be carried over to all higher layers, where the encoder simply chooses the BLSkip or IntraBL mode, depending on how the BL is coded.

2.3.4 Proposed multilayer mode decision for MGS

In the presented multilayer mode decision scheme for CGS, we use the same reference frame at the BL as the highest layer does (i.e., from the highest layer), so as to tune

the mode decision toward the highest layer. We apply the same strategy for reference frame selection for the MGS structure, i.e., the reference frame at the BL should be the one used by the highest layer. For the key frames, the highest layer uses the BL as the reference, thus in the proposed method, BL also uses the BL (i.e., the current layer) as the reference, with

$$\hat{f} = \hat{f}_0. \tag{2.24}$$

The Lagrangian parameters are also derived using the QP of the BL, i.e.,

$$\lambda = \lambda\left(\mathrm{QP}_0\right). \tag{2.25}$$

However, for the motion search and Intra-prediction mode decision use the highest layer:

$$.\lambda_{\mathrm{MV}} = \lambda_{\mathrm{MV}}\left(\mathrm{QP}_L\right), \quad \lambda_{\mathrm{Intra}} = \lambda_{\mathrm{Intra}}\left(\mathrm{QP}_L\right). \tag{2.26}$$

To summarize, the best MV at the BL is determined by

$$\begin{aligned} v_0^* &= \arg\min_{v_0} J_{\mathrm{Inter}}\left(v_0; f, \hat{f}_0, \lambda_{\mathrm{MV}}\left(\mathrm{QP}_L\right)\right) \\ &= \arg\min_{v_0}\left(D_{\mathrm{MV}}\left(f, \hat{f}_0\left(v_0\right)\right) + \lambda_{\mathrm{MV}}\left(\mathrm{QP}_0\right) R\left(v_0\right)\right), \end{aligned} \tag{2.27}$$

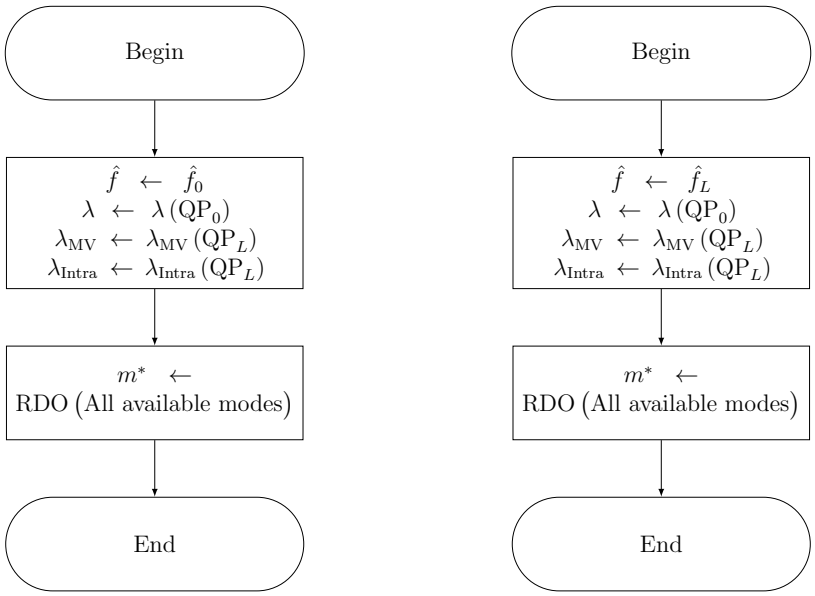
and the Intra-prediction is determined using (2.7) and (2.26). The best mode derived from

$$\begin{aligned} m_0^* &= \arg\min_{m_0} J\left(m_0; f, \hat{f}_0, \lambda\left(\mathrm{QP}_0\right)\right) \\ &= \arg\min_{m_0}\left(D\left(f, \hat{f}_0\left(m_0\right)\right) + \lambda\left(\mathrm{QP}_0\right) R\left(m_0, f - \hat{f}_0\left(m_0\right)\right)\right). \end{aligned} \tag{2.28}$$

For the non-key frames, the same motion estimation and mode decision scheme (2.10) and (2.11) has the same fashion as we proposed for the CGS. Intra-prediction is also determined using the same strategy, with (2.7) and (2.20).

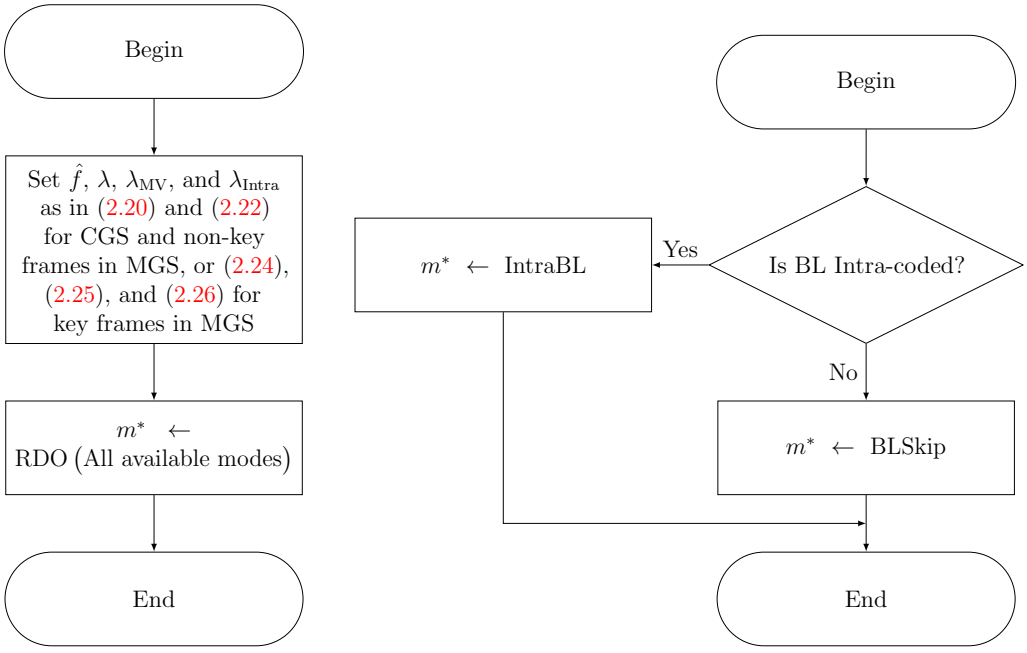
Note that for MGS, L-th layer is used as reference during both mode decision and encoding stages for non-key frames. Fig. 2.7 illustrate the flowcharts of the proposed mode decision algorithm for key and non-key frames at the BL in MGS coding structure.

To summarize the scenarios of CGS and MGS, the flowchart of proposed mode decision algorithm is illustrated in Fig. 2.8.



(a) Key frame (b) Non-key frame

Figure 2.7: Proposed mode decision algorithm for key frame and non-key frame in MGS coding structure.



(a) Base layer (b) Enhancement layer

Figure 2.8: Flowchart of proposed constrained mode decision at the BL and the EL. In the BL where the mode decision occurs, the highest layer is used as reference to tune the mode toward higher layers.

2.3.5 Discussion on inter-layer Intra-prediction

For the inter-layer Intra-prediction, one may have two options to inherit the lower layer Intra-mode:

1. Find a common Intra-prediction mode, and reuse it at all layers. More specifically, the BL determines the best Intra-prediction mode (taking higher layers into account), and the higher layers simply reuse the same prediction method.

2. Use the inter-layer Intra-prediction tool provided in H.264/SVC, i.e., BL determines the best Intra-mode (only for the BL), and the higher layers simply choose the IntraBL mode.

We have implemented both schemes in JSVM encoder, and the simulations results from the first option show quite noticeable coding efficiency degradation compared with using the second option. In this section, we discuss the reason why the first algorithm does not work well.

There are several reasons that the first strategy may degrade the coding performance. The first is that the inter-layer correlation is not well captured by the Intra-prediction modes, thus there may not exist a near-optimal mode for all layers. Since the higher layers are not available when coding the current layer, to determine the common Intra-mode, the original frame is required to be used as the reference frame. In the Intra-prediction modes, only the left and above boundary pixels are used for prediction, i.e., two 1-D arrays containing total of 13 pixels for the 4×4 partition size (as illustrated in Fig. 1.5) and 49 pixels for the 16×16 block size. However for the Inter-modes, with block size from 8×8 to 16×16, the motion field is captured by a 2-D block with 64 to 256 pixels. Compared to the Inter-prediction, the Intra-prediction between adjacent layers are prone to be affected by the quantization error, due to the reduced dimension and number of reference pixels.

Secondly, even a Intra-mode can be found as near-optimal for all layers, the correlations between the residuals are not well exploited. In a two-layer scenario, suppose the BL is quantized independently using the quantization stepsize q_0, and the EL using q_1, then the quantization error $\sigma_{q_0}^2$ at the BL and $\sigma_{q_1}^2$ at the EL are generally given by [25]

$$\sigma_{q_0}^2 = \epsilon^2 \sigma_0^2 2^{-2\lambda \tilde{R}_0}, \tag{2.29}$$

$$\sigma_{q_1}^2 = \epsilon^2 \sigma_1^2 2^{-2\lambda \tilde{R}_1}, \tag{2.30}$$

where σ^2 is the prediction error, and ϵ and λ are constants depending on the video sequence statistics and the encoder. A uniformly distributed signal has $\epsilon^2 = 1$ [25]. $\tilde{R}_0$ and $\tilde{R}_1$ are the number of bits to encode the residual at the BL and the EL respectively, which can be expressed by

$$\tilde{R}_0 = \frac{1}{2\lambda} \log_2 \left(\epsilon^2 \frac{\sigma^2}{\sigma_{q_0}^2} \right), \tag{2.31}$$

$$\tilde{R}_1 = \frac{1}{2\lambda} \log_2 \left(\epsilon^2 \frac{\sigma^2}{\sigma_{q_1}^2} \right). \tag{2.32}$$

For the IntraBL mode, at the EL, the quantization is operated on the BL reconstruction error $\sigma_{q_0}^2$ instead of σ^2. At the same quantization error as $\sigma_{q_1}^2$, the error produced by this re-quantization process is given by

$$\sigma_{q_1}^2 = \tilde{\epsilon}^2 \sigma_{q_0}^2 2^{-2\lambda \tilde{R}_\mathrm{I}}, \tag{2.33}$$

where $\tilde{R}_\mathrm{I}$ represents the number of bits to encode the residual in IntralBL mode, which is

$$\begin{aligned} \tilde{R}_\mathrm{I} &= \frac{1}{2\lambda} \log_2 \left(\tilde{\epsilon}^2 \frac{\sigma_{q_0}^2}{\sigma_{q_1}^2} \right) \\ &= \frac{1}{2\lambda} \log_2 \tilde{\epsilon}^2 + \frac{1}{2\lambda} \log_2 \frac{\sigma_{q_0}^2}{\sigma_{q_1}^2} \\ &= \frac{1}{2\lambda} \log_2 \tilde{\epsilon}^2 + \tilde{R}_1 - \tilde{R}_0. \end{aligned} \tag{2.34}$$

Here the constant $\tilde{\epsilon}$ depends on the distribution of the residual. For a well-designed quantizer, the residual has a near-uniform distribution, i.e., $\tilde{\epsilon}^2 \approx 1$, which leads to

$$\tilde{R}_\mathrm{I} \approx \tilde{R}_1 - \tilde{R}_0. \tag{2.35}$$

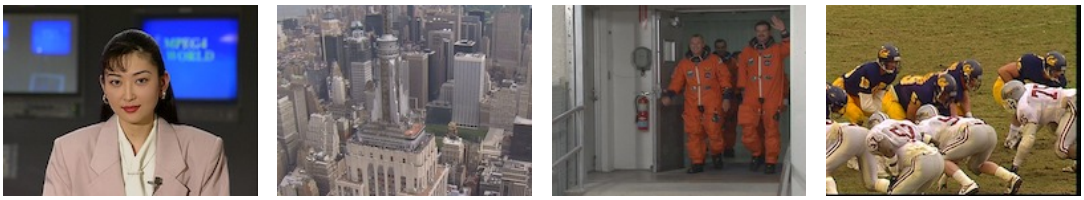
Compared to $\tilde{R}_1$ that is derived using the lower layer Intra-mode, the rate yield by the IntraBL mode (2.35) has the inter-layer correlation eliminated natively, while the decorrelation between (2.29) and (2.30) relies on the residual prediction process. Since the IntraBL mode could also benefit from the adaptive residual prediction, the adaptive residual prediction contributes to the coding efficiency gain over using the lower layer Intra-mode.

Another reason is due to the header bits consumed in encoding the syntax elements for mode indication. Since the mode reusing the lower layer Intra-mode is not provided in the H.264 standard, one must encode the reused lower layer mode explicitly, while the IntraBL mode only carries a one-bit flag to specify the inherited mode.

2.4 Performance evaluation and discussions

2.4.1 Simulation configurations

Seven video sequence with CIF (352×288) resolution and three HD sequences with 720p (1280×720) resolution are chosen from the Joint Video Team [26] test sequences pool, as



(a) akiyo (b) city (c) crew (d) football

(e) foreman (f) ice (g) waterfall

Figure 2.9: CIF resolution test sequences

(a) mobcal (b) parkrun (c) shields

Figure 2.10: 720p resolution test sequences

Table 2.1: QP configuration for different content

Resolution	Sequence	Frames	Layer	QP			
CIF	akiyo ice	289	0	36	40	44	48
			1	30	34	38	42
			2	24	28	32	36
	city crew foreman	289	0	30	34	38	42
			1	24	28	32	36
			2	18	22	26	30
	football waterfall	257	0	30	34	38	42
			1	24	28	32	36
			2	18	22	26	30
720p	mobcal parkrun	497	0	30	34	38	42
			1	24	28	32	36
	shelds	289	2	18	22	26	30

illustrated in Fig. 2.9 and 2.10. These sequences are encoded with three CGS and MGS layers respectively, using the latest reference software JSVM 9.19.15 [12] implemented with the proposed algorithm. The original JSVM 9.19.15 with RDO enabled is also evaluated for benchmark comparison. Adaptive residual prediction is enabled for both encoders, and the number of reference frames is constrained to one. The QP difference between adjacent quality layers is fixed to be 6. The QPs are chosen to cover a wide range while providing reasonable perceptual quality for all layers (with PSNR between 28 and 42), as detailed in Table 2.1.

The hierarchical B-frame structure is applied to support temporal scalability. GOP

length of 16 is used for the CGS coding structure. For the MGS structure, since the highest layer is used as the reference in the decoder, due to the limitation in the decoding buffer, the GOP length is set to 8. I-frames are inserted at a period of 64 frames, for both CGS and MGS configurations.

Even though AVC and SVC support the block partition size for Inter-mode from 16x16 down to 4x4, according to our experiments, we have noticed that coding efficiency is degraded less than 1% (in terms of BD-Rate) by disabling block size less than 8×8 in Inter-modes, but with quite significant 25% encoder complexity reduction compared with the default JSVM encoding. This is also confirmed during the High-efficiency video coding (HEVC) standardization that smaller block size (less than 8×8) does not provide significant coding efficiency improvement for Inter-frames but with dramatic overhead for memory access and computing. Hence, 4×4 block based motion compensation is not used in H.265/HEVC [8]. In this work, we also do not consider the block partitions smaller than 8×8 in Inter-modes, in both original JSVM and the one implemented with the proposed algorithm.

In SVC, the inter-layer motion prediction (ILMP) allows the EL to derive the PMV using the MV of collocated MB in the lower layer instead from current layer. A fully RDO based encoder adopts adaptive ILMP, i.e., the ME is conducted twice at each EL, with and without ILMP. In our experiments, we have noticed that the coding efficiency gain brought by ILMP is marginal, as also reported by Li *et. al* [27]. Moreover, in our proposed algorithm, as the highest layer is used as reference, the MV obtained in BL captures the actual motion quite well. Hence we enforce the EL to apply ILMP, i.e, the PMV always comes from the lower layer. It is also applied in the original JSVM for fair comparison.

The experiments are conducted on a Linux desktop server equipped with Intel Xeon (E5405@2.00GHz) processor and 8GB memory, running Ubuntu 12.04 server edition. Each individual encoding process is executed exclusively, without interfering with other

running programs. The relative reduction of total encoding time ΔT (for all layers) is defined as

$$\Delta T = \frac{T_{\text{JSVM}} - T_{\text{Prop}}}{T_{\text{JSVM}}} \times 100\%, \tag{2.36}$$

averaged over all the QPs, where T_{JSVM} and T_{Prop} are the total encoding time for the default JSVM and proposed constrained low-complexity algorithm, respectively, and measured using the timing function provided by the operating system. ΔT_m is the reduction of time in mode decision (including the motion search, transform, and quantization) at each layer, which is derived in the similar manner, i.e.,

$$\Delta T_m = \frac{T_{m,\text{JSVM}} - T_{m,\text{Prop}}}{T_{m\text{JSVM}}} \times 100\%, \tag{2.37}$$

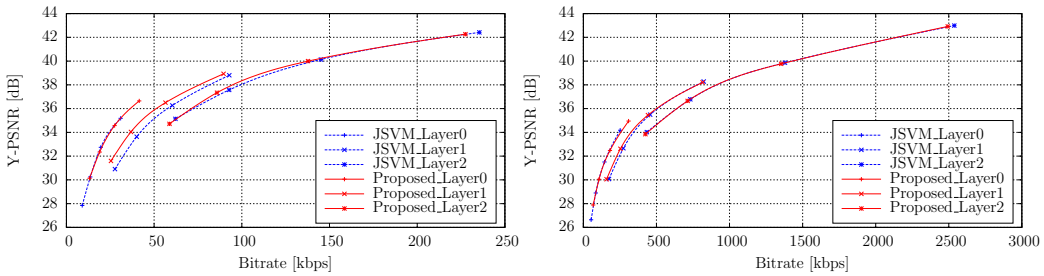
with $T_{m,\text{JSVM}}$ and $T_{m,\text{Prop}}$ being the time spent on mode decision in the JSVM and the one implemented with the proposed algorithm, respectively. Note that for each block, the time consumed in mode decision could be lower than the minimum precision provided by the system timing function, thus we measure the CPU cycle count, and convert it back to time using 2.00 GHz frequency (the CPU frequency is fixed to 2.00 GHz when running the simulations).

2.4.2 Evaluation under the CGS coding structure

Fig. 2.11 plots the coding efficiency evaluation for the CIF test sequences using CGS structure, with ESD mode disabled. As can be observed in the figures, the proposed mode decision algorithm has better R-D performance than the JSVM encoder. This gain is contributed by using the top layer as reference, and the cross-layer mode decision.

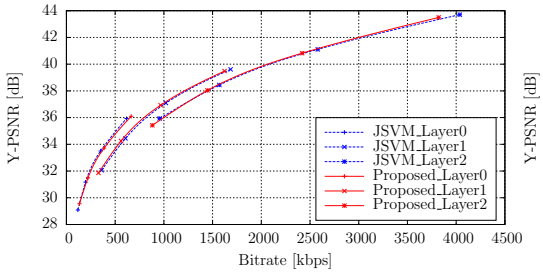
The encoding complexity is measured in terms of the encoding time. The overall time of CIF test sequences for encoding all three layers is shown in Fig. 2.12. For all the sequences, the proposed method achieves almost 50% saving in the total encoding time.

A more detailed inspection of the encoding complexly consumed in mode decision (including the motion estimation) at each layer is illustrated in Fig. 2.13. As expected,

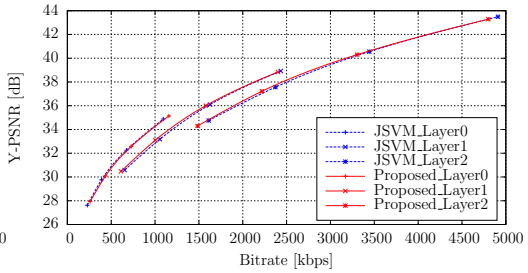


(a) akiyo

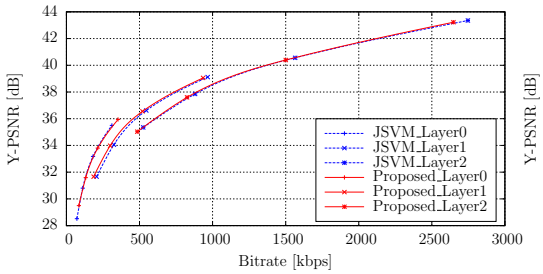
(b) city



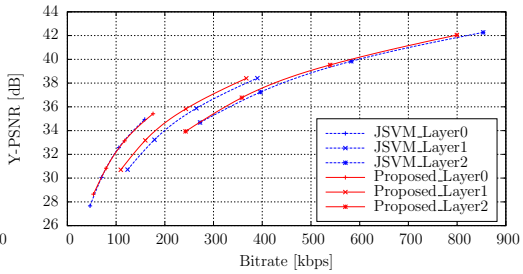
(c) crew



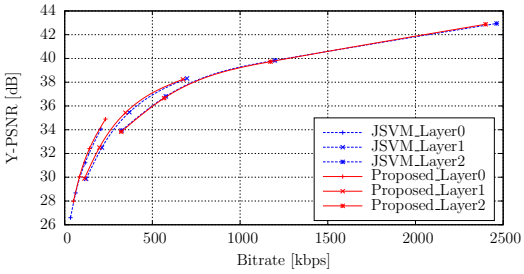
(d) football



(e) foreman

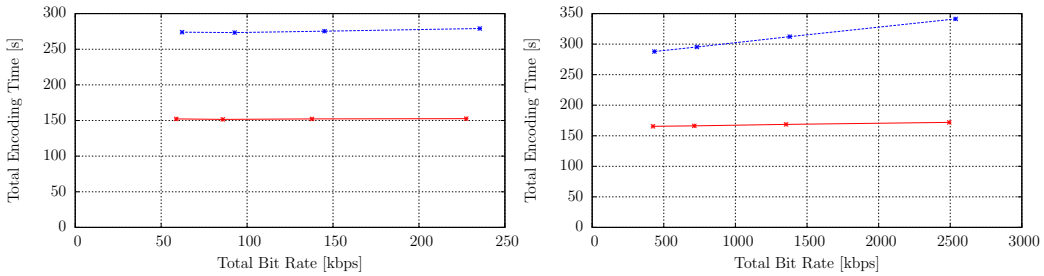


(f) ice



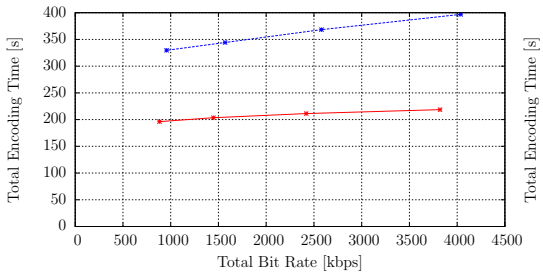
(g) waterfall

Figure 2.11: Performance comparison for coding efficiency of proposed algorithm v.s. default JSVM of CIF test sequences using CGS coding structure.

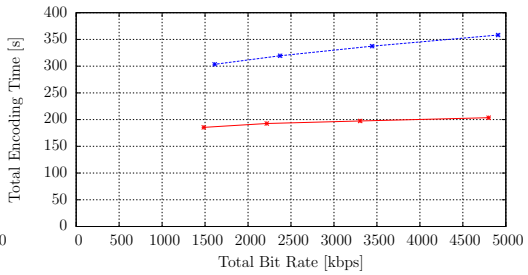


(a) akiyo

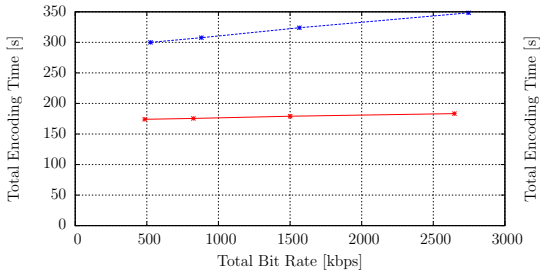
(b) city



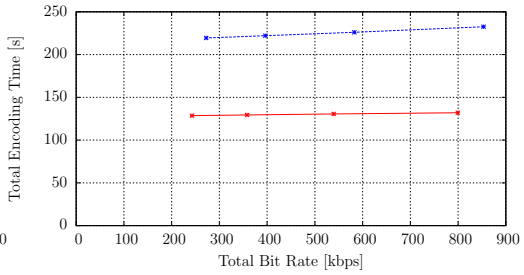
(c) crew



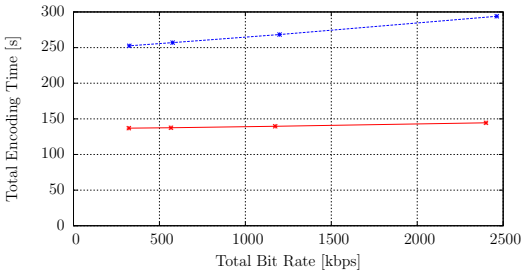
(d) football



(e) foreman

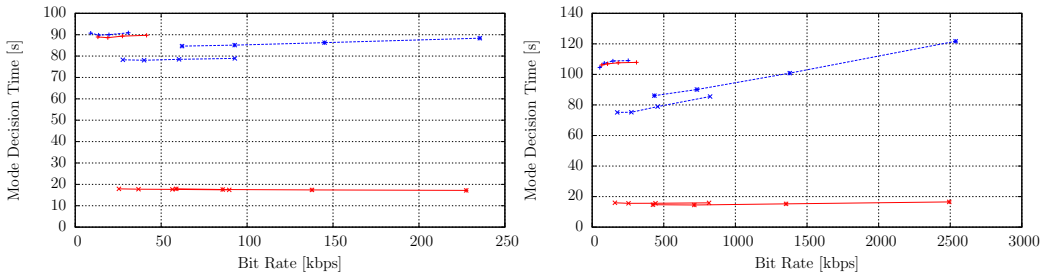


(f) ice



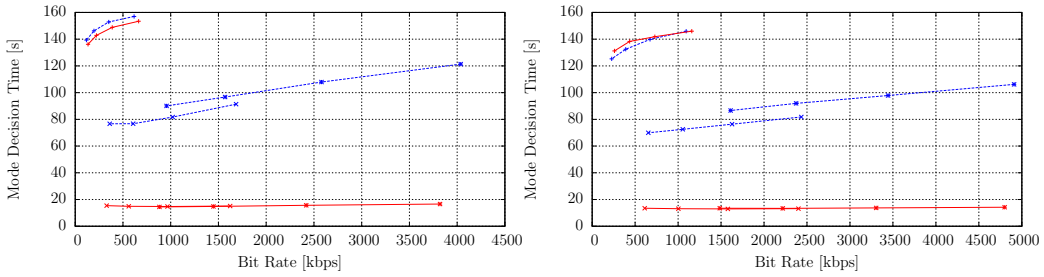
(g) waterfall

Figure 2.12: Performance comparison for total encoding time of proposed algorithm v.s. default JSVM of CIF test sequences using CGS coding structure.



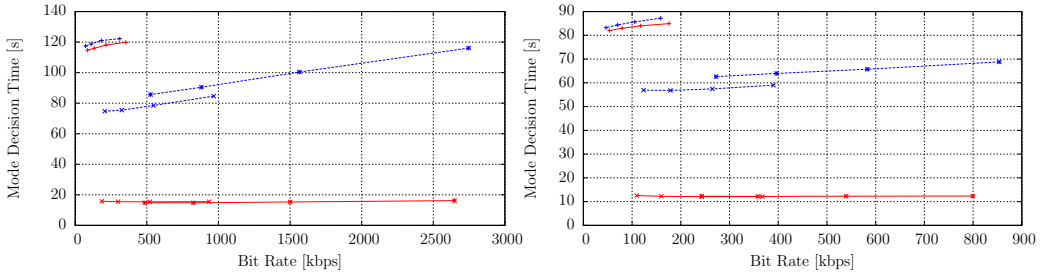
(a) akiyo

(b) city



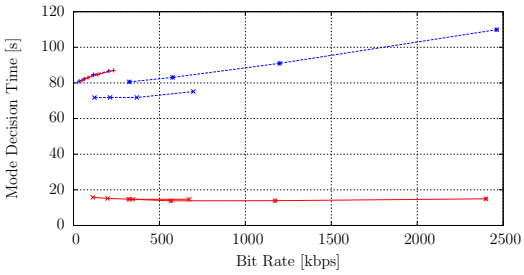
(c) crew

(d) football



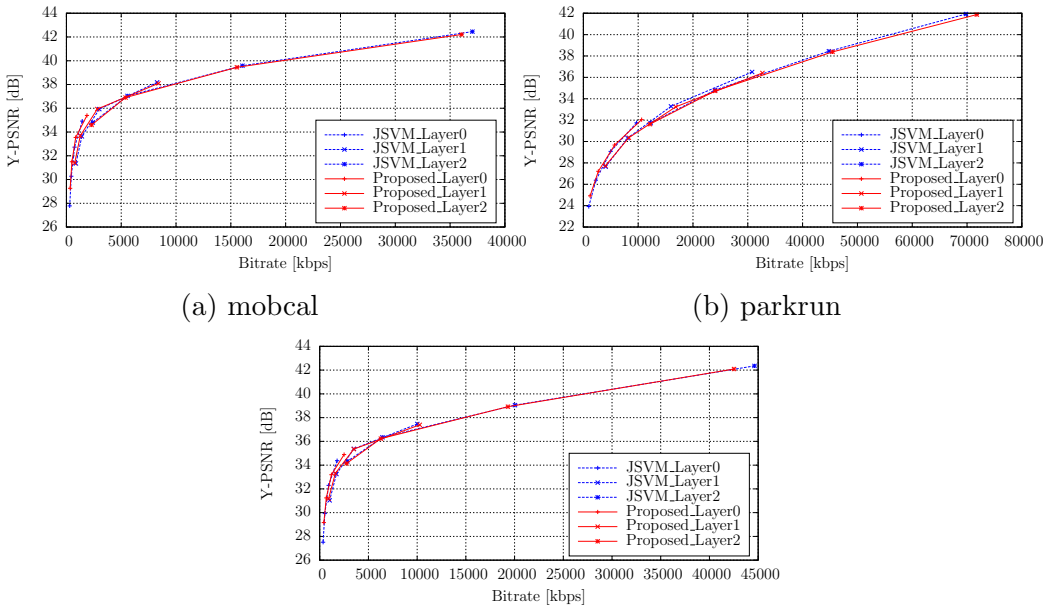
(e) foreman

(f) ice



(g) waterfall

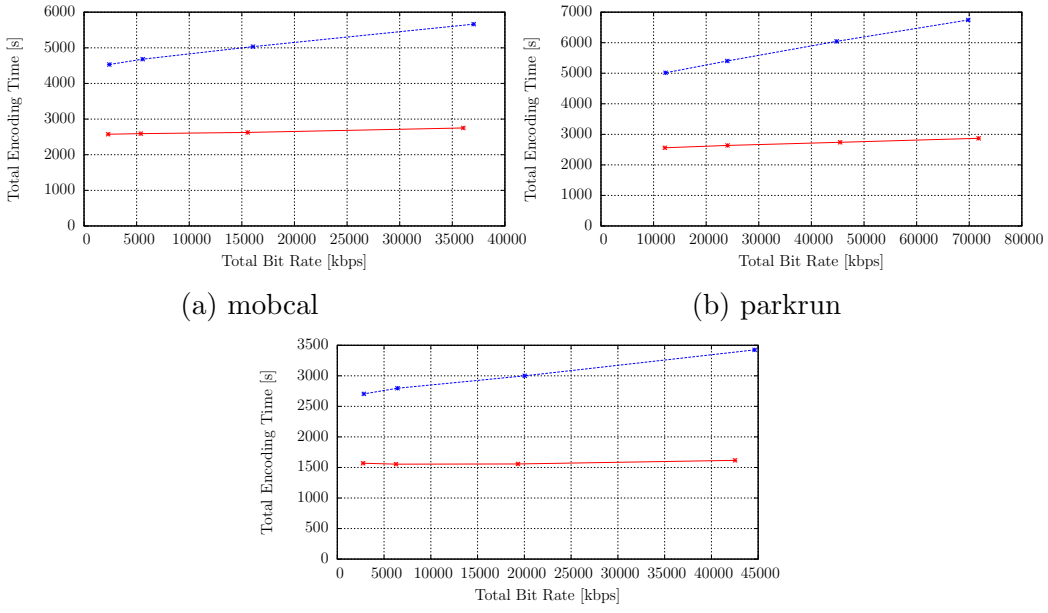
Figure 2.13: Performance comparison for mode decision (including motion estimation) time of proposed algorithm v.s. default JSVM of CIF test sequences using CGS coding structure.



(a) mobcal (b) parkrun

(c) shields

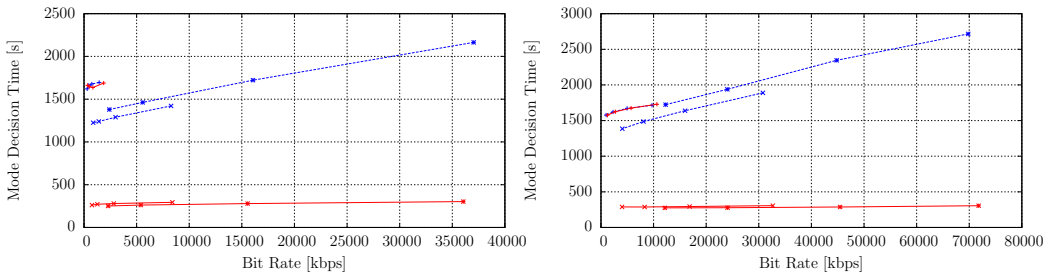
Figure 2.14: Performance comparison for coding efficiency of proposed algorithm v.s. default JSVM of 720p test sequences using CGS coding structure.



(a) mobcal (b) parkrun

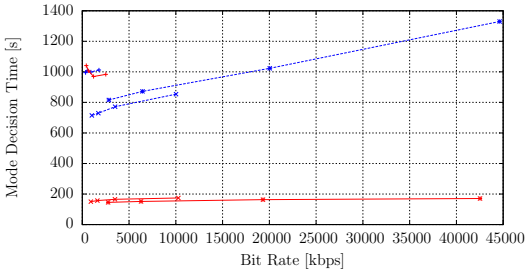
(c) shields

Figure 2.15: Performance comparison for total encoding time of proposed algorithm v.s. default JSVM of 720p test sequences using CGS coding structure.



(a) mobcal

(b) parkrun



(c) shields

Figure 2.16: Performance comparison for mode decision (including motion estimation) time of proposed algorithm v.s. default JSVM of 720p test sequences using CGS coding structure.

with our proposed method, the mode decision time at the EL is reduced significantly and remains almost constant, and appears sequence independent. However at the BL, the complexity is almost the same.

It is noticed that in the default JSVM, the EL takes less time to encode than the BL. This is due to the forced ILMP together with the fast motion search, where the MV from the lower layer is used as PMV, resulting the motion search engine terminates at an early stage. More complexity saving is expected if adaptive ILMP is enabled or the fast motion search is disabled.

The complete simulation results for the CIF test sequences are detailed in Table 2.2. As the experiments show, our proposed multilayer mode decision algorithm achieves an average 43.9% total time reduction for encoding all three layers, with average 0.7%, 79.9%, and 83.5% time reduction for mode decision at the BL, EL #1 and EL #2, respectively.

Table 2.2: Performance Evaluation of Proposed Algorithm for CIF using CGS

Sequence	Layer	BD-Rate	ΔT	ΔT_m
akiyo	0	1.2%	44.8%	1.4%
	1	−11.6%		77.5%
	2	−2.4%		79.6%
city	0	1.0%	45.5%	0.3%
	1	−4.4%		79.9%
	2	0.1%		84.6%
crew	0	4.1%	42.2%	
	1	−2.9%		81.5%
	2	−0.7%		85.1%
football	0	2.7%	40.8%	−2.6%
	1	−2.5%		82.4%
	2	−1.3%		85.6%
foreman	0	1.5%	44.3%	2.3%
	1	−5.1%		80.1%
	2	−1.0%		84.3%
ice	0	0.0%	42.1%	1.9%
	1	−8.6%		78.7%
	2	−2.9%		81.2%
waterfall	0	−5.0%	47.8%	−0.5%
	1	−5.7%		79.2%
	2	0.7%		84.0%
Average	0	0.8%	43.9%	0.7%
	1	−5.8%		79.9%
	2	−1.1%		83.5%

Table 2.3: Performance Evaluation of Proposed Algorithm for 720p using CGS

Sequence	Layer	BD-Rate	ΔT	ΔT_m
mobcal	0	−9.0%	46.8%	0.3%
	1	−7.9%		78.6%
	2	2.2%		83.4%
parkrun	0	−5.1%	53.1%	−0.3%
	1	3.9%		81.4%
	2	2.2%		86.6%
shields	0	−2.1%	46.8%	0.3%
	1	−3.9%		78.9%
	2	1.4%		84.0%
Average	0	−5.4%	48.9%	0.1%
	1	−2.6%		79.6%
	2	1.9%		84.7%

Recall that the motion estimation is conducted at the BL using the finest reconstruction as reference instead of from the current layer. This leads to average 0.8% BD-Rate increment at the BL. However, the higher layers benefit from the mode chosen at the BL, resulting an average BD-Rate reduction of 5.8% and 1.1% at EL #1 and EL #2, respectively. Note that the bitrate measured at EL has its all lower layers included.

Reported above are the simulation results for the CIF test sequences. The performance evaluation for 720p test sequences are shown in Fig. 2.14, 2.15, and 2.16, in terms of the coding efficiency, total encoding time, and the mode decision time for each layer, respectively. Like the CIF sequences, the proposed algorithm also works rather well on the 720p sequences, as it can be easily observed with almost the same coding efficiency, a near 50% saving of total encoding time, and significant saving on the mode decision time at the ELs, compared with the convention method in the JSVM encoder.

The detailed experiment results and comparisons are listed in Table 2.3. With the proposed algorithm, the coding efficiency is similar to or better than the JSVM software, with the average 48.9% saving for the total encoding time. At each layer, the time saving

percentages in mode decision are 0.1%, 79.6%, and 84.7%, respectively.

2.4.3 Evaluation under the MGS coding structure

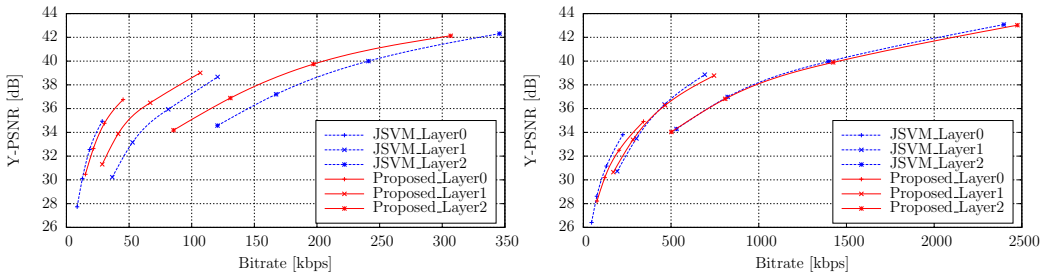
The performance evaluation results for the CIF sequences using the MGS coding structure are visualized in Fig. 2.17, 2.18, and 2.19, reporting the coding efficiency, the total encoding time, and the time consumed by mode decision, respectively.

Since the top layer is already used as reference in MGS, the proposed method is expected to have certain level of coding efficiency degradation compared to the conventional algorithm. However, as shown in Fig. 2.17, the sequences *akiyo* and *ice* have huge coding efficiency gain compared with the original JSVM. Our investigation shows that the contribution is from the MGS key frames. Since these two sequences has almost stationary background, most of the bits are consumed in coding the key frames. While the JSVM encoder is optimized only for the current layer, our proposed cross-layer mode decision algorithm provides much higher coding efficiency at the ELs, by tune the motion search and Intra-prediction mode decision toward higher layers, and forcing all layers to uses the same mode.

As for the complexity reduction shown in Fig. 2.18, the trend for MGS structure is similar to that using CGS coding structure. The overall encoding time is reduced by almost half, and significant time reduction for mode decision is observed. The mode decision time at the ELs is also reduced by a significant amount, as shown in Fig. 2.19.

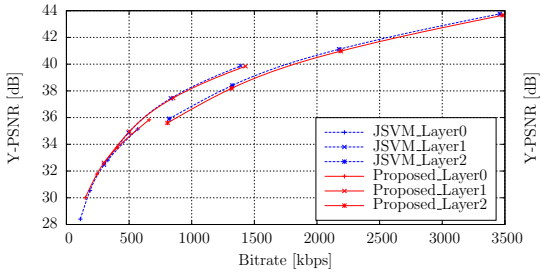
The complete results for the CIF test sequences are listed in Table. 2.4. The bitrate at the BL has an average overhead of 5.6%, but at EL #1 and #2, our method achieves average 6.0% and 1.4% bitrate reduction respectively. The total encoding time is saved by 42.7% on average, with near 80% saving on the mode decision at the ELs.

For the HD sequences at 720p resolution, the performance evaluation in terms of coding efficiency, total encoding time, and mode decision time are demonstrated in Fig. 2.20, 2.21, and 2.22, respectively, and the complete results are reported in Table 2.5.

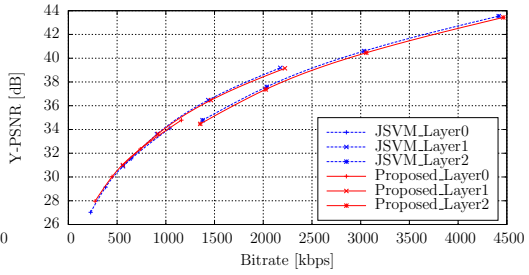


(a) akiyo

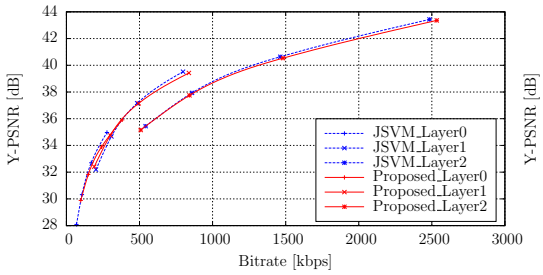
(b) city



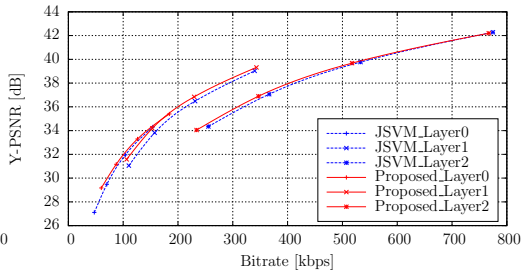
(c) crew



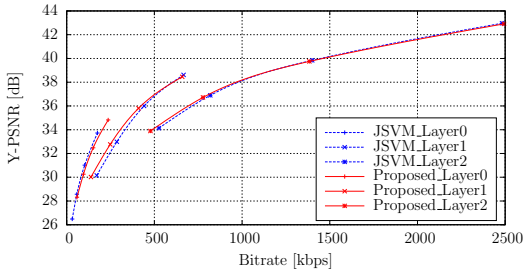
(d) football



(e) foreman

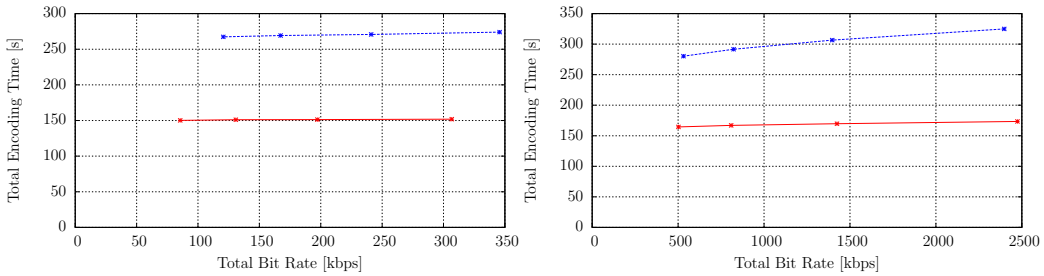


(f) ice



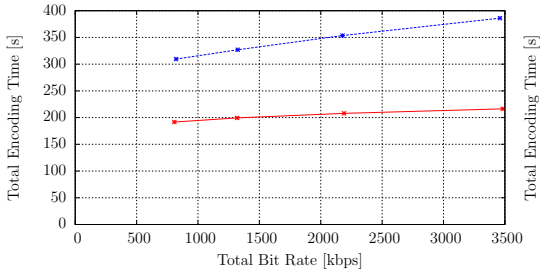
(g) waterfall

Figure 2.17: Performance comparison for coding efficiency of proposed algorithm v.s. default JSVM of CIF test sequences using MGS coding structure.

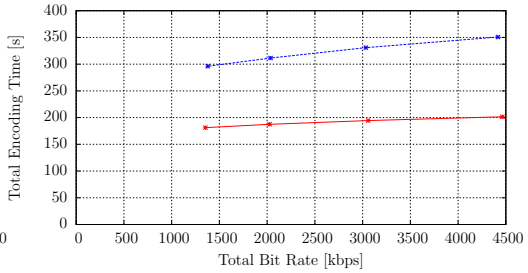


(a) akiyo

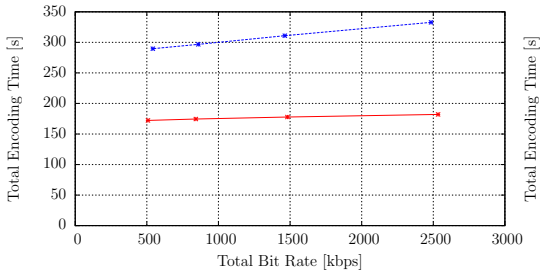
(b) city



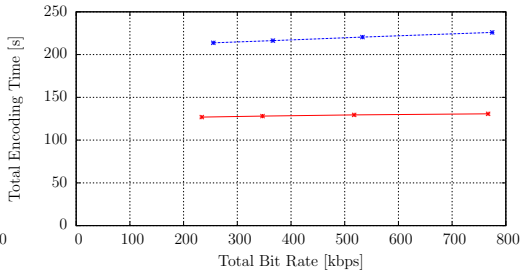
(c) crew



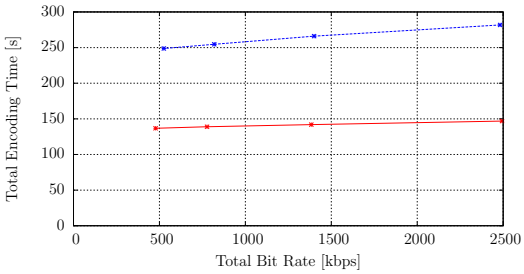
(d) football



(e) foreman

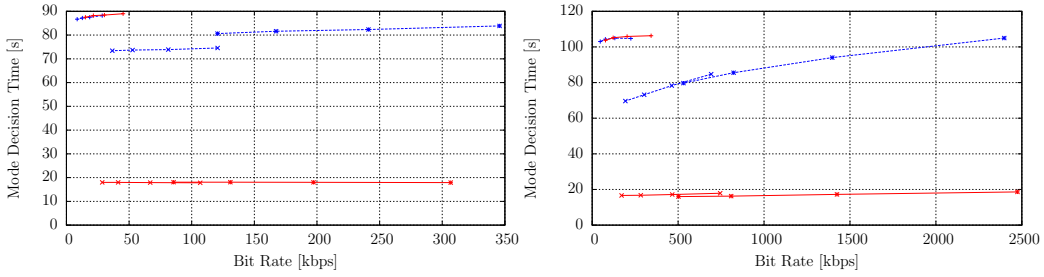


(f) ice



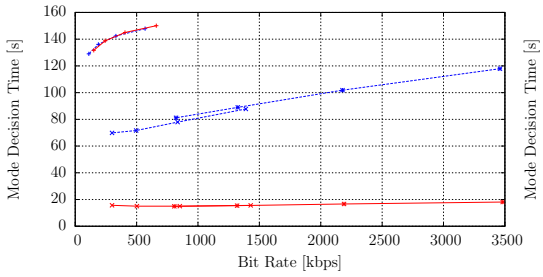
(g) waterfall

Figure 2.18: Performance comparison for total encoding time of proposed algorithm v.s. default JSVM of CIF test sequences using MGS coding structure.

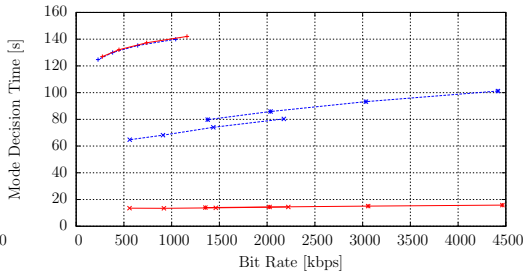


(a) akiyo

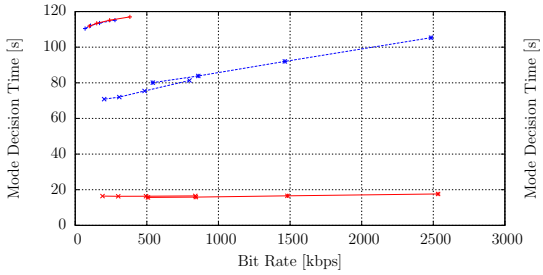
(b) city



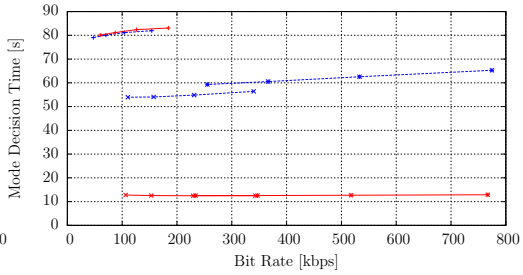
(c) crew



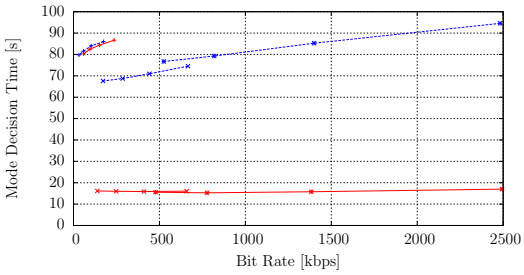
(d) football



(e) foreman

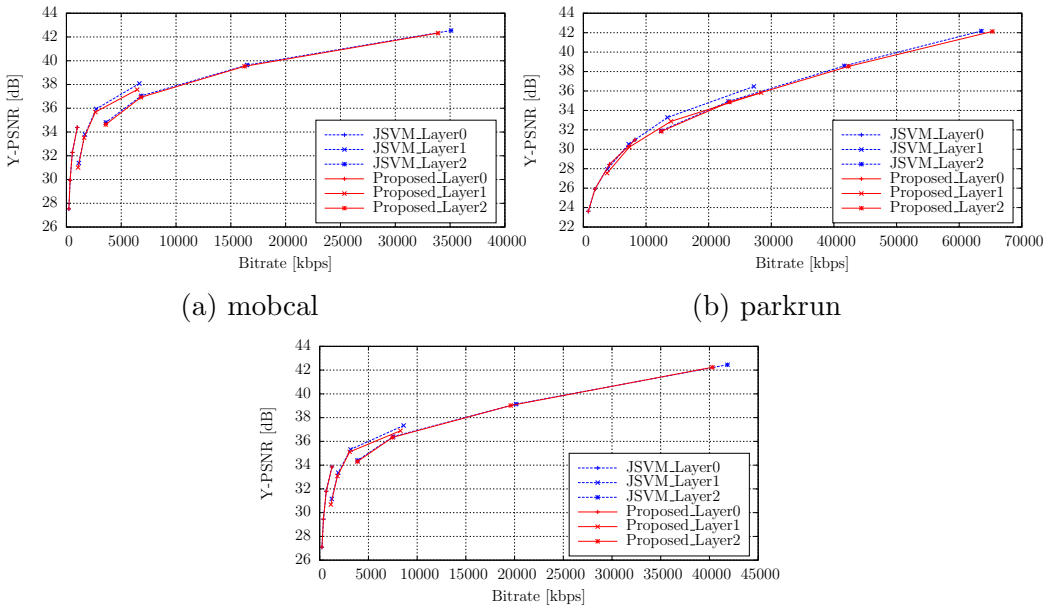


(f) ice



(g) waterfall

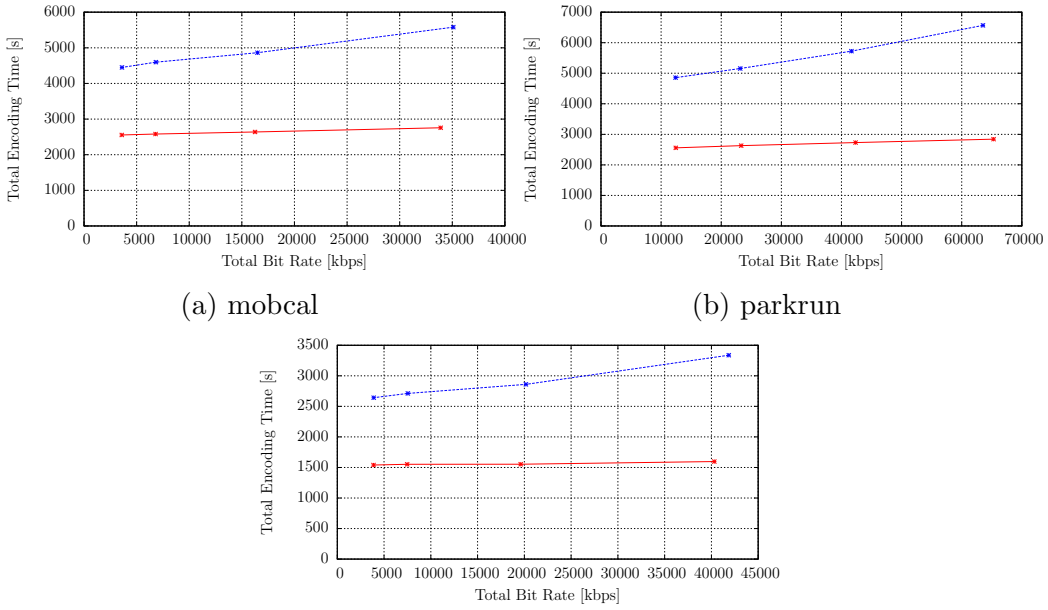
Figure 2.19: Performance comparison for mode decision (including motion estimation) time of proposed algorithm v.s. default JSVM of CIF test sequences using MGS coding structure.



(a) mobcal (b) parkrun

(c) shields

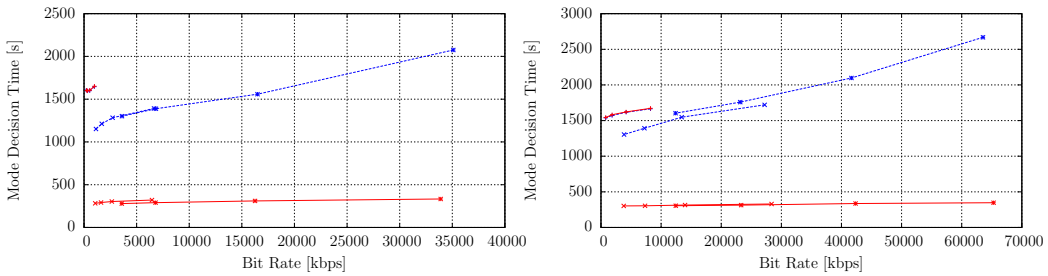
Figure 2.20: Performance comparison for coding efficiency of proposed algorithm v.s. default JSVM of 720p test sequences using MGS coding structure.



(a) mobcal (b) parkrun

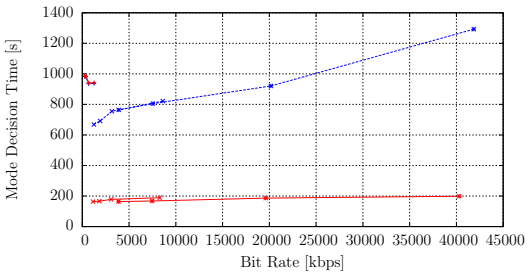
(c) shields

Figure 2.21: Performance comparison for total encoding time of proposed algorithm v.s. default JSVM of 720p test sequences using MGS coding structure.



(a) mobcal

(b) parkrun



(c) shields

Figure 2.22: Performance comparison for mode decision (including motion estimation) time of proposed algorithm v.s. default JSVM of 720p test sequences using MGS coding structure.

For the coding efficiency, slight degradation (up to 7.4% BD-Rate increment) is observed at the middle layer on average, but the overall encoding time saving reaches 47.5% on average, with average 77.1% and 81.5% saving in the mode decision time at EL #1 and #2.

Table 2.4: Performance Evaluation of Proposed Algorithm for CIF using MGS

Sequence	Layer	BD-Rate	ΔT	ΔT_m
akiyo	0	12.9%	44.1%	−1.0%
	1	−26.7%		75.7%
	2	−16.9%		78.0%
city	0	14.8%	43.8%	−0.9%
	1	−1.6%		77.5%
	2	2.1%		81.2%
crew	0	−4.0%	40.5%	−1.8%
	1	0.5%		79.9%
	2	3.4%		83.1%
football	0	−3.3%	40.6%	−1.6%
	1	1.1%		80.7%
	2	2.6%		83.5%
foreman	0	6.3%	42.5%	−1.3%
	1	−1.0%		78.0%
	2	2.3%		81.7%
ice	0	−4.8%	41.2%	−1.4%
	1	−6.9%		77.1%
	2	−2.4%		79.5%
waterfall	0	9.5%	46.2%	−0.7%
	1	−7.4%		77.2%
	2	−1.0%		81.0%
Average	0	4.5%	42.7%	−1.3%
	1	−6.0%		78.0%
	2	−1.4%		81.2%

Table 2.5: Performance Evaluation of Proposed Algorithm for 720p using MGS

Sequence	Layer	BD-Rate	ΔT	ΔT_m
mobcal	0	0.0%	45.7%	−0.1%
	1	4.7%		76.2%
	2	2.6%		80.4%
parkrun	0	0.0%	51.3%	−0.4%
	1	12.5%		78.9%
	2	2.3%		83.5%
shields	0	0.1%	45.6%	−0.5%
	1	5.0%		76.2%
	2	1.0%		80.5%
Average	0	0.0%	47.5%	−0.3%
	1	7.4%		77.1%
	2	2.0%		81.5%

2.5 Summary and discussions

In this chapter, we first investigate the mode decision algorithm used by the conventional SVC encoder. Then we propose a novel multilayer mode decision scheme by exploiting the cross-layer correlation. In our method, the joint mode decision takes place at the bottom layer, while considering the higher layers by using the highest layer as reference and sometimes using the highest layer QP to derive the Lagrangian parameter. Once the best mode has been determined at the BL, this mode is then reused by the collocated blocks in all higher layers, thus the higher layers are exempt from the computationally intensive motion search and mode decision. The experimental results show that the proposed algorithm achieves slightly worse and sometimes better overall coding efficiency, but significant complexity savings at the enhancement layers, compared to the conventional mode decision method. For the base layer, the proposed method has the same complexity as the conventional method. The complexity reduction for the base layer will be presented in Chapter 3.

Chapter 3

Early Skip/Direct mode decision for AVC and SVC

In this chapter, we investigate a low-complexity mode decision technique called early skip. By including our unified Direct mode, we extend the early skip technique to the early Skip/Direct (ESD) mode decision. With this method, when certain conditions are satisfied, the RDO-based mode decision is bypassed. The proposed ESD conditions are based on only a few thresholds that require light-weighted comparisons. We also present a systematic approach to derive the ESD thresholds. When combined with the multilayer mode decision presented in Chapter 2, the mode decision algorithm at the EL is modified to enable the light-weighted motion search (only at 16×16 block size) for blocks where the motion estimation (ME) has not been conducted in lower layers. Overall, the ME is conducted at most once among all layers. The simulation results demonstrate slight coding efficiency degradation, but significant complexity saving.

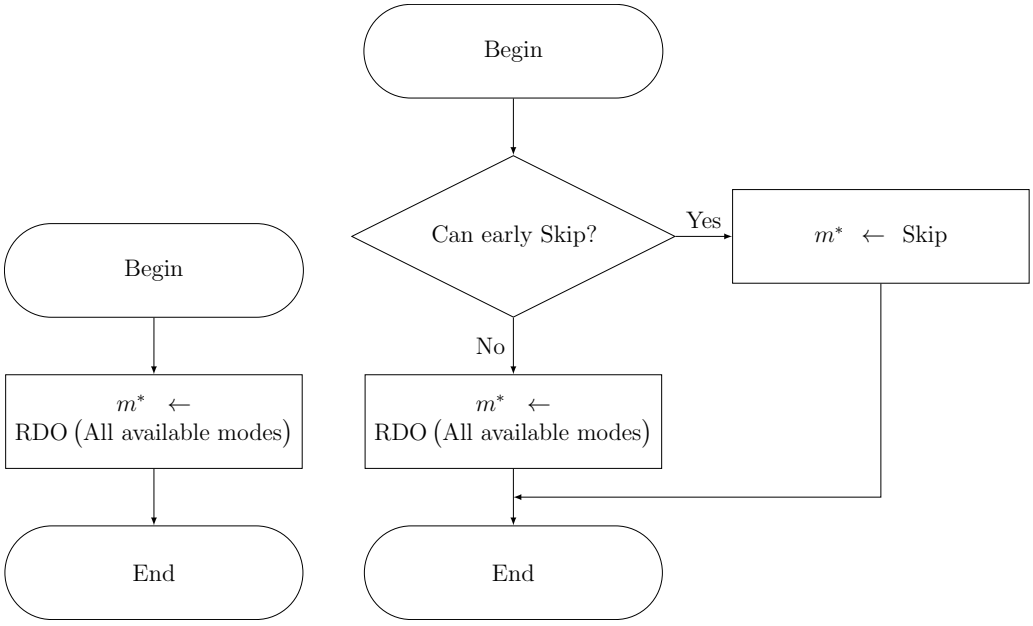
3.1 Motivation and related works

In spite that various fast motion estimation algorithms have been developed (for example, the TZ-Search introduced in JSVM [12]), the motion search module is still the major factor of the high complexity in the encoder. In SVC, by exploiting the inter-layer correlation in our proposed mode decision algorithm described in Chapter 2, the complexity for motion estimation and mode decision at ELs has been reduced to a negligible level, however the BL is still responsible for the motion search and requires heavy computation. In this section, we summarize the techniques to reduce the encoder complexity for AVC and SVC.

In the RDO-based encoder, the distortion is measured in terms of Sum of Squared Error (SSE), which is computational expensive. In H.264 reference software JM [28] an error metric called Sum of Absolute Transformed Difference (SATD) was introduced, and widely used in a number of low-complexity mode decision algorithms.

A low-complexity mode decision technique for H.264/AVC called early skip was introduced by Jeon *et. al* [29], where under certain conditions, a special Inter-mode called the Skip mode is selected without evaluating all the possible modes using RDO approach, as demonstrated in Fig. 3.1.

This approach has brought vast interest in H.264 video coding, and numerous early skip conditions have been designed. In [30], the motion filed is analyzed, and a statistics model is proposed to guide the mode selection. In [31], the Lagrangian multiplier is modeled to assist the early skip decision. In [32], the temporal correlation between frames is utilized in the early skip threshold derivation. [33] presents three methods for the early skip decision, using the ρ-domain rate model, the spatial-temporal prediction, and the restricted reference frame. For quality scalability in SVC, the early skip is also studied in [34], where the lower layer information is used to assist the early skip decision at CGS enhancement layers.



(a) Without early skip (b) With early skip

Figure 3.1: Demonstration for the mode decision algorithm without and with early skip technique.

These approaches [30, 31, 32, 33, 34] rely on either multiple thresholds, or multiple motion compensations for comparison, and some also requires the storage of historical data. In this chapter, we present a simple yet effective early Skip/Direct (ESD) mode decision scheme by extending the early termination technique to include the Direct mode, which requires only one motion compensation, and single threshold for the Skip and Direct mode. It can be applied for both AVC and SVC.

3.2 Proposed Early Skip/Direct mode decision for AVC

3.2.1 Generalized Direct mode

Among all the Inter-prediction modes in AVC, the Skip and Direct modes are two special macroblock types that do not require the motion estimation.

The Skip mode is available in P- and B-frames (noted as P_SKIP and B_SKIP macroblock type in the H.264/AVC standard). In the coded bitstream, the Skip mode is signaled by a one-bit flag, with other information (e.g., reference frame, MB partition, MV, etc.) derived by the rules specified in the H.264 standard. For the Skip mode, the MB partition is set to 16×16, and the PMV is used as MV, with no residual is coded.

The Direct mode is only available in B-frames (noted as B_Direct in H.264/AVC). Similar to the Skip mode, Direct mode also derives the MV directly from PMV, but the residual is coded [35]. Another difference is that the Direct mode can be partitioned into small blocks, while the Skip mode can only use the 16×16 block size.

The idea of introducing the Direct mode for P-frames was investigated by Tourapis *et. al* [36]. In this work, we unify the Skip mode and Direct mode in P- and B-frames, by extending the Direct mode to the P-frames. We internally create a so-called P_Direct mode, with its block size fixed to 16×16 as in the Skip mode. Since this mode is not specified in the standard, to produce a standard-compliant bitstream, the 16×16 Inter-mode syntax is used to represent this mode. Like the Skip mode, this mode also uses the 16×16 MB partition, and use the PMV as MV. In addition, the 16×16 Inter-mode enables the residual to be coded as in B_Direct mode.

With the unified Direct mode, the early skip technique can be applied on the Direct mode in both P- and B-frames, which is denoted as early Skip/Direct (ESD) mode decision in this work.

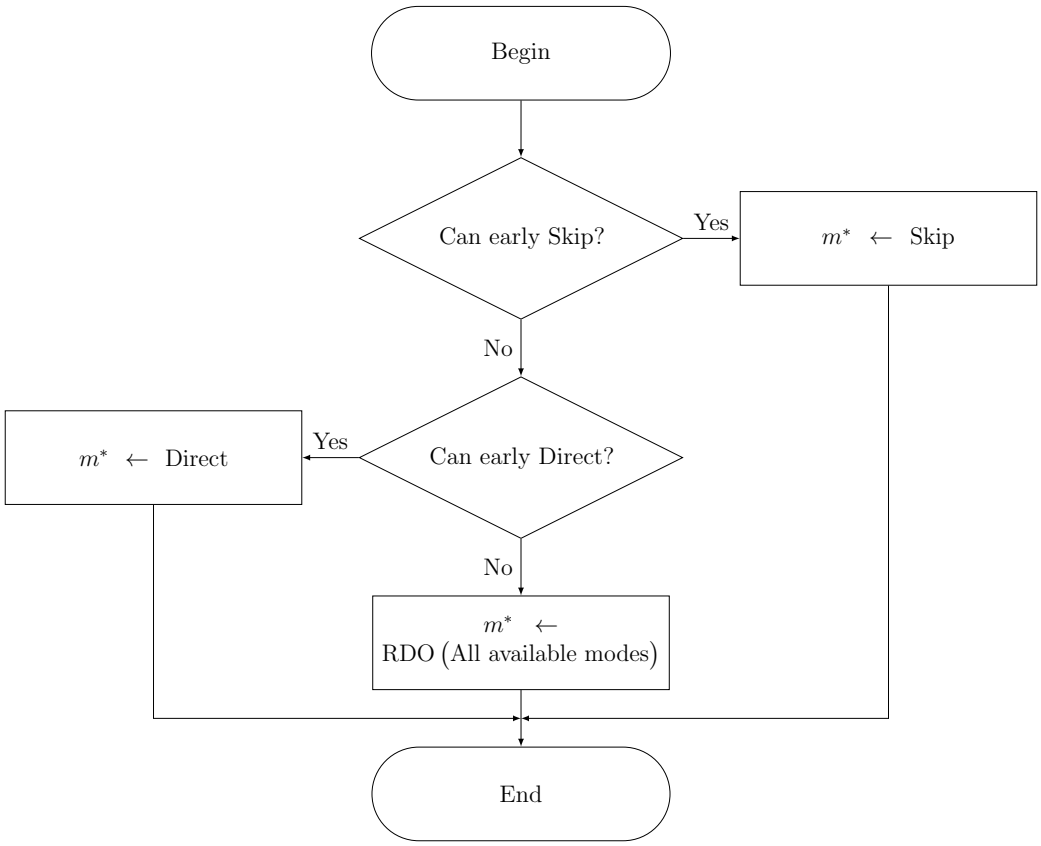


Figure 3.2: Demonstration for the mode decision algorithm with early Skip/Direct technique.

3.2.2 Early Skip/Direct mode decision

In the RDO-based mode decision algorithm (2.2), the Skip and Direct modes are exhaustively evaluated together with other modes. Since for Skip and Direct modes, the PMV is used as MV, if these two modes can be chosen at an early stage, the motion estimation of other Inter-modes could be bypassed to save heavy computation. With our generalized Direct mode, we propose to apply the early skip technique on both Skip and Direct modes, where the Early Skip/Direct (ESD) conditions are checked in the beginning of the mode decision process. If the conditions are satisfied, the Skip/Direct is then selected (and internally noted as the ESD mode) without evaluating other modes, as illustrated in 3.2.

We propose a simple yet effective method to determine the ESD condition, using the thresholding approach. The key idea is to compare the motion-compensated prediction error using PMV (rather than the reconstruction error due to the intensive computation in the transform process) with certain thresholds. The ESD conditions are satisfied if the prediction error is below the thresholds.

To deduce the ESD conditions with non-RDO approach, we investigate how Skip mode is selected by a RDO encoder. Since the PMV is used as MV and the residual is not coded in the Skip mode, for the RDO mode decision algorithm to select the Skip mode, the PMV needs to be accurate, and the prediction error has to be negligibly small. Besides, the 16×16 block partitions used by the Skip mode implies the homogeneous prediction error, otherwise the encoder would favor a smaller partition size.

In our approach, we first derive the PMV from spatial neighbor blocks and temporal collocated blocks, following the algorithm specified in H.264/AVC standard, and then using this PMV to generate the predicted block through motion compensation, with the previous reconstructed frame being the reference. Instead of using the 16×16 block size, the prediction error is computed for each 8×8 sub-block. The error for each sub-block must be smaller than a given Skip threshold, to guarantee the homogeneity of the residual signal, which is also shown to be effective in [32]. Moreover, not only the luma component, but also the chroma components are taken into account, using another threshold. It is necessary to examine the error in chroma component to eliminate false alarms that usually appears when the frame is highly quantized.

For the Direct mode, similar approach can be applied. Since the residual in Direct mode is always coded, it has higher error tolerance than in the Skip mode. Therefore, a relaxed threshold is chosen for the Direct mode. In addition, since the error in chroma components are coded, it is sufficient to check the prediction error only for the luma component. This is because motion estimation is only done based on the luma information (using default JSVM configuration). If the PMV already gives a

relatively small prediction error in the luma component, it is likely that even after motion estimation, a motion vector similar to the PMV would be chosen.

The proposed ESD mode decision algorithm is described as follows. Denote D_l and D_c as the prediction error in luma and chroma components respectively for an 8×8 sub-block. If for all the sub-blocks, D_l is below the thresholds $T_{1,\text{Luma}}$ and D_c is below $T_{1,\text{Chroma}}$, then the Skip mode is selected for the current MB. In the default encoder configuration, only the luma component is considered in the motion search, thus it is necessary to ensure that the error for the chroma component is also small. If the Skip mode criterion is not satisfied, but D_l is still below a more relaxed threshold T_2, then the Direct mode is selected. Otherwise, the RDO based mode decision is performed.

This ESD mode decision is applicable for the ELs as well, where the ESD mode is considered first, and other modes (including the conventional AVC modes and the inter-layer modes in SVC) are checked only if the ESD condition is not satisfied. However, slight modification is required when combining the ESD with proposed constrained mode decision at EL, which will be discussed in Section 3.3.

3.2.3 Early Skip/Direct threshold derivation

The presented ESD mode decision relies on three thresholds so far: $T_{1,\text{Luma}}$, $T_{1,\text{Chroma}}$, and T_2. In this section we demonstrated that only two thresholds are sufficient, and we also present how these thresholds are determined.

In the Skip mode, since the residual is not coded, it can be considered to be quantized to zero. Intuitively, the original residual error should be smaller than the expected quantization error. Therefore to determine the thresholds for the early Skip condition, we investigate the properties of the quantization error e_q.

The quantization error e_q depends mainly on the quantization stepsize q, as well as the prediction error signal distribution. Assuming the video has no scene change, and is coded using fixed QP, then e_q does not have large fluctuation among the entire

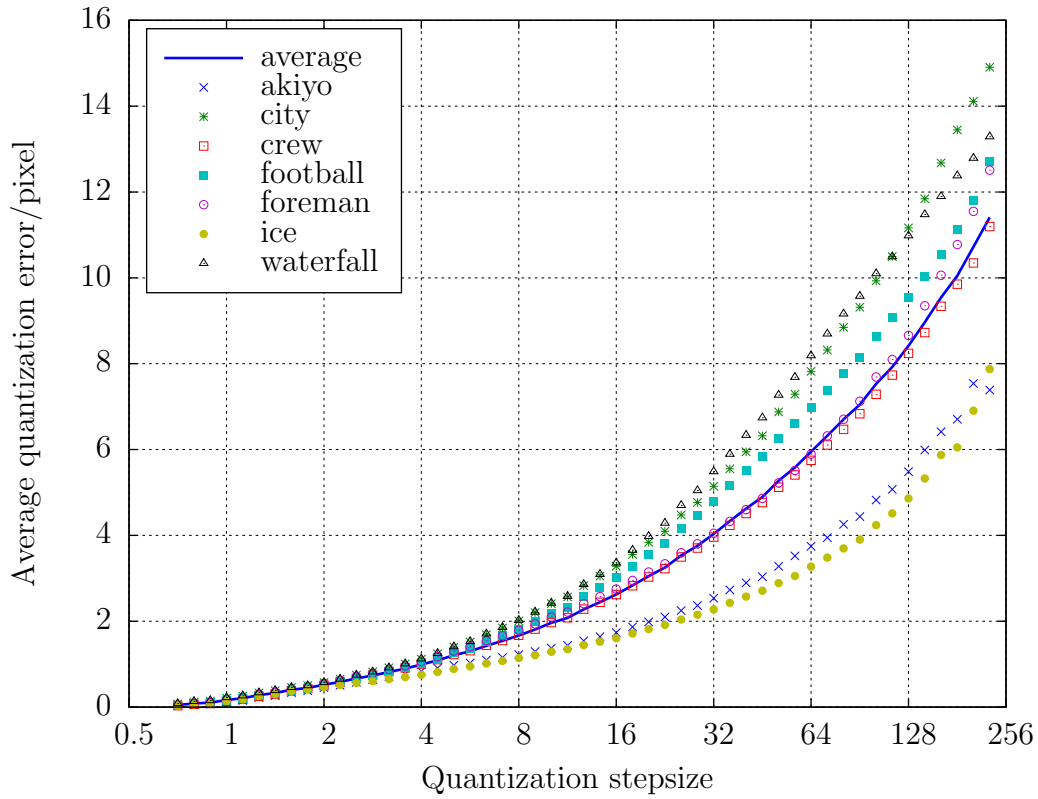


Figure 3.3: Quantization error in luminance v.s. quantization stepsize for seven test sequences. The quantization error is measured using SAD, and averaged over the entire sequence.

sequence. Denote $\bar{e}_q$ as the quantization error averaged for all blocks coded with non-Skip mode over the entire sequence, then in the statistical sense, each MB would yield a quantization error $\bar{e}_q$. Suppose a block has prediction error less than $\bar{e}_q$, then it will probably be quantized to zero, even if we decide to code the residual error. Hence the average quantization error in luma and chroma can be used as the early skip thresholds $T_{1,\text{Luma}}$ and $T_{1,\text{Chroma}}$.

To determine the thresholds, seven CIF test sequences described in Section 2.4.1 are coded by JSVM using single layer encoding configuration. The entire QP range in H.264/AVC (from 1 to 51) is covered in this experiment. The relationship between $\bar{e}_q$ for the luma component and q for these test sequences are presented in Fig. 3.3, where

Table 3.1: Evaluation for error measurement metrics. The number listed are the BD-Rate for the proposed method v.s. original JSVM.

Sequence	SSE(%)	SAD(%)
akiyo	−0.87	−0.92
city	−0.85	−0.89
crew	−0.18	−0.13
football	−0.22	−0.13
foreman	−2.00	−1.81
ice	0.70	0.45
waterfall	−0.88	−0.87

$\bar{e}_q$ is averaged over all non-skipped blocks in the sequence. All data points from different test sequences follow the same trend, despite slight variations among the sequences. Although the individual threshold could be chosen for each sequence, our experiments show that the $\bar{e}_q$ averaged over the seven test sequences serves well as $T_{1,\text{Luma}}$.

To avoid the computational expensive transform, the prediction and quantization error are computed in the pixel domain, as it is equal to that in the transform domain if measured in terms of Sum of Squared Error (SSE), according to Parseval's theorem.

Because SSE requires heavy computation, the Sum of Absolute Difference (SAD) is widely adopted in low-complexity encoders as the error metric. In this work, we have conducted experiments using the JSVM implemented with the Early Skip mode decision using both SSD and SAD in pixel domain as the error measurement metric, and compared it with the default JSVM that uses RDO-based mode decision without Early Skip technique. The seven CIF test sequences are coded with single layer configuration using QP 24, 28, 32, 36, 40, 44, and 48, and the result is listed in Table 3.1, in terms of the BD-Rate. It is noticed that the R-D performance difference between using SAD and SSD is negligible. Moreover, Fig. 3.3 captures the same trend as appeared using SSE (which is not presented in this work), therefore we choose SAD as the error metric, as it requires less computation.

By using the default configurations in JSVM, only the luma component is used in the

motion search, therefore a small prediction error in luminance does not necessarily imply a small error in chroma components. The $\bar{e}_q$ for chroma components are also collected in our experiment, and shown to be approximately equal to half of $T_{1,\text{Luma}}$. However, since human eyes are less sensitive in chrominance than in luminance, we choose a more relaxed threshold, i.e., $T_{1,\text{Luma}}$, for the chroma components as well, and denote it as T_1. Now the ESD mode decision requires only two thresholds: T_1 to determine the Skip mode, and T_2 to determine the Direct mode.

For Direct mode, since both luma and chroma error will be coded, we only check the threshold T_2 for the luminance to ensure the accuracy of PMV. Our experiment show that choosing $T_2 = 1.2T_1$ yields less than 0.5% BD-Rate increment compared with the case where there is no early Direct mode decision, but with noticeable encoder complexity reduction (i.e., more than 5%).

To summarize the ESD mode decision, the flowchart of ESD mode decision is illustrated in Fig. 3.4, where the internal *ESD flag* indicates whether the ESD conditions are satisfied. Note that we need to check whether ESD condition is satisfied in each 8×8 sub-block of the current MB. Only if all sub-blocks satisfy the Skip condition, this MB will be coded using Skip mode. If all sub-blocks satisfy the Direct condition but at least one does not satisfy the Skip condition, this MB will be coded as Direct mode. In case that the ESD condition is not satisfied in one of the sub-block, this block will not be coded as Skip or Direct mode at an early stage (but these two modes may still be selected in RDO mode decision), and it is unnecessary to check the remaining sub-blocks.

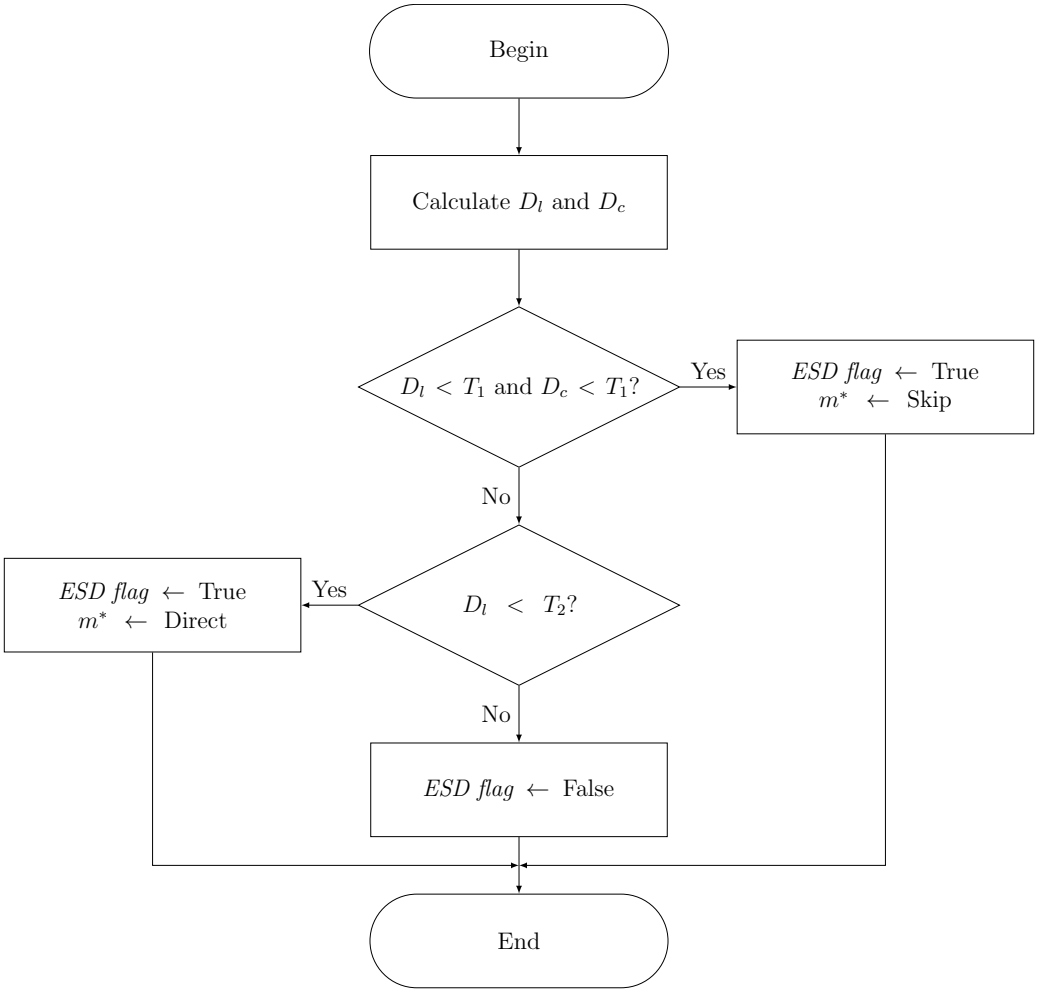


Figure 3.4: Flowchart of ESD mode decision, where D_l and D_c are the prediction error in luma and chroma component, respectively.

3.3 Multilayer Early Skip/Direct mode decision

In SVC, the BL is designed to be compatible with AVC, therefore the proposed ESD mode decision can be applied directly on the BL. Combined with the proposed multilayer mode decision, Fig. 3.5 shows the flowchart of the constrained mode decision at the BL. The ESD condition is checked first, using the reference frame set as discussed in Chapter cha:1p, and the RDO-based mode decision is performed only if the ESD condition is not satisfied.

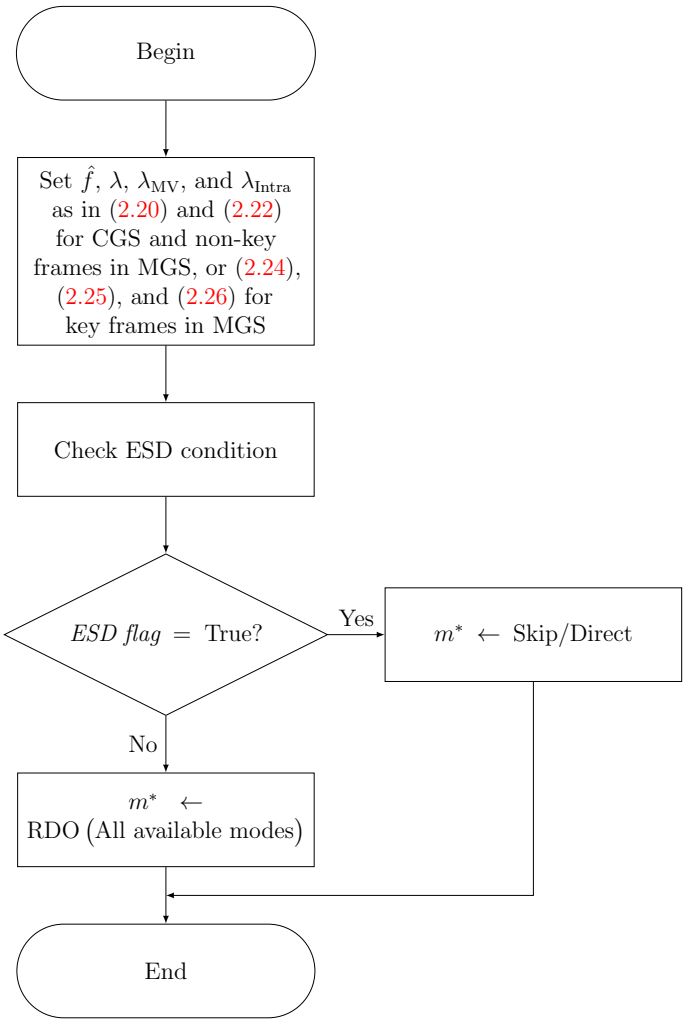


Figure 3.5: Flowchart of constrained mode decision at the BL. The highest layer is used as reference to tune the mode toward higher layers.

When the block at BL is determined as the ESD mode, its MV (reused from PMV) may not reflect the actual motion, because the motion estimation (ME) is bypassed. If such MV is carried to the higher layers, there is no guarantee that this MV remains near-optimal. To resolve this problem, we perform a light-weighted motion search at the EL, with the following constraints:

1. The motion estimation is conducted only if it has not been performed in lower layers, i.e., ESD condition has been satisfied in all lower layers.

2. The block size is always set to 16×16.

The first condition guarantees the current layer to use an accurate MV. Once the accurate MV is obtained, it can be safely forward to higher layers. With this approach, the motion search does not necessarily take place at the BL, but can be at any layer. Note that once the ME is conducted in one of the layers due to the ESD condition is not satisfied, the higher layers do not need ME any more. Hence the motion search is performed at most once for the same block among all layers.

If the lower layers satisfy the ESD condition, it implies that the texture of this block is easy to predict, yielding relatively homogeneous prediction error. Thus the 16×16 block partition size is sufficient in the motion estimation. In such case, the best mode at i-th layer for the CGS structure and MGS non-key frames is determined using

$$\begin{aligned} m_i^* &= \arg\min_{m_i} J\left(m_i; f, \hat{f}_L, \lambda\left(\mathrm{QP}_i\right)\right) \\ &= \arg\min_{m_i} \left(D\left(f, \hat{f}_L\left(m_i\right)\right) + \lambda\left(\mathrm{QP}_i\right) R\left(m_i, f - \hat{f}_L\left(m_i\right)\right)\right), \end{aligned} \tag{3.1}$$

and the best MV is selected using

$$\begin{aligned} v_i^* &= \arg\min_{v_i} J_{\mathrm{Inter}}\left(v_i; f, \hat{f}_L, \lambda_{\mathrm{MV}}\left(\mathrm{QP}_L\right)\right) \\ &= \arg\min_{v_i} \left(D_{\mathrm{MV}}\left(f, \hat{f}_L\left(v_i\right)\right) + \lambda_{\mathrm{MV}}\left(\mathrm{QP}_L\right) R_{\mathrm{MV}}\left(v_i\right)\right). \end{aligned} \tag{3.2}$$

For MGS key frames, the best mode and MV are determined by

$$\begin{aligned} m_i^* &= \arg\min_{m_i} J\left(m_i; f, \hat{f}_0, \lambda\left(\mathrm{QP}_i\right)\right) \\ &= \arg\min_{m_i} \left(D\left(f, \hat{f}_0\left(m_i\right)\right) + \lambda\left(\mathrm{QP}_i\right) R\left(m_i, f - \hat{f}_0\left(m_i\right)\right)\right), \end{aligned} \tag{3.3}$$

$$\begin{aligned} v_i^* &= \arg\min_{v_i} J_{\mathrm{Inter}}\left(v_i; f, \hat{f}_0, \lambda_{\mathrm{MV}}\left(\mathrm{QP}_L\right)\right) \\ &= \left(D_{\mathrm{MV}}\left(f, \hat{f}_0\left(v_i\right)\right) + \lambda_{\mathrm{MV}}\left(\mathrm{QP}_i\right) R_{\mathrm{MV}}\left(v_i\right)\right). \end{aligned} \tag{3.4}$$

Note that the ESD mode does not affect the Intra-prediction mode decision. The flowchart of constrained mode decision combined with ESD at the EL is shown in Fig. 3.6.

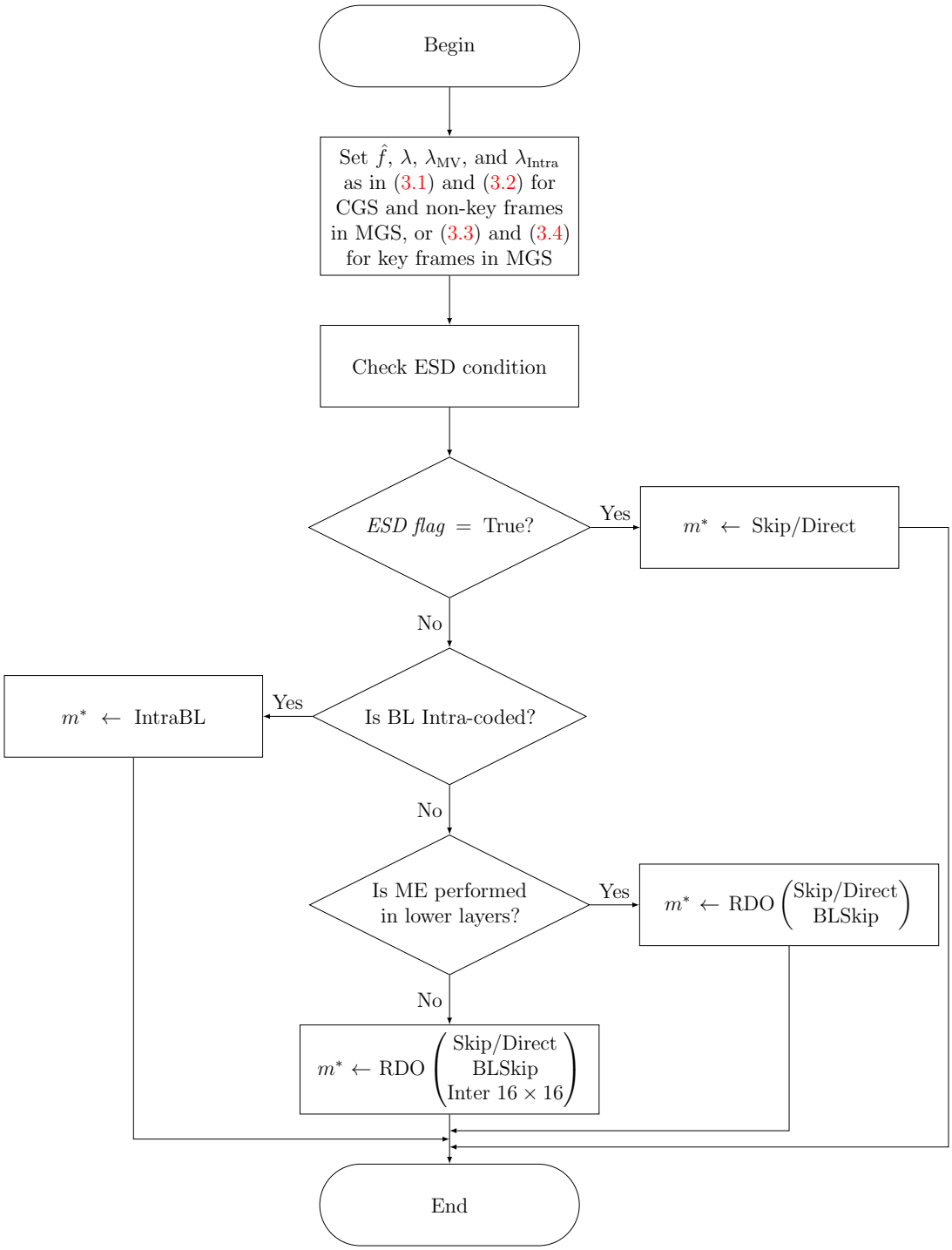


Figure 3.6: Flowchart of constrained mode decision at the EL.

If the ESD condition is not satisfied, the inferred mode inherited from BL is taken into consideration, depending on whether the BL is Inter- or Intra-coded. In the case that the BL is Inter-coded and ME has not been performed in all lower layers due to ESD, we perform ME, but only at 16×16 basis.

3.4 Performance evaluation and discussions

3.4.1 Evaluation under the CGS coding structure

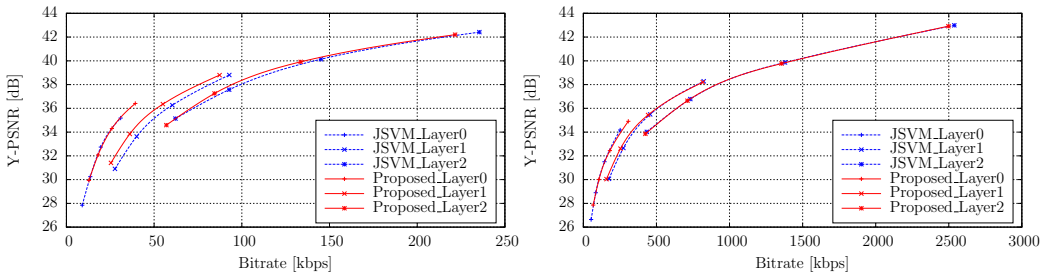
The same coding configurations described in 2.4.1 is also applied on the evaluation for multilayer ESD mode decision.

Fig. 3.7 visualizes the coding efficiency evaluation for the CIF test sequences using CGS coning structure. It is observed that our proposed method has very close or higher coding efficiency than the original JSVM.

The encoder complexity measure in terms of the total encoding time is shown in Fig. 3.8. All the test sequences show above 50% time saving for encoding all three layers. Especially for the sequence akiyo, which has a stationary background and consequently extremely small prediction error, over 70% reduction in the total encoding time is reached.

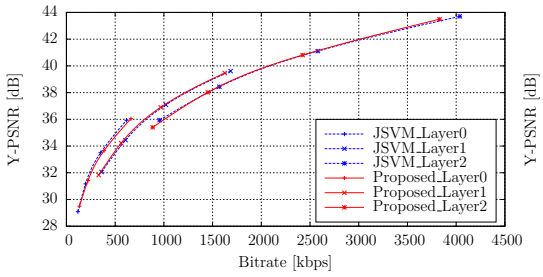
Fig. 3.9 plots the encoding time consumed in mode decision at each layer. While the ELs has rather low complexity, the BL also enjoys different levels of complexity reduction, as expected.

The complete simulation results for the CIF test sequences are listed in Table 3.2, and the results without the ESD mode is also listed for convenient comparison. By introducing the ESD mode, the coding efficiency drops slightly, however the time reduction for encoding all layers boosts to 57.7% on average, with an average mode decision time reduction of 33.0% at the BL. The BLs are also beneficial from the ESD

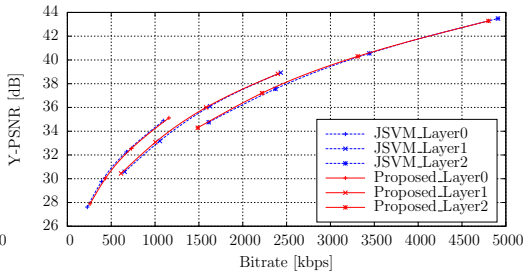


(a) akiyo

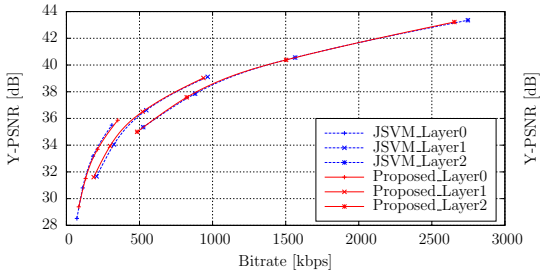
(b) city



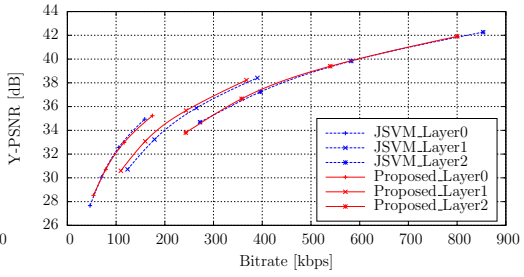
(c) crew



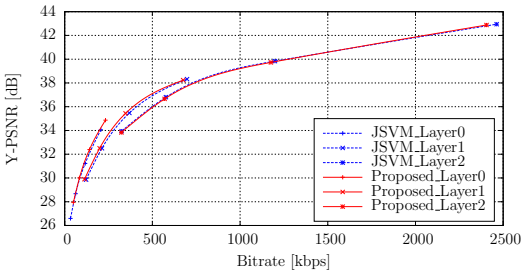
(d) football



(e) foreman

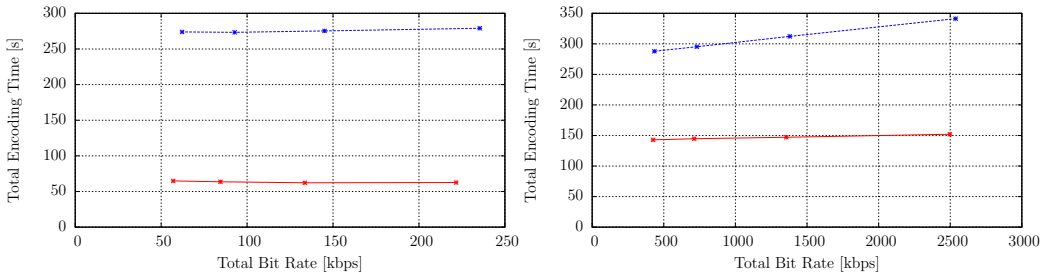


(f) ice



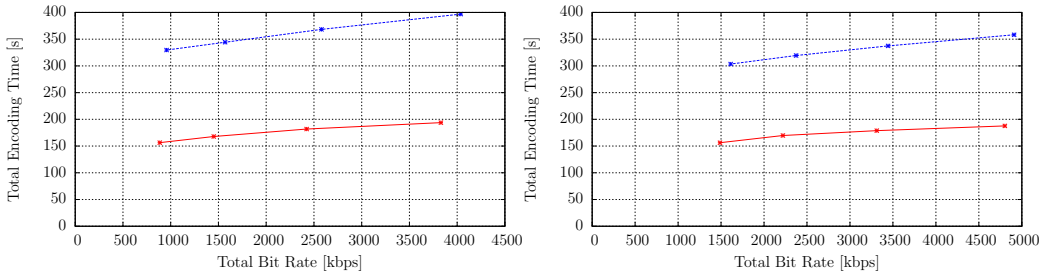
(g) waterfall

Figure 3.7: Performance comparison for coding efficiency of proposed algorithm v.s. default JSVM of CIF test sequences using CGS coding structure with ESD mode decision enabled.



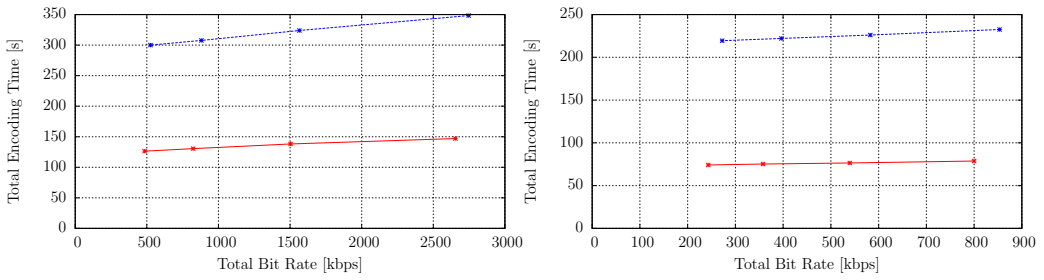
(a) akiyo

(b) city



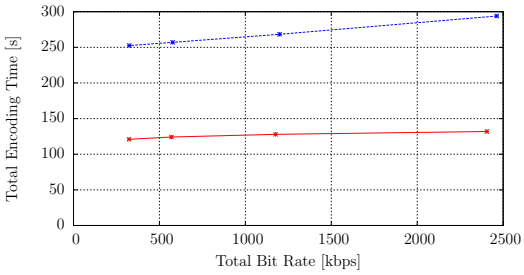
(c) crew

(d) football



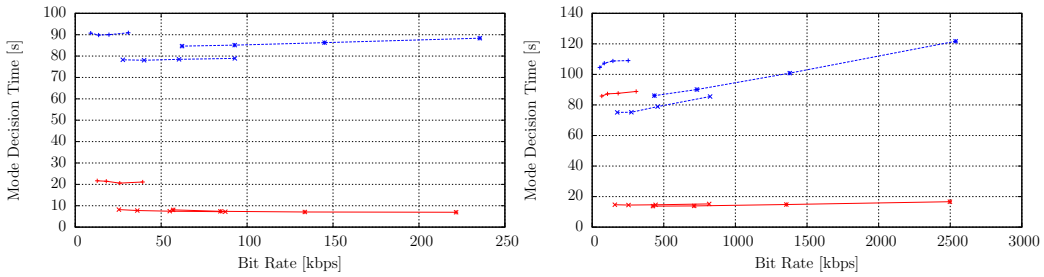
(e) foreman

(f) ice



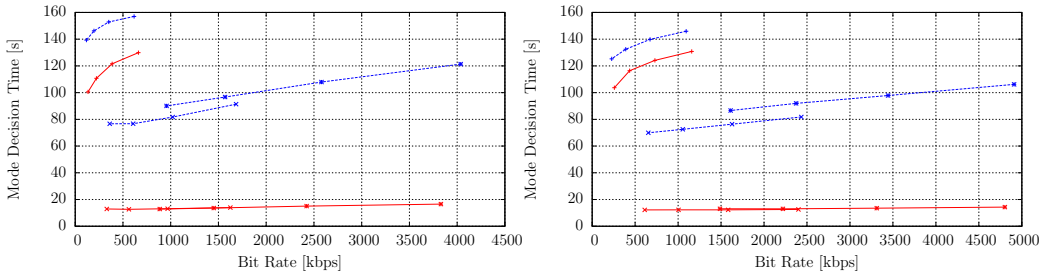
(g) waterfall

Figure 3.8: Performance comparison for total encoding time of proposed algorithm v.s. default JSVM of CIF test sequences using CGS coding structure with ESD mode decision enabled.



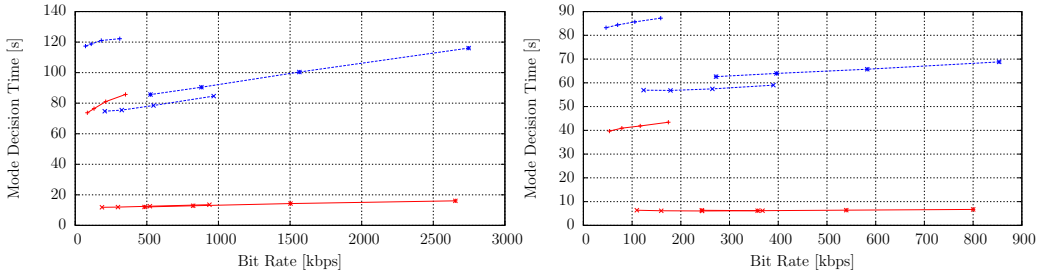
(a) akiyo

(b) city



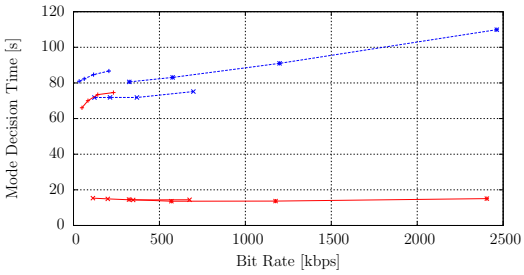
(c) crew

(d) football



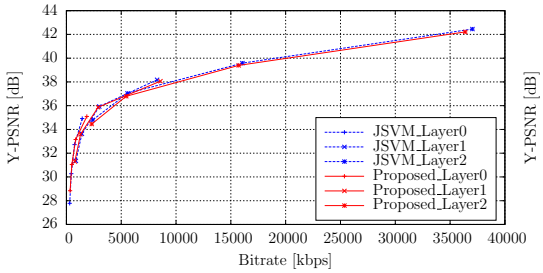
(e) foreman

(f) ice

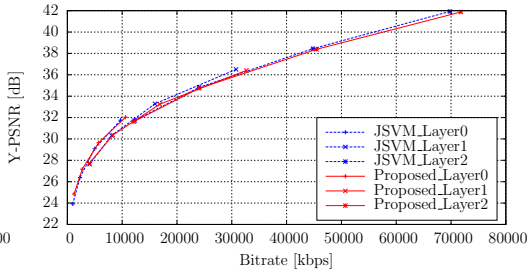


(g) waterfall

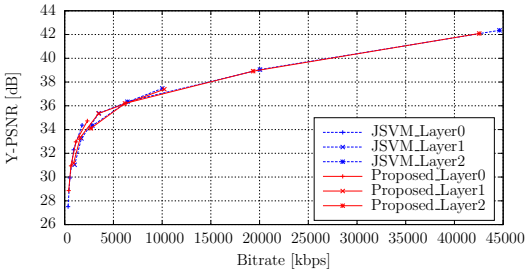
Figure 3.9: Performance comparison for mode decision (including motion estimation) time of proposed algorithm v.s. default JSVM of CIF test sequences using CGS coding structure with ESD mode decision enabled.



(a) mobcal



(b) parkrun



(c) shields

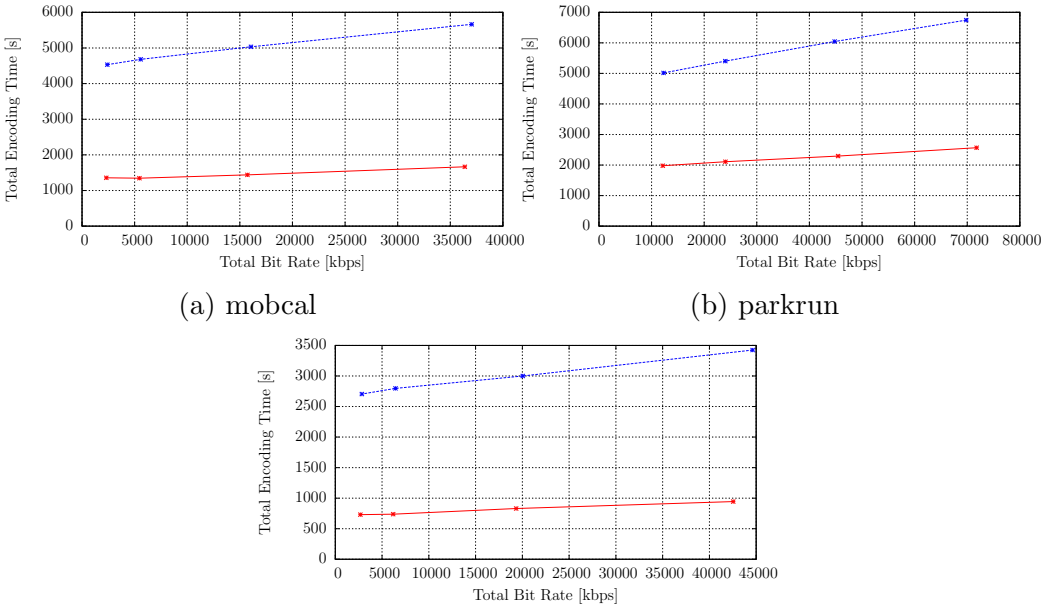
Figure 3.10: Performance comparison for coding efficiency of proposed algorithm v.s. default JSVM of 720p test sequences using CGS coding structure with ESD mode decision enabled.

mode decision, with average over 80% time reduction in the mode decision time.

The results for the 720p test sequences are plotted in Fig. 3.10, 3.11, and 3.12, and detailed in Table 3.3. Following the same trend as in the CIF sequences, the 720p sequences, has slight coding efficiency degradation with ESD mode enabled, but with average of 68.4% reduction in total encoding time, and over 50% time saving in the mode decision at the BL.

Table 3.2: Performance Evaluation of Proposed Algorithm for CIF using CGS with ESD

Sequence	Layer	without ESD			with ESD		
		BD-Rate	ΔT	ΔT_m	BD-Rate	ΔT	ΔT_m
akiyo	0	1.2%	44.8%	1.4%	0.6%	77.0%	76.5%
	1	−11.6%		77.5%	−11.0%		90.2%
	2	−2.4%		79.6%	−3.4%		91.4%
city	0	1.0%	45.5%	0.3%	1.4%	52.4%	18.7%
	1	−4.4%		79.9%	−4.2%		81.2%
	2	0.1%		84.6%	0.2%		85.1%
crew	0	4.1%	42.2%	2.4%	5.6%	51.4%	22.5%
	1	−2.9%		81.5%	−2.2%		83.8%
	2	−0.7%		85.1%	−0.2%		86.0%
football	0	2.7%	40.8%	−2.6%	3.6%	47.5%	12.7%
	1	−2.5%		82.4%	−2.1%		83.5%
	2	−1.3%		85.6%	−1.2%		85.9%
foreman	0	1.5%	44.3%	2.3%	2.9%	57.6%	33.9%
	1	−5.1%		80.1%	−4.2%		84.1%
	2	−1.0%		84.3%	−0.8%		85.9%
ice	0	0.0%	42.1%	1.9%	1.7%	66.2%	51.3%
	1	−8.6%		78.7%	−6.6%		89.2%
	2	−2.9%		81.2%	−1.1%		90.2%
waterfall	0	−5.0%	47.8%	−0.5%	−4.6%	52.8%	15.1%
	1	−5.7%		79.2%	−5.4%		79.7%
	2	0.7%		84.0%	1.0%		85.2%
Average	0	0.8%	43.9%	0.7%	1.6%	57.7%	33.0%
	1	−5.8%		79.9%	−5.1%		84.5%
	2	−1.1%		83.5%	−0.8%		87.0%



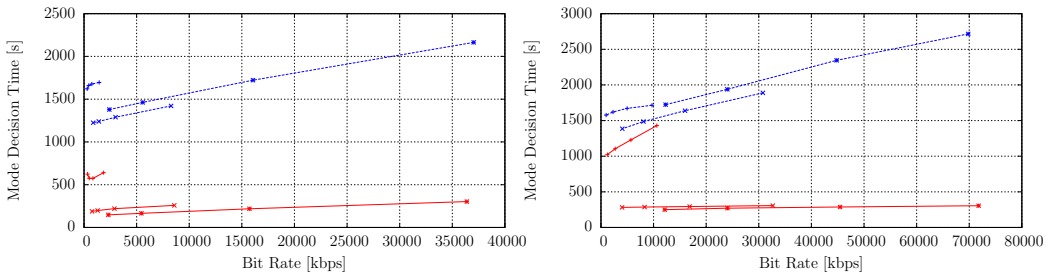
(a) mobcal (b) parkrun

(c) shields

Figure 3.11: Performance comparison for total encoding time of proposed algorithm v.s. default JSVM of 720p test sequences using CGS coding structure with ESD mode decision enabled.

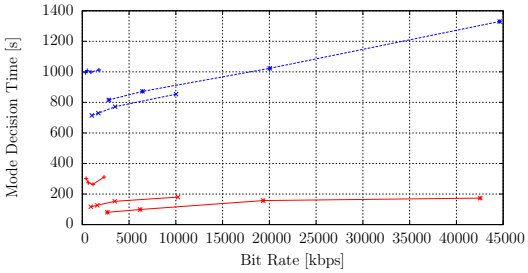
Table 3.3: Performance Evaluation of Proposed Algorithm for 720p using CGS with ESD

Sequence	Layer	without ESD			with ESD		
		BD-Rate	ΔT	ΔT_m	BD-Rate	ΔT	ΔT_m
mobcal	0	−9.0%	46.8%	0.3%	0.0%	70.8%	63.7%
	1	−7.9%		78.6%	−3.6%		83.4%
	2	2.2%		83.4%	6.2%		87.8%
parkrun	0	−5.1%	53.1%	−0.3%	−4.5%	61.4%	27.5%
	1	3.9%		81.4%	3.9%		81.6%
	2	2.2%		86.6%	2.2%		87.0%
shields	0	−2.1%	46.8%	0.3%	−0.2%	72.8%	71.3%
	1	−3.9%		78.9%	−4.5%		81.4%
	2	1.4%		84.0%	1.4%		87.6%
Average	0	−5.4%	48.9%	0.1%	−1.6%	68.4%	54.2%
	1	−2.6%		79.6%	−1.4%		82.1%
	2	1.9%		84.7%	3.3%		87.5%



(a) mobcal

(b) parkrun



(c) shields

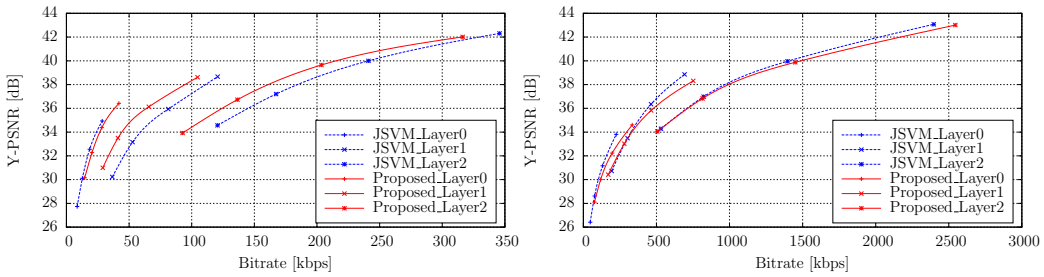
Figure 3.12: Performance comparison for mode decision (including motion estimation) time of proposed algorithm v.s. default JSVM of 720p test sequences using CGS coding structure with ESD mode decision enabled.

3.4.2 Evaluation under the MGS coding structure

The coding efficiency, the total encoding time, and the time consumed by mode decision, for the CIF test sequences using the MGS coding structure are demonstrated in Fig. 3.13, 3.14, and 3.15, respectively.

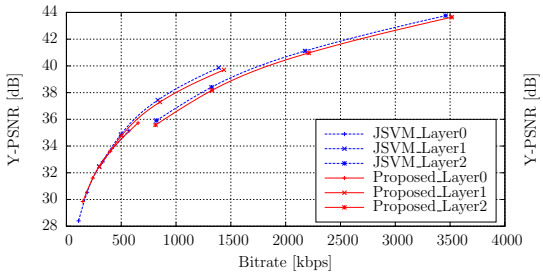
As expected, compared with the non-ESD case, the coding efficiency drops slightly with the ESD mode enabled, however average 57.1% complexity saving for mode decision at the BL is achieved, resulting the 66.5% saving in total encoding time on average for the CIF test sequences, as detailed in Table 3.4.

The performance evaluation for the 720p test sequences are visualized in Fig. 3.16, 3.17, and 3.18, respectively. The detailed results are shown in Table 3.5. Similar to the CIF sequences, slight coding efficiency loss is observed, but with total encoding time saving reaching 74.5% on average. The time saving for the mode decision is quite

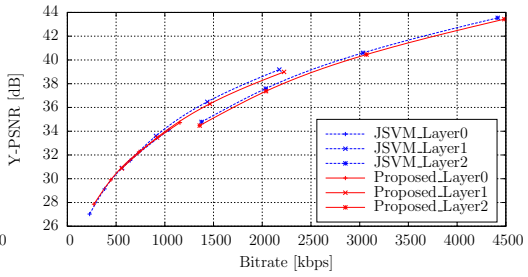


(a) akiyo

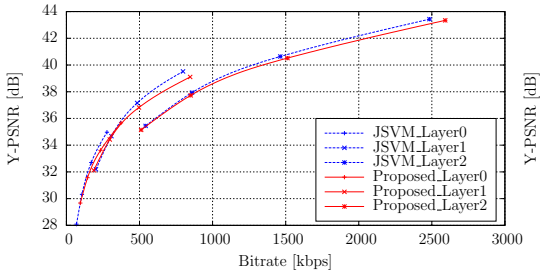
(b) city



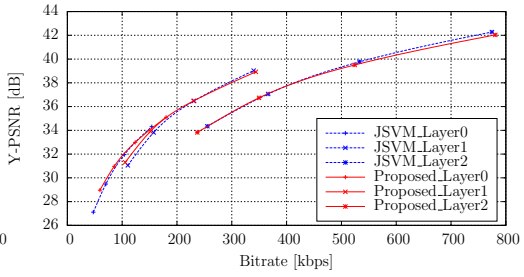
(c) crew



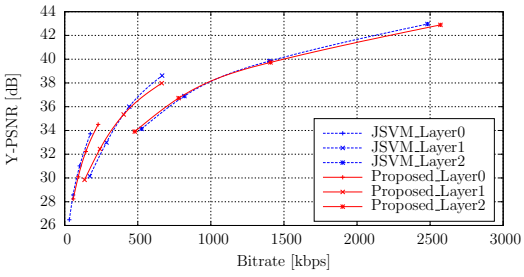
(d) football



(e) foreman

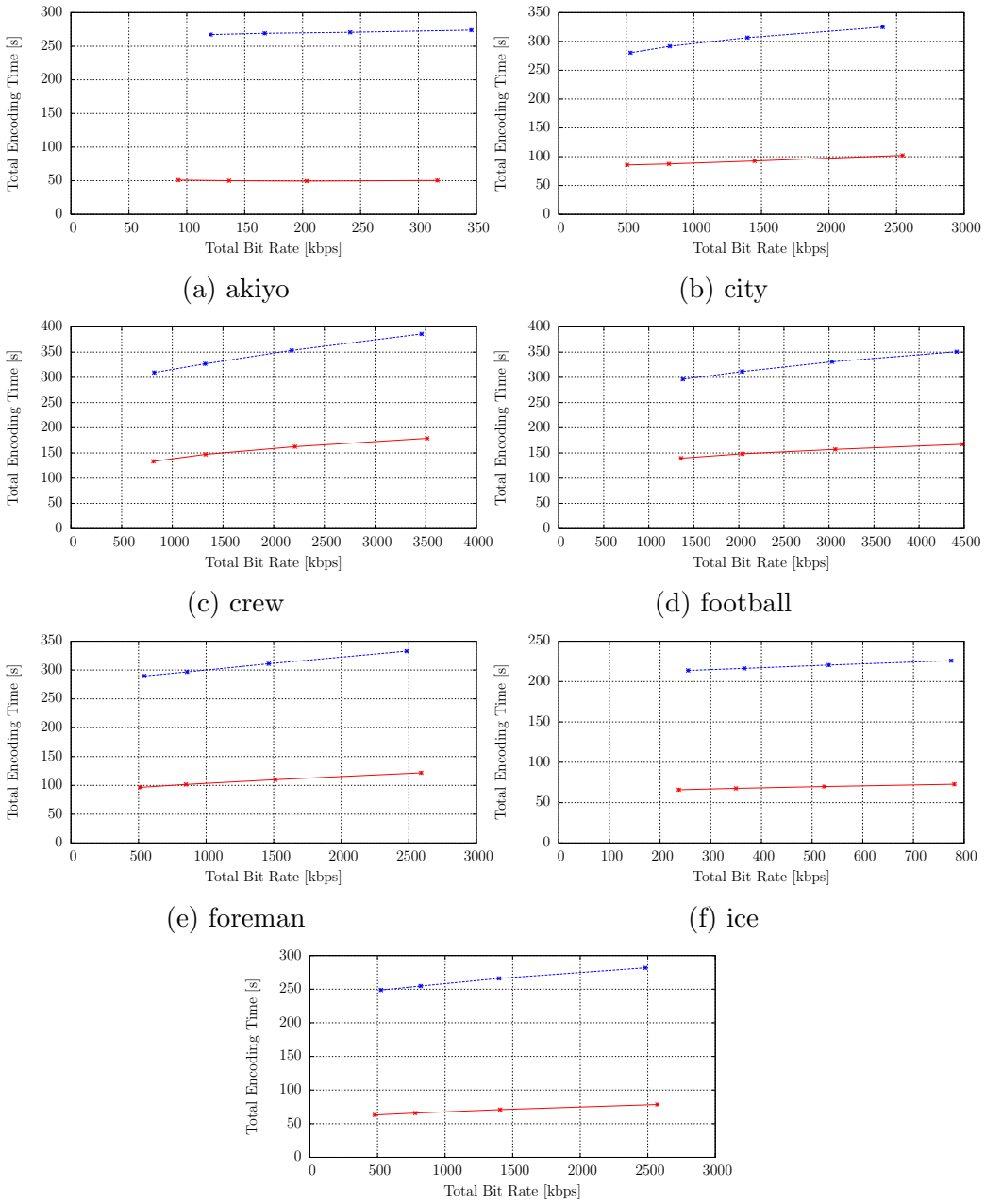


(f) ice



(g) waterfall

Figure 3.13: Performance comparison for coding efficiency of proposed algorithm v.s. default JSVM of CIF test sequences using MGS coding structure with ESD mode decision enabled.



(a) akiyo

(b) city

(c) crew

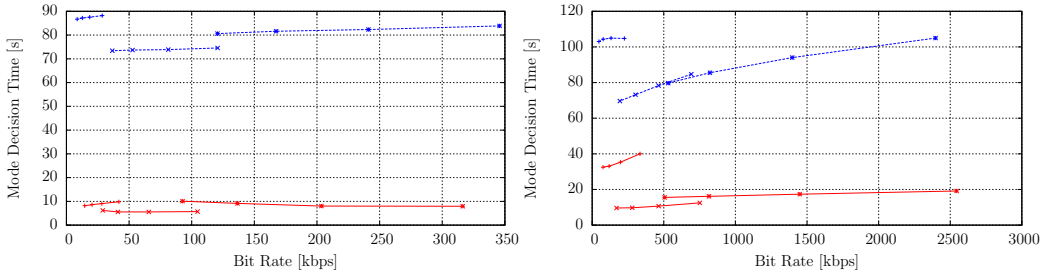
(d) football

(e) foreman

(f) ice

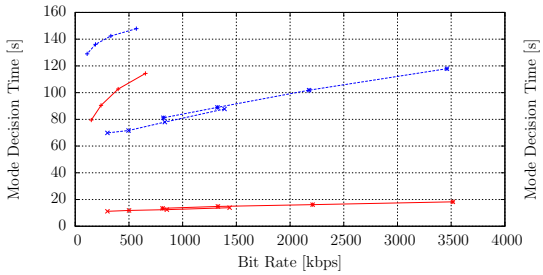
(g) waterfall

Figure 3.14: Performance comparison for total encoding time of proposed algorithm v.s. default JSVM of CIF test sequences using MGS coding structure with ESD mode decision enabled.

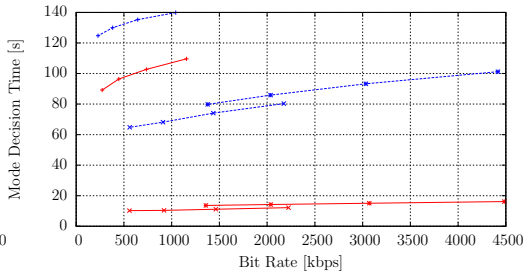


(a) akiyo

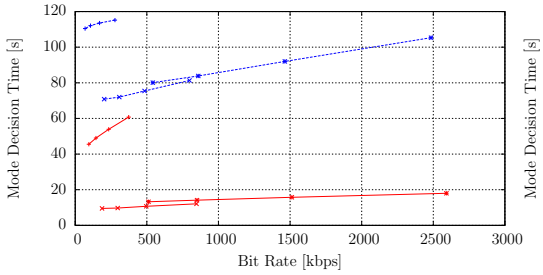
(b) city



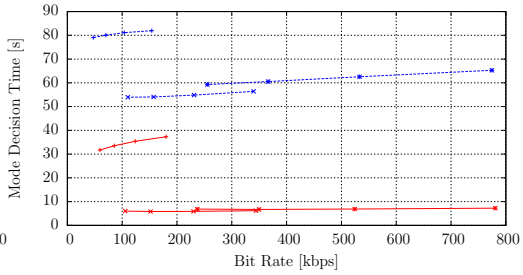
(c) crew



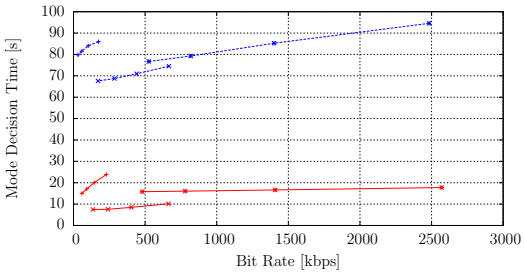
(d) football



(e) foreman

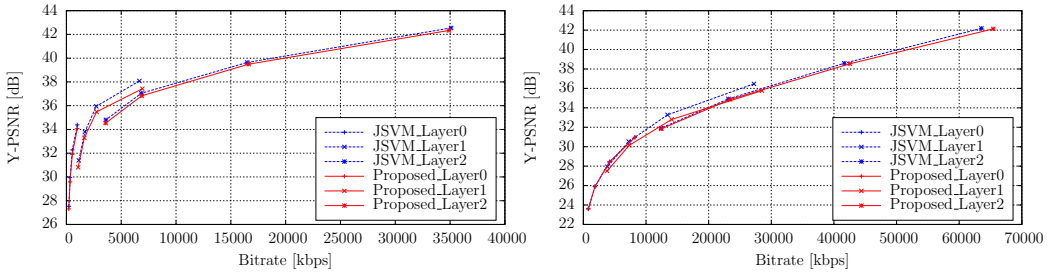


(f) ice

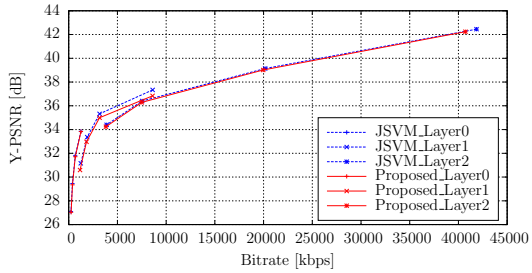


(g) waterfall

Figure 3.15: Performance comparison for mode decision (including motion estimation) time of proposed algorithm v.s. default JSVM of CIF test sequences using MGS coding structure with ESD mode decision enabled.



(a) mobcal (b) parkrun



(c) shields

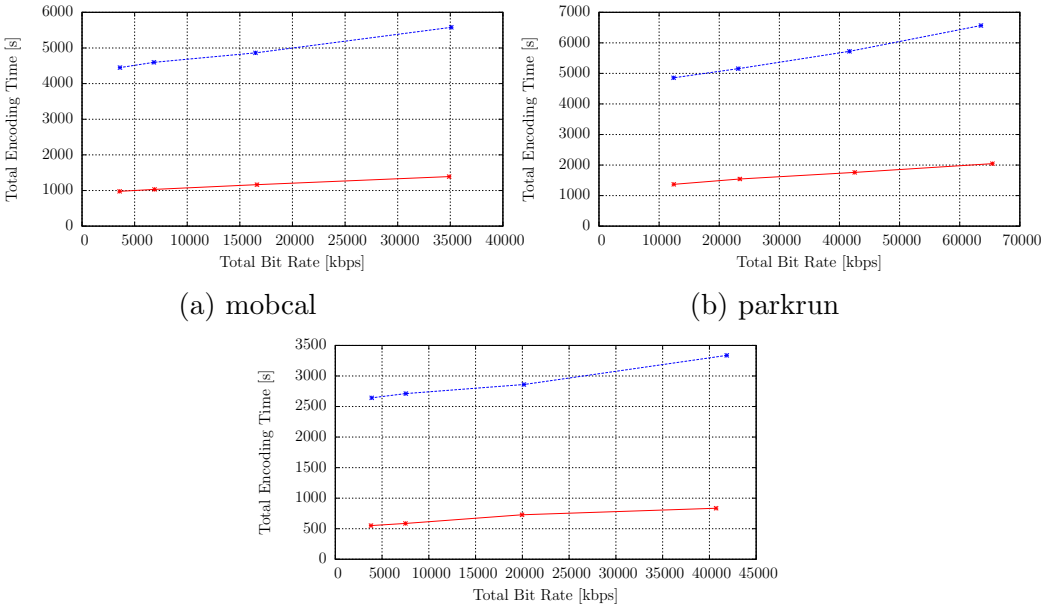
Figure 3.16: Performance comparison for coding efficiency of proposed algorithm v.s. default JSVM of 720p test sequences using MGS coding structure with ESD mode decision enabled.

significant with average of 75.6%.

Compared with existing low-complexity mode decision algorithms for SVC [37, 17, 18, 19, 20], our simulation results show coding efficiency gain for a number of test sequences, which is due to the multilayer mode decision, whereas in other approaches, the mode decision is tuned toward current layer. With the proposed scheme, in the worst case, at the EL, the encoder either directly chooses the IntraBL mode, or from the four Inter-modes (i.e., Skip, Direct, BLSkip, and Inter 16×16) using RDO based approach. Note that the computational intensive ME is only required by the Inter 16×16 mode, which is enabled when the ME has not been conducted at lower layers, whereas in the existing published algorithms, the encoder still needs to try multiple motion searches. For example, in [17, 19], the encoder always needs to conduct ME for each MB at the EL. Moreover, [18] adopts a threshold based approach, similar to the proposed ESD mode, but in the worst case, when the early stop condition is not satisfied, all the Inter-

Table 3.4: Performance Evaluation of Proposed Algorithm for CIF using MGS with ESD

Sequence	Layer	without ESD			with ESD		
		BD-Rate	ΔT	ΔT_m	BD-Rate	ΔT	ΔT_m
akiyo	0	12.9%		−1.0%	13.3%		89.8%
	1	−26.7%	44.1%	75.7%	−22.9%	81.5%	92.2%
	2	−16.9%		78.0%	−11.8%		89.3%
city	0	14.8%		−0.9%	16.9%		66.2%
	1	−1.6%	43.8%	77.5%	4.5%	69.5%	86.1%
	2	2.1%		81.2%	3.7%		81.2%
crew	0	−4.0%		−1.8%	−1.7%		30.6%
	1	0.5%	40.5%	79.9%	4.2%	54.9%	83.9%
	2	3.4%		83.1%	4.5%		83.9%
football	0	−3.3%		−1.6%	−1.7%		25.0%
	1	1.1%	40.6%	80.7%	3.8%	52.5%	84.7%
	2	2.6%		83.5%	3.1%		83.6%
foreman	0	6.3%		−1.3%	8.9%		53.7%
	1	−1.0%	42.5%	78.0%	4.9%	65.1%	86.0%
	2	2.3%		81.7%	4.2%		83.1%
ice	0	−4.8%		−1.4%	−2.7%		57.2%
	1	−6.9%	41.2%	77.1%	−2.7%	68.5%	89.1%
	2	−2.4%		79.5%	1.2%		88.8%
waterfall	0	9.5%		−0.7%	10.4%		77.2%
	1	−7.4%	46.2%	77.2%	−3.1%	73.6%	88.1%
	2	−1.0%		81.0%	0.4%		80.2%
Average	0	4.5%		−1.3%	6.2%		57.1%
	1	−6.0%	42.7%	78.0%	−1.6%	66.5%	87.2%
	2	−1.4%		81.2%	0.8%		84.3%



(a) mobcal

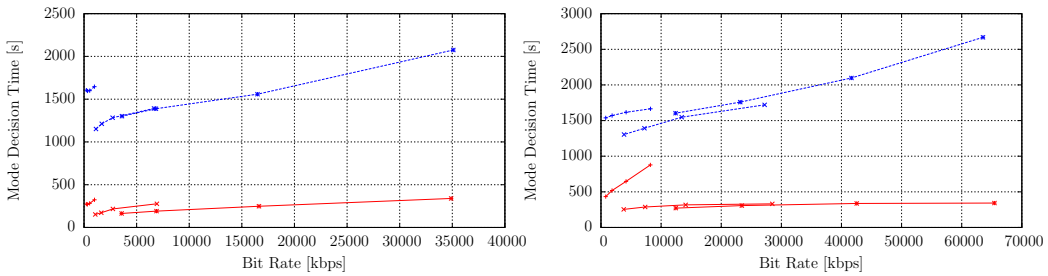
(b) parkrun

(c) shields

Figure 3.17: Performance comparison for total encoding time of proposed algorithm v.s. default JSVM of 720p test sequences using MGS coding structure with ESD mode decision enabled.

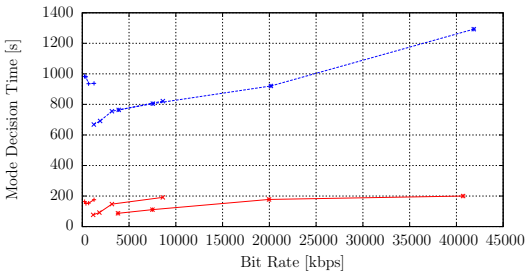
Table 3.5: Performance Evaluation of Proposed Algorithm for 720p using MGS with ESD

Sequence	Layer	without ESD			with ESD		
		BD-Rate	ΔT	ΔT_m	BD-Rate	ΔT	ΔT_m
mobcal	0	0.0%	45.7%	−0.1%	5.9%	76.7%	82.2%
	1	4.7%		76.2%	13.4%		83.9%
	2	2.6%		80.4%	6.2%		85.3%
parkrun	0	0.0%	51.3%	−0.4%	0.5%	70.0%	61.6%
	1	12.5%		78.9%	14.9%		80.1%
	2	2.3%		83.5%	2.7%		84.2%
shields	0	0.1%	45.6%	−0.5%	3.7%	76.8%	83.2%
	1	5.0%		76.2%	10.2%		83.1%
	2	1.0%		80.5%	2.7%		85.0%
Average	0	0.0%	47.5%	−0.3%	3.4%	74.5%	75.6%
	1	7.4%		77.1%	12.8%		82.4%
	2	2.0%		81.5%	3.9%		84.8%



(a) mobcal

(b) parkrun



(c) shields

Figure 3.18: Performance comparison for mode decision (including motion estimation) time of proposed algorithm v.s. default JSVM of 720p test sequences using MGS coding structure with ESD mode decision enabled.

and Intra-modes are evaluated.

For the Intra-mode decision at the EL, the proposed scheme only performs IntraBL mode, whereas [18] needs to performed nine Intra-predictions under 4×4 block size, plus the IntraBL mode. Other methods [37, 19, 20] require even more computational resource to further check 16×16 block based predictions in addition to 4×4 block based prediction and IntraBL.

Note that the proposed ESD mode decision is hardware-friendly, as the thresholds for different QPs can be stored in a look-up table, and no additional memory is required to store the spatial or temporal historical data. In addition, since it is designed for H.264/AVC, it can be directly applied on AVC and the BL in SVC without any modification.

3.5 Energy consumption savings with proposed algorithm

The reduction in encoding complexity (i.e., computation cycles) can lead to energy consumption savings. With the proposed mode decision algorithm, roughly 50% of encoding time can be achieved for each frame. [38] has demonstrated power consumption reduction by exploiting modern devices (CPUs or other processing chips) that employ dynamic voltage and frequency scaling (DVFS), under the scenario that each frame is restricted by a fixed time budget to finish coding.

Power consumption models for different DVFS enabled CPUs have been studied in [39]. The dynamic power P_{dyn} is

$$P_{\text{dyn}} = K_{\text{eff}} V_{\text{dd}}^2 f, \tag{3.5}$$

with K_{eff} as the effective circuit capacitance, V_{dd} is the supportable voltage, and f is the clock frequency. V_{dd} can be modeled by

$$V_{\text{dd}} = \omega f^{\phi} + \theta, \tag{3.6}$$

where ω, ϕ, and θ are CPU-dependent constants. For the Intel Pentium M 1.6 GHz processor, $\omega = 5.5 \times 10^{-10}$, $\phi = 1$, and $\theta = 0.61$. For conventional encoder, assuming it requires the CPU running at peak frequency 1.6 GHz, which leads to the dynamic power consumption of $3.55 K_{\text{eff}} \times 10^9$, whereas with the proposed algorithm requires only half of the frequency, leading to consumption of $0.88 K_{\text{eff}} \times 10^9$, i.e., approximately 75% energy savings.

For the ARM Cortex A8 600 MHz processor, in addition to the dynamic power, the non-negligible leakage power need to be considered. The total power consumption P is modeled in [39]

$$P\left(V_{\text{dd}}\right) = 0.145 V_{\text{dd}}^{1.44} + 1.12 V_{\text{dd}} \exp\left(7.05 V_{\text{dd}}\right) + 0.12, \tag{3.7}$$

where V_{dd} has the same functional form as shown in (3.6), with $\omega = 6 \times 10^{-16}$, $\phi = 1.69$, and $\theta = 0.91$. With conventional mode decision algorithm, assuming the CPU runs at maximum frequency of 600 MHz, leading to 0.5 Watt power consumption. With 50% complexity savings in the proposed algorithm, the CPU only needs 300 MHzfor encoding in the same scenario, leading to 0.29 Watt power consumption, i.e., with 42% energy reduction.

3.6 Summary and discussions

In this chapter, we present an early Skip/Direct (ESD) mode decision algorithm, by utilizing the unified Skip and Direct modes. When the ESD condition is satisfied, the Skip or Direct mode is selected without going through the RDO-based mode decision. When combining the ESD mode decision method with the multilayer mode decision scheme presented in Chapter 2, slight modifications are made for the EL. In the case that the motion estimation (ME) has not been conducted at lower layers, it will be performed at the current layer using block size of 16×16 only. With this approach, the ME is conducted at most once among all layers. The simulation results for the CGS and MGS structure show slightly worse coding efficiency compared to the case where ESD is disabled, but with ESD mode, the complexity reduction at the BL is quite noticeable.

For CIF resolution, ESD boosts the average mode decision complexity savings at the BL from 0.7% to 33.0% for CGS, and -1.3% to 57.1% for MGS, with overall complexity savings from 43.9% to 57.7% for CGS, and from 42.7% to 66.5% for MGS. 720p test sequences achieve even greater complexity reduction, with mode decision time reduced by 54.2% and 75.6% for CGS and MGS respectively, the average overall encoding time saving boosts from 48.9% to 68.4% for CGS, and from 47.5% to 74.5% for MGS. Such complexity reduction is achieved with only slight coding efficiency loss.

When implemented on the devices supporting dynamic voltage and frequency scaling,

with roughly 50% saving in the CPU cycles, the proposed algorithm could lead to 75% and 42% savings in energy consumption for Intel Pentium M 1.6 GHz processor and ARM Cortex A8 600 MHz processor, respectively.

Chapter 4

Rate and distortion modeling

In this chapter we investigate the relationship between the video bitrate and prediction error (for the best mode selected by RDO encoder) together with the quantization error. We also investigate the the relationship between the quantization error and the quantization stepsize. We have found analytical forms that fit the measured data well. With that, the video bitrate (for single layer video) can be predicted from the prediction error and the quantization stepsize. A low-complexity encoding scheme is also presented with utilizing the proposed rate and quantization error model. Moreover, the rate model is applied on the temporal scalability, and the result demonstrates high Pearson correlation between measured rates and the ones predicted by the proposed model.

4.1 Motivation and related works

Several prior research works have investigated the rate modeling in non-scalable video, and proposed the rate modes that is related with the quantization stepsize q.

Ding and Liu reported the following model [40]

$$R = \frac{\theta}{q^{\gamma}}, \tag{4.1}$$

where θ and γ are model parameters, with $0 \leq \gamma \leq 2$. Chiang and Zhang suggested the following model [41]

$$R = \frac{A_1}{q} + \frac{A_2}{q^2}. \tag{4.2}$$

This so-called quadratic rate model has been used for rate-control in MPEG-4 reference encoder [42]. Only the quadratic term was included in the model by Ribas-Cobera and Lei [43], i.e.,

$$R = \frac{A}{q^2}. \tag{4.3}$$

More recently, He *et. al* [44] proposed the ρ-model,

$$R\left(\mathrm{QP}\right) = \theta\left(1 - \rho\left(\mathrm{QP}\right)\right), \tag{4.4}$$

with ρ denoting the percentage of zero quantized transform coefficients with a given quantization parameter. This model has been shown to have high accuracy for rate prediction. A problem with the ρ-model is that it does not provide explicit relation between QP and ρ. Therefore, it does not lend itself to theoretical understanding of the impact of QP on the rate.

Although these rate models may successfully predict the overall bitrate for the entire coded video sequences, for one particular frame, the rate may deviate greatly from the average rate. In the next section, we relate the video rate with the prediction error, with considering of the temporal scalability structure.

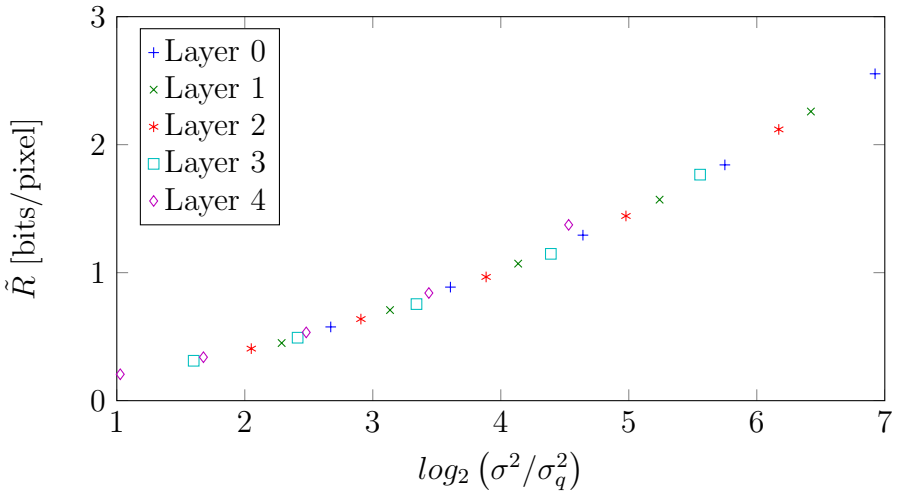


Figure 4.1: Illustration of $\log_2 \frac{\sigma^2}{\sigma_q^2}$ v.s. $\tilde{R}$ for test sequences football.

4.2 Rate model for single layer video

4.2.1 Predicting rate from prediction error and quantization error

For an i.i.d Gaussian signal with variance σ^2, a widely used R-D model that relates the bits per pixel $\tilde{R}$ and the quantization error σ_q^2 is [25]

$$\sigma_q^2 = \epsilon^2 \sigma^2 2^{-2\lambda \tilde{R}}, \tag{4.5}$$

where ϵ and λ are constants depending on the encoding method. For video signals, σ^2 is the error between the source video and the prediction, and σ_q^2 is the error between the source and the reconstruction. Under this model, the rate is given by

$$\tilde{R} = \frac{1}{2\lambda} \log_2 \left(\epsilon^2 \frac{\sigma^2}{\sigma_q^2} \right). \tag{4.6}$$

Note that $\tilde{R}$ only counts the bits used for coding the prediction error, not including the bits for motion and mode information.

Model (4.6) suggests a linear relationship between $\log_2 \frac{\sigma^2}{\sigma_q^2}$ and $\tilde{R}$. Unfortunately, this is only true when the bitrate is high. More details of the rate and distortion relationship at low bitrate are discussed in [45] and [46].

In this work, we have collected $\tilde{R}$, σ_q^2, and σ^2 for all blocks (including the ones coded in Skip mode) from the CIF test sequences described in Sec. 2.4.1. The sequences are encoded with 5 temporal layers, i.e., hierarchical structure with GOP length of 16. 5 QPs are used in the experiment: 16, 20, 24, 28, 32. Fig. 4.1 illustrates $\log_2 \frac{\sigma^2}{\sigma_q^2}$ v.s. $\tilde{R}$ for the test sequence football. It clearly shows a non-linear relationship between $\log_2 \frac{\sigma^2}{\sigma_q^2}$ and $\tilde{R}$. Thus we propose the following model:

$$\tilde{R} = \left(\frac{\sigma^2}{\sigma_q^2}\right)^{\alpha} - \beta, \tag{4.7}$$

with parameters $\alpha < 1$ and $\beta \approx 1$. (4.7) does not differentiate the layers, while (4.8) considers the homogeneity in the layers. Note that in the extreme case where the quantization stepsize q is sufficiently large, everything is quantized to zero. Thus $\sigma_q^2 = \sigma^2$ and $\tilde{R} = 0$, therefore, idealy $\beta = 1$. However, for Skip mode in H.264, the quantization is bypassed, making it probable that $\sigma_i^2 < \sigma_q^2$. Therefore we allow β to deviate slightly in the neighborhood of 1.

Fig 4.3 plots the simulation results for two test sequences with model (4.6) and (4.7). The same data are shown in the left and right columns, but the data in the right column has low bitrate such that the high rate assumption does not hold. At this low rate, (4.6) (the dotted curve) shows a systematic under-estimation for $\tilde{R}$, which could lead to invalid negative $\tilde{R}$. When combining with (4.19), this problem becomes more severe as the error is cumulated exponentially as the number of layers increases.

For the sequence *football*, which has intensive motion, our model (4.7) shows higher accuracy than the conventional model (4.6) in both high and low rate range.

For the sequence *akiyo*, which has a stationary background and consequently a lot of Skip modes, the R-D relationship behaves differently for different temporal layers.

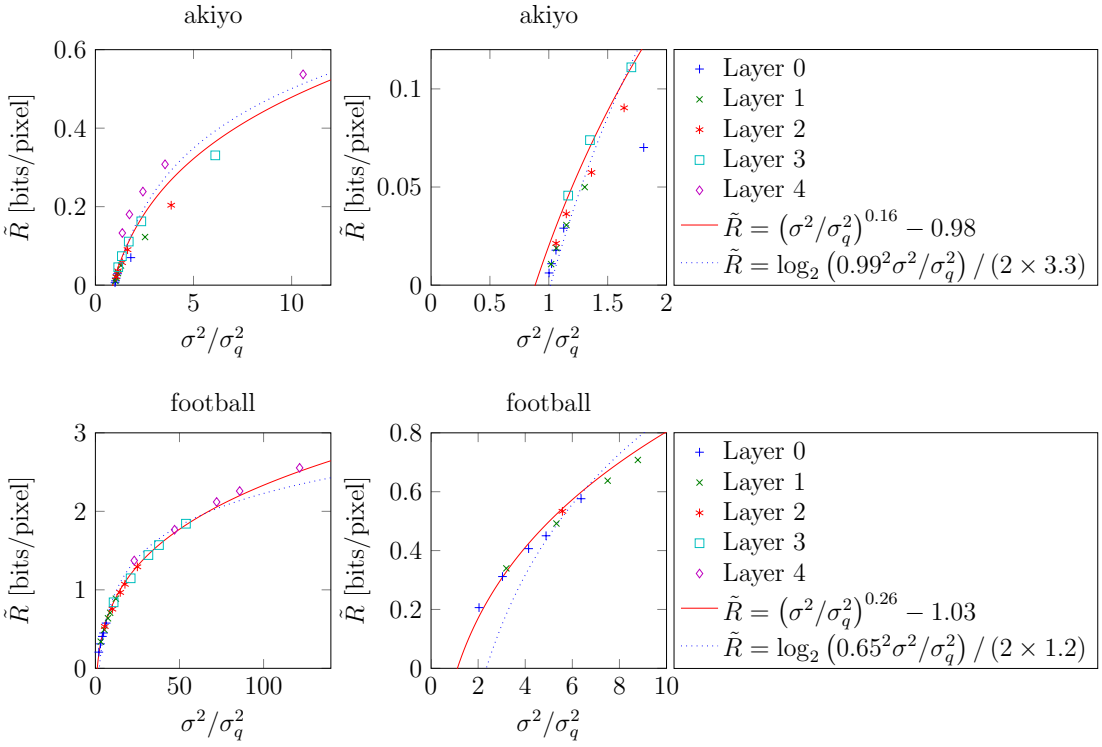


Figure 4.2: Illustration of proposed rate model for test sequences akiyo and football. The points show the measured data. The solid curves are fitted from the data using proposed model (4.7), while the dotted curves are fitted using the conventional model (4.6). The left column shows the entire test range for the test data, while the right column emphasizes on the low rate range.

Because the frames at different temporal layers are predicted from the references that has different frame distance, σ^2 may not be the same among the layers, thus we denote σ_i^2 as the prediction error at l-th layer. However, after quantization, σ_q^2 does not depend on the temporal layer, as all the layers are using the same quantizer. (This will be shown in Sec. 4.4.) Hence for temporal scalability, (4.7) can written as

$$\tilde{R}_i = \left(\frac{\sigma_i^2}{\sigma_q^2}\right)^{\alpha_i} - \beta. \tag{4.8}$$

However, this form introduces too many parameters for all the layers, which is impractical for underlying applications. With a given sequences, these parameters could be predicted from a set of features using similar method as described in our prior works [47, 48]. This

is deferred to future study.

4.2.2 Predicting quantization error from quantization stepsize

In H.264 encoder, the prediction error is transformed using the integer transform, which is very close to the Discrete Cosine Transform (DCT). The transform coefficients are then quantized and coded. For image and video coding, the probability density function (PDF) of the prediction error in the pixel domain is often approximated by a Laplacian distribution with parameter λ:

$$p(x) = \frac{\lambda}{2} \mathrm{e}^{-\lambda|x|}. \tag{4.9}$$

Thus the coefficients in the transform domain have the Cauchy distributions PDF with parameter μ:

$$f(x) = \frac{1}{\pi} \frac{\mu}{\mu^2 + x^2}. \tag{4.10}$$

Given a fixed quantization stepsize q, the quantization error σ_q^2 is

$$\sigma_q^2 = \sum_{i=-\infty}^{\infty} \int_{(i-\frac{1}{2})q}^{(i+\frac{1}{2})q} |x - iq|^2 f(x)\, dx. \tag{4.11}$$

For the Cauchy source (4.10), (4.11) can be derived in a complicated analytical form, which could be aproximated by a power function of q, as reported by Altunbasak *et. al* [49].

In H.264, the quantizer is designed to have a dead-zone, and is also designed differently for Inter- and Intra-modes [50]. However our simulation shows that the power function is still a fairly good approximation for σ_q^2, i.e.,

$$\sigma_q^2 = \gamma q^{\delta}, \tag{4.12}$$

with γ and δ being the model parameters, and q is derived from QP using (1.1).

Fig. 4.3 illustrates the simulation results for two CIF test sequences. As claimed in Sec. 4.2.1, σ_q^2 does not depend on the temporal layer but only on q.

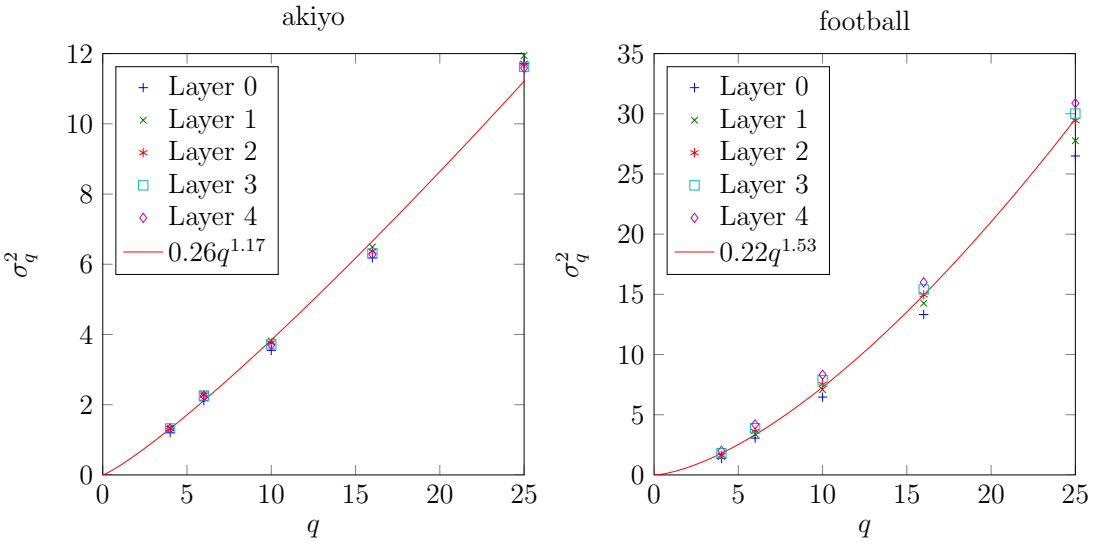


Figure 4.3: σ_q^2 v.s. q for the test sequence akiyo and football coded with H.264. The points show the measured data, and the curves are fitted from the data using (4.12).

4.2.3 Proposed rate model

Combining (4.7) with (4.12), we have the rate model for a single layer video that relates the bitrate $\tilde{R}$ with the prediction error σ^2 and the quantization stepsize q:

$$\tilde{R} = \left(\frac{\sigma^2}{\gamma q^{\delta}}\right)^{\alpha} - \beta. \tag{4.13}$$

In this model, the prediction error σ^2 and the quantization stepsize q are available directly from the encoder during the mode decision stage, while the four model parameters α, β, γ, and δ are sequence dependent, and can be predicted from content features through light-weighted preprocessing, as described in our prior works [47, 48]. With all these parameter obtained by the encoder, (4.13) can be used to guide the mode decision without performing actual coding. The predicting of model parameters and its application in mode decision is deferred as our future work.

4.3 Low-complexity mode decision using proposed rate and distortion model

In this section, we briefly discuss a possible low-complexity mode decision scheme using the proposed rate and distortion model.

For one particular block, the best mode m^* determined by the RDO-based encoder (AVC or SVC) is using

$$m^* = \arg\min_m D(m) + \lambda R(m), \tag{4.14}$$

and the encoder exhaustively searches all available modes to find the one with the lowest R-D cost, which requires to encode this block to obtain D and R, which is too expensive for the low-complexity encoder. A commonly used method is to use estimated D and R instead of real ones.

The quantization error D can be estimated from q using (4.12). For the rate R, it consists of two parts: the header bits R_h, and $\tilde{R}$, which is used to code the prediction error. With the quantization error model (4.12) rate model (4.13), the cost function J can be expressed by

$$J = \gamma q^{\delta} + \lambda(q) M \left(R_h + \left(\frac{\sigma^2}{\gamma q^{\delta}} \right)^{\alpha} - \beta \right), \tag{4.15}$$

where M denotes the number of pixels in that block. Due to the complicated entropy coding, R_h is not easy to model. Modeling of R_h and the implementation of such mode decision algorithm are deferred as our future study.

4.4 Rate model for temporal scalability

In this section, we apply the single layer video rate model (4.13) to SVC coded video that adopts temporal scalability. For scalable video, in order to decode a given layer, the dependent layers (e.g., all the lower layers for a dyadic temporal structure illustrated in

Fig. 1.7) are also needed. Thus for a given layer, its bitrate includes that in all dependent layers.

In our prior research works collaborated with Ma *et. al* [47, 48], a novel rate model has been proposed for both AVC and SVC considering the impact of both quantization stepsize q and the frame rate t, i.e.,

$$R(q,t) = R_{\max}\left(\frac{q}{q_{\min}}\right)^{-a}\left(\frac{t}{t_{\max}}\right)^{b}, \tag{4.16}$$

where the constants $q_{\min}$ and $t_{\max}$ is determined by the encoding configuration of the underlying application, and $R_{\max}$ denotes the actual rate when coding the video at $q_{\min}$ and $t_{\max}$. a and b are the model parameters relating q and t, respectively. With this model, the bitrate for a specific temporal layer coded using given QP can be accurately estimated. However, even within the same layer, the frame bitrate may still fluctuate. To resolve this issue, we first remove he bitrate contributed by lower layers, then apply the single layer rate model (4.13) to the net bitrate of current layer.

For a video coded with L dyadic temporal layers, the total bitrate is the sum of the rates at each layer, i.e.,

$$R = \sum_{i=0}^{L-1} R_i, \tag{4.17}$$

where R_i is the rate at i-th temporal layer. Denote M as the number of pixels per frame, t_i and $\tilde{R}_i$ as the frame rate and the rate measured as bits per pixel at layer i, then the bitrate for l-th temporal layer (including all lower layers) is

$$R_l = \sum_{i=0}^{l} \tilde{R}_i M t_i. \tag{4.18}$$

Because of the dyadic coding structure in the temporal scalability, we have $t_1 = t_0$, and $t_i = 2t_{i-1} = t_0 2^{i-1}$ for $i > 1$. Thus (4.18) can be written as

$$R_l = \tilde{R}_0 M t_0 + \sum_{i=1}^{l} \tilde{R}_i M t_0 2^{i-1}. \tag{4.19}$$

With this expression, we can focus on $\tilde{R}_i$ that depends only on the current layer rather than the lower layers.

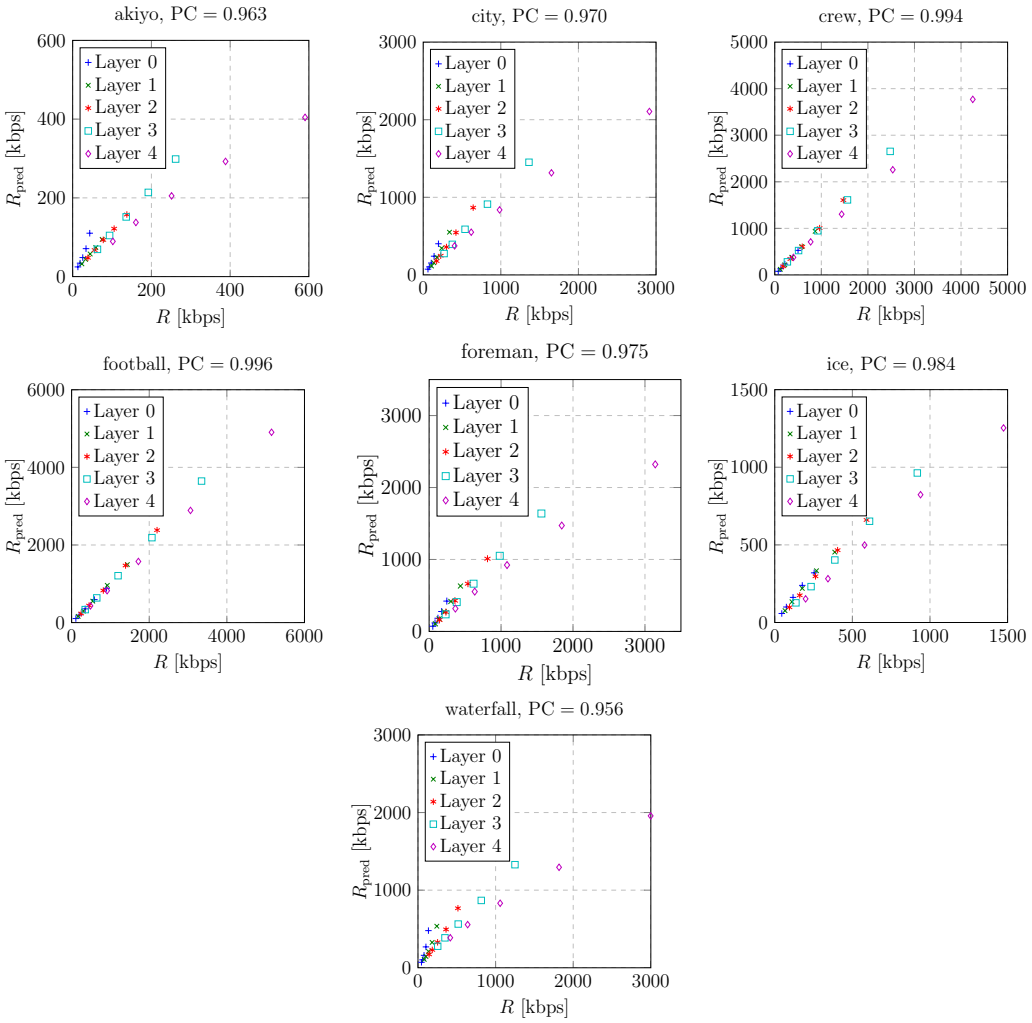


Figure 4.4: Evaluation for proposed rate model with seven test sequences. R denotes the actual bitrate measured at each temporal layer and each QP, and R_{pred} is calculated from (4.20) using the fitted parameters.

Combining with (4.18), the rate for temporal layer l (including the lower layers) is

$$R_l = \sum_{i=0}^{l} \left(\left(\frac{\sigma^2}{\gamma q^{\delta}} \right)^{\alpha} - \beta \right) M t_i. \tag{4.20}$$

The proposed model (4.20) is evaluated on two CIF test sequences, coded with dyadic temporal scalability with GOP length of 16 (i.e., containing 5 temporal layers), with each layer coded with the same QP, as illustrated in Fig. 4.4. 5 QP values are tested: 16, 20, 24, 28, 32. The model parameters for the test sequences are listed in Table 4.1.

Table 4.1: Model parameters for seven CIF sequences.

Sequence	α	β	γ	δ
akiyo	0.16	0.87	0.26	1.17
city	0.32	0.32	0.26	1.56
crew	0.33	1.00	0.26	1.38
football	0.26	1.03	0.22	1.53
foreman	0.30	0.90	0.29	1.42
ice	0.14	0.99	0.21	1.24
waterfall	0.38	0.84	0.28	1.50

In this experiment, the measured bitrate contains the header bits that is not captured in the proposed model. However, the results still show high Pearson correlation (PC) for the seven test sequences (above 0.95 for all sequences).

4.5 Summary and discussions

In this chapter, we present a rate model that predicts the video bitrate for coding the prediction error from the prediction error (in terms of mean square error) and the quantization stepsize q. The prediction error can be calculated for each block to yield the estimated rate for block given the prediction error associated with a particular prediction mode, or at the frame level, given the expected prediction error. While the conventional rate model fails in the low-bitrate range, our model works for both low and high rate. The proposed model relies on four parameters, which can be either directly obtained from the encoder, based on the rate obtained from previously coded blocks, or predicted by video content features through preprocessing. The basic rate model that predicts the bits needed to code each pixel is also used to derive the total bitrate required in a temporal scalable encoder using hierarchical B-structure. The simulation for the proposed model shows high Pearson correlation between the predicted rate and the measured actual rate.

Chapter 5

Conclusion

In this thesis, we investigate the complexity in current scalable video encoder by scrutinizing the mode decision algorithm, and propose a low-complexity multilayer mode decision scheme, with marginally worse or better coding efficiency. We also model the video bitrate with the prediction error and quantization stepsize.

First, we examine the current mode decision scheme adopted by the JSVM encoder with rate-distortion optimization enabled, where the computational intensive motion estimation and mode decision is performed repeatedly with low efficiency. We propose a novel multilayer mode decision scheme, where the encoder performs the mode decision (including motion estimation) only once for all layers, and this near-optimal mode (and motion vectors for Inter-mode) are used by all layers. This common mode is determined at the base layer, while all higher layers simply reuse the lower layer mode using the inter-layer mode prediction tool provided in H.264/SVC. In order to determine the common mode at the base layer when the higher layers have not been coded, we tune the mode decision configuration toward the highest layer, i.e., using the reference frame used by the highest layer, and using QP from the highest layer to derived Lagrangian parameter used in motion estimation and Intra-prediction mode decision. The Lagrangian multiplier for mode decision at current layer is derived using the current layer QP, to provide

near-optimal mode for current layer. Simulation results show significant complexity reduction at the enhancement layers.

Second, we further reduce the complexity at the base layer via the use of proposed early Skip/Direct technique. We unify the notation of Skip and Direct modes in P- and B-frames, and then apply the early stop technique to terminate the mode decision at an early stage (without computational intensive motion estimation) if certain conditions are satisfied. The prediction error (in terms of sum of absolute error of all pixels) of an 8×8 sub-block is used as the early stop threshold. For a given macroblock, if the prediction errors (using the predicted motion vector) are below the thresholds in all sub-blocks, the mode decision is terminated at an early stage, with Skip or Direct mode selected as the best mode. For early Skip determination, both luma and chroma components are examined, whereas for Direct mode, only the luma component is checked. The proposed method applies for both AVC and SVC. When integrated with the presented multilayer mode decision scheme for SVC, the algorithms is slightly adjusted at the enhancement layers. We perform a light-weighted motion estimation using only 16×16 block size, in the case that the motion estimation has never been conducted in the lower layers (i.e., the early stop conditions are satisfied in all lower layers). With this scheme, the motion estimation does not necessarily take place at the lowest layer, but is still conducted at most once throughout all layers. Simulation results show additional complexity reduction when the proposed early stop technique is enabled.

Third, we investigate the rate and distortion modeling, and find analytical forms to match the measured data. The proposed rate model relates the video bitrate with the prediction error and the quantization stepsize. The model requires four parameters, which could be predicted by content features obtained via light-weighted preprocessing. A possible low-complexity mode decision method is also presented using the proposed model. Moreover, the proposed model is applied on temporal scalable video coding, and the simulation results show high Pearson correlation between the measured rates and

the ones predicted by the proposed model.

Bibliography

[1] "Cisco Visual Networking Index: Forecast and Methodology, 2012-2017," May 2013. [Online]. Available: http://www.cisco.com/en/US/solutions/collateral/ns341/ns525/ns537/ns705/ns827/white_paper_c11-481360.pdf

[2] *Video codec for audiovisual services at* $p \times 64$ *kbit/s*, Rec. ITU-T H.263 Std., 1988–1993.

[3] *Information technology – Coding of moving pictures and associated audio for digital storage media at up to about 1,5 Mbit/s – Part 2: Video*, ISO/IEC 11172-2 (MPEG-1 Video) Std., 1993–2006.

[4] *Information technology – Generic coding of moving pictures and associated audio information: Video*, Rec. ITU-T H.262 and ISO/IEC 13818-2 (MPEG-2 Video) Std., 1996–2011.

[5] *Video coding for low bit rate communication*, Rec. ITU-T H.263 Std., 1996–2005.

[6] *Information technology – Coding of audio-visual objects – Part 2: Visual*, ISO/IEC 14496-2 (MPEG-4 Visual) Std., 1999–2009.

[7] *Advanced Video Coding for Generic Audiovisual Services*, Rec. ITU-T H.264 and ISO/IEC 14496-10 (MPEG-4 AVC) Std., Rev. 8.0, 2003–2013.

[8] *High efficiency video coding*, Rec. ITU-T H.265 and ISO/IEC 23008-2 (MPEG-H HEVC) Std., Rev. 1.0, 2013.

[9] *Studio encoding parameters of digital television for standard 4:3 and wide screen 16:9 aspect ratios*, Rec. ITU-R BT.601 Std., 1982–2011.

[10] T. Wiegand, G. J. Sullivan, G. Bjontegaard, and A. Luthra, "Overview of the H.264/AVC video coding standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, no. 7, pp. 560–576, 2003.

[11] H. Schwarz, D. Marpe, and T. Wiegand, "Overview of the scalable video coding extension of the H.264/AVC Standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 17, no. 9, pp. 1103–1120, Sep. 2007.

[12] *ITU-T and ISO/IEC JTC 1, Reference Software for Scalable Video Coding*, ITU-T Rec. H.264.2 (reference software for ITU-T H.264 advanced video coding) and ISO/IEC 14496-5:Amd 24 (reference software for scalable video coding), 2009–2013.

[13] X. Li, P. Amon, A. Hutter, and A. Kaup, "Lagrange multiplier selection for rate-distortion optimization in SVC," in *Picture Coding Symposium.* IEEE, 2009, pp. 1–4.

[14] H. Schwarz and T. Wiegand, "R-D optimized multi-layer encoder control for SVC," in *Proc. IEEE Int'l Conf. on Image Process.*, vol. 2, 2007, pp. II–281.

[15] X. Li, P. Amon, A. Hutter, and A. Kaup, "One-pass multi-layer rate-distortion optimization for quality scalable video coding," in *Proc. IEEE Int'l Conf. on Acous., Speech, and Signal Process.*, 2009, pp. 637–640.

[16] D. Alfonso, M. Gherardi, A. Vitali, and F. Rovati, "Performance analysis of the scalable video coding standard," in *Packet Video 2007.* IEEE, 2007, pp. 243–252.

[17] B. Lee, M. Kim, S. Hahm, I.-J. Cho, and C. Park, "A low complexity encoding scheme for coarse grain scalable video coding," in *Proc. of IET Int'l Conf. on Visual Inf. Eng.*, 2008, pp. 753–758.

[18] C. S. Park, B. K. Dan, H. Choi, and S. J. Ko, "A statistical approach for fast mode decision in scalable video coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 19, no. 12, pp. 1915–1920, 2009.

[19] S.-T. Kim, K. Konda, C.-S. Park, C.-S. Cho, and S.-J. Ko, "Fast mode decision algorithm for inter-layer coding in scalable video coding," *IEEE Trans. Consum. Electron.*, vol. 55, no. 3, pp. 1572–1580, 2009.

[20] Z. Deng, X. Cai, and Y. Cui, "Fast mode decision algorithm for inter-layer intra prediction in SVC," in *Proc. IEEE Broadband Netw. and Multimedia Technol.*, 2011, pp. 212–216.

[21] T. Wiegand, M. Lightstone, D. Mukherjee, T. G. Campbell, and S. K. Mitra, "Rate-distortion optimized mode selection for very low bit rate video coding and the emerging h. 263 standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 6, no. 2, pp. 182–190, 1996.

[22] G. J. Sullivan and T. Wiegand, "Rate-distortion optimization for video compression," *IEEE Signal Process. Mag.*, vol. 15, no. 6, pp. 74–90, 1998.

[23] T. Wiegand, H. Schwarz, A. Joch, F. Kossentini, and G. Sullivan, "Rate-constrained coder control and comparison of video coding standards," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, no. 7, pp. 688–703, 2003.

[24] J. Reichel, H. Schwarz, and M. Wien, "Joint Scalable Video Model 11 (JSVM 11)," in *Joint Video Team, Doc. JVT-X202*, Jul. 2007.

[25] Y. Wang, J. Ostermann, and Y.-Q. Zhang, *Video processing and communications*. Prentice Hall Upper Saddle River, 2002, vol. 5.

[26] Joint Video Team. [Online]. Available: http://wftp3.itu.int/av-arch/jvt-site/

[27] X. Li, P. Amon, A. Hutter, and A. Kaup, "Performance analysis of inter-layer prediction in scalable video coding extension of H. 264/AVC," *IEEE Trans. Broadcast.*, vol. 57, no. 1, pp. 66–74, 2011.

[28] *ITU-T and ISO/IEC JTC 1, Reference Software for H.264*, ITU-T Rec. H.264.2 (reference software for ITU-T H.264 advanced video coding) and ISO/IEC 14496-5: (Information technology – Coding of audio-visual objects – Part 5: Reference software, 2009–2012.

[29] B. Jeon and J. Lee, "Fast mode decision for H.264," in *Joint Video Team, Doc. JVT-J033*, Dec. 2003.

[30] M. Bystrom, I. Richardson, and Y. Zhao, "Efficient mode selection for H. 264 complexity reduction in a Bayesian framework," *Signal Processing: Image Communication*, vol. 23, no. 2, pp. 71–86, 2008.

[31] C. S. Kannangara, I. E. Richardson, M. Bystrom, J. R. Solera, Y. Zhao, A. MacLennan, and R. Cooney, "Low-complexity skip prediction for H.264 through Lagrangian cost estimation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 16, no. 2, pp. 202–208, 2006.

[32] Y. Ivanov and C. Bleakley, "Skip prediction and early termination for fast mode decision in H.264/AVC," 2006, pp. 7–7.

[33] A. Saha, K. Mallick, J. Mukherjee, and S. Sural, "SKIP prediction for fast rate distortion optimization in H. 264," *IEEE Trans. Consum. Electron.*, vol. 53, no. 3, pp. 1153–1160, 2007.

[34] L. Shen, Y. Sun, Z. Liu, and Z. Zhang, "Efficient SKIP mode detection for coarse grain quality scalable video coding," *IEEE Signal Process. Lett.*, vol. 17, no. 10, pp. 887–890, 2010.

[35] A. M. Tourapis, F. Wu, and S. Li, "Direct mode coding for bipredictive slices in the H.264 standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 15, no. 1, pp. 119–126, 2005.

[36] "Direct macroblock coding for predictive(P) pictures in the H.264 standard, author="tourapis, alexis m and wu, feng and li, shipeng", booktitle="electronic imaging 2004", pages="364–371", year="2004", organization="international society for optics and photonics"."

[37] H.-C. Lin, W.-H. Peng, H.-M. Hang, and W.-J. Ho, "Layer-adaptive mode decision and motion search for scalable video coding with combined coarse granular scalability

(CGS) and temporal scalability," in *Proc. IEEE Int'l Conf. on Image Process.*, vol. 2. IEEE, 2007, pp. II–289.

[38] E. Akyol and M. van der Schaar, "Complexity model based proactive dynamic voltage scaling for video decoding systems," *IEEE Trans. Multimedia*, vol. 9, no. 7, pp. 1475–1492, 2007.

[39] Z. Ma, H. Hu, and Y. Wang, "On complexity modeling of H.264/AVC video decoding and its application for energy efficient decoding," *IEEE Trans. Multimedia*, vol. 13, no. 6, pp. 1240–1255, 2011.

[40] W. Ding and B. Liu, "Rate control of MPEG video coding and recording by rate-quantization modeling," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 6, no. 1, pp. 12–20, 1996.

[41] T. Chiang and Y.-Q. Zhang, "A new rate control scheme using quadratic rate distortion model," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 7, no. 1, pp. 246–250, 1997.

[42] T. Chiang, H.-J. Lee, and H. Sun, "An overview of the encoding tools in the MPEG-4 reference software," in *Proc. IEEE Int'l. Symp. on Circuit and Syst.*, vol. 1. IEEE, 2000, pp. 295–298.

[43] J. Ribas-Corbera and S. Lei, "Rate control in DCT video coding for low-delay communications," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 9, no. 1, pp. 172–185, 1999.

[44] Z. He and S. Mitra, "A novel linear source model and a unified rate control algorithm for H.264/MPEG-2/MPEG-4," in *Proc. IEEE Int'l Conf. on Acous., Speech, and Signal Process.* IEEE, 2001.

[45] T. Wiegand, "Rate Distortion Theory and Quantization," Lecture note. [Online]. Available: http://iphome.hhi.de/wiegand/assets/pdfs/DIC_rd_theory_quantization_07.pdf

[46] B. Bunin, "Rate-Distortion Function for Gaussian Markov Processes," Alcatel-Lucent, Tech. Rep. 9, Nov. 1969. [Online]. Available: http://www.alcatel-lucent.com/bstj/vol48-1969/articles/bstj48-9-3059.pdf

[47] Z. Ma, M. Xu, Y.-F. Ou, and Y. Wang, "Modeling of rate and perceptual quality of compressed video as functions of frame rate and quantization stepsize and its applications," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 22, no. 5, pp. 671–682, 2012.

[48] Z. Ma, H. Hu, M. Xu, and Y. Wang, "Rate model for compressed video considering impacts of spatial, temporal and amplitude resolutions and its applications for video coding and adaptation," *arXiv preprint arXiv:1206.2625*, 2012.

[49] Y. Altunbasak and N. Kamaci, "An analysis of the dct coefficient distribution with the h.264 video coder," in *Proc. IEEE Int'l Conf. on Acous., Speech, and Signal Process.*, vol. 3. IEEE, 2004, pp. 177–180.

[50] "Quantization Techniques in JM/KTA Part 2." [Online]. Available: http://www.h265.net/2009/06/quantization-techniques-in-jmkta-part-2.html

List of Abbreviations and Notations

SD	Standard-Definition
HD	High-Definition
CIF	Common Intermediate Format (Resolution: 352×288)
720p	720 Lines Progressive (Resolution: 1280×720)
AVC	Advanced video coding (H.264 single-layered video coding)
SVC	Scalable Video Coding (H.264 scalable video coding)
CGS	Coarse Grain Scalability
MGS	Medium Grain Scalability
I-	Intra- (the prediction is from the self frame)
P-	Predictive- (the prediction is from the previous frame)
B-	Bi-predictive- (the prediction is from both previous and future frames)
GOP	Group-Of-Pictures
MB	Macroblock (consisting of 16×16 pixels)
ME	Motion Estimation
MV	Motion Vector
PMV	Predicted motion vector
QP	Quantization Parameter
R-D	Rate-Distortion
RDO	Rate-Distortion Optimization
SAD	Sum of Absolute Difference
SATD	Sum of Absolute Transformed Difference
SSE	Sum of Squared Error
i.i.d	Independent and identically distributed
PC	Pearson Correlation
PDF	Probability Density Function
PSNR	Peak-Signal-to-Noise-Ratio
SNR	Signal-to-Noise-Ratio

R	Bitrate
D	Distortion
$D_{\mathbf{Inter}}$	Distortion in Inter-prediction
$D_{\mathbf{Intra}}$	Distortion in Intra-prediction
J	R-D cost function
$J_{\mathbf{Inter}}$	R-D cost function to evaluate the Inter-mode associated with different MVs
$J_{\mathbf{Intra}}$	R-D cost function to evaluate the Intra-mode associated with different prediction methods
f	Original signal
$\hat{f}$	Reconstructed signal
m	Macroblock mode
v	Motion vector
λ	Lagrangian parameter used in mode decision
$\lambda_{\mathbf{Inter}}$	Lagrangian parameter used in motion search
$\lambda_{\mathbf{Intra}}$	Lagrangian parameter used in Intra-mode decision
L	Total number of quality layers
q	Quantization stepsize

List of Publications

1. M. Xu, Z. Ma, and Y. Wang, "Low-complexity mode decision for quality scalable video coding," submitted to *IEEE Trans. Circuits Syst. Video Technol.*, on 01/09/2014.

2. M. Xu, Z. Ma, and Y. Wang, "One-pass mode decision for low-complexity and high-efficiency encoding of quality scalable video," in *Proc. of IEEE Int'l Conference on Multimedia and Expo. Workshop*, Jul. 2013, pp. 1–5.

3. Z. Ma, M. Xu, Y.-F. Ou, and Y. Wang, "Modeling of rate and perceptual quality of compressed video as functions of frame rate and quantization stepsize and its applications," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 22, no. 5, pp. 671–682, 2012.

4. Z. Ma, H. Hu, M. Xu, and Y. Wang, "Rate model for compressed video considering impacts of spatial, temporal and amplitude resolutions and its applications for video coding and adaptation," *arXiv preprint arXiv:1206.2625*, 2012.

5. Z. Ma, M. Xu, K. Yang, and Y. Wang, "Modeling rate and perceptual quality of video and its application to frame rate adaptive rate control," in *Proc. IEEE Int'l Conf. on Image Process.*, IEEE, 2011, pp. 3321 – 3324.