

國立交通大學

電子工程學系

碩士論文

H.264/AVC 之碼率控制技術研究



**A Study on Rate Control Techniques of
H.264/AVC**

指導教授：王聖智 博士

研究生：蔣宗翰

中華民國九十五年六月

H.264/AVC 之碼率控制技術研究

A Study on Rate Control Techniques of H.264/AVC

研究生：蔣宗翰

Student: Tsung-Han Chiang

指導教授：王聖智

Advisor: Sheng-Jyh Wang



A Thesis

Submitted to Department of Electronics Engineering & Institute of Electronics
College of Electrical and Computer Engineering
National Chiao Tung University
in partial Fulfillment of the Requirements
for the Degree of Master
In Electronics Engineering
June 2006
HsinChu, Taiwan, Republic of China

中華民國九十五年六月

H.264/AVC 之碼率控制技術研究

研究生：蔣宗翰

指導教授：王聖智 博士

國立交通大學

電子工程學系 電子研究所碩士班

摘要

在本文中，我們針對影像編碼當中編碼端一個相當重要的部分作研究。碼率控制主要目的是藉由調整編碼而達到預期的資料大小，我們的討論將建構在 H.264/AVC 標準上，首先分析量化參數、移動補償資料與壓縮後資料量之間的關係，進而針對壓縮後的檔頭資料作分析，然後重新建立一個針對 H.264/AVC 編碼特性的碼率失真模型。對於每張畫面的位元配置，我們也利用前後張影像的資料關係去調整，以改善影像品質跟穩定度。最後，為了改良原本使用在 JM 上的 MAD 值預測，我們利用移動向量來預測每個大區塊的 MAD 值。整合上述的方法，能改善原本在低碼率時，在緩衝器上的不佳效果，特別是藉由較精準的碼率失真模型能預測出準確的量化參數。經過實驗，可以發現在緩衝器的穩定性上能有不錯的效果，在影像品質上更能獲得明顯的效果。

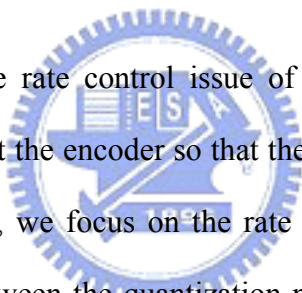
A Study on Rate Control Techniques of H.264/AVC

Student: Tsung-Han Chiang

Advisor: Dr. Sheng-Jyh Wang

Institute of Electronics
National Chiao Tung University

Abstract



In this thesis, we study the rate control issue of a video compression system. The purpose of rate control is to adjust the encoder so that the number of encoded bits can match the number of desired bits. Here, we focus on the rate control of an H.264/AVC encoder. First, we analyze the relation between the quantization parameter, MAD, and the coded bit number. We also analyze the encoding of header bits. Based on these analyses, we build up a new rate-distortion model. The bit allocation of each picture is adaptively determined depending on the relations among the picture and its front and rear pictures. This adaptive change can improve the quality and stability of the coded videos. Finally, we use motion vectors to predict the MAD value, which is originally predicted by the JM model. By combining the above methods, we can improve the performance of the buffer in low bit-rate encoding. Especially, with a more accurate R-D model, we can better predict the quantization parameter. In experiments, we show that our approach not only improves the stability of encoder buffer but also makes an obvious improvement of visual quality.

誌謝

能完成這篇論文，首先我要感謝我的指導教授王聖智老師，來交大的兩年研究所生涯中，在老師的指導之下學習到做研究的方法，從什麼都不懂開始慢慢學習到能寫完這篇論文，另外更讓我學習到許多研究之外的人生道理。除此之外，還要感謝實驗室的學長跟學姊，讓我在交大的生活不會有孤單的感覺，也從你們身上學習到了不少東西。更重要的是要感謝我的家人，我的父母，姐姐和外婆，提供這十幾年的求學生涯，讓我能專心的學習不需要煩惱其他事情。

最後要感謝美詩這兩年的陪伴，在我研究煩悶時有你默默的在背後支持我，最後也要感謝 KERORO 在我研究煩悶閒暇時，能讓我開心的充電休息，持續的支持到最後，謝謝。

Contents

摘要	i
Abstract.....	ii
誌謝	iii
Figures	v
Tables.....	vii
Chapter 1 Introduction.....	1
Chapter 2 Background.....	2
2.1 Introduction to video compression systems	2
2.2 Introduction to H.264/AVC	4
2.2.1 Highlights of H.264/AVC	5
2.2.2 Video Coding Layer (VCL).....	6
2.2.3 H.264/AVC Profile	12
2.3 Introduction to Rate Control.....	15
2.3.1 Rate Control of H.264/AVC	16
2.3.2 H.264/AVC Rate Control Scheme.....	19
Chapter 3 Modified Rate-Distortion Model for H.264/AVC.....	24
3.1 Previous R-D Models	24
3.2 Our Rate-Distortion Model.....	27
3.3 The Relation between Header Bits and Macroblock Mode.....	49
3.4 Frame Level Header Bits Prediction	51
Chapter 4 Bit allocation of H.264/AVC	54
4.1 Frame Level.....	55
4.1.1 Frame Complexity	56
4.1.2 Frame Importance.....	59
4.1.3 Selection different QP causes the MAD change of next frame	61
4.2 Macroblock Level.....	66
4.2.1 MAD prediction from forward frame (add motion vector).....	67
4.2.2 A solution of coding too many bits in picture level.....	70
Chapter 5 Conclusion	74
Bibliography	75

Figures

Figure 2-1 Block diagram of a typical video encoder [2]	3
Figure 2-2 Structure of H.264/AVC [3]	4
Figure 2-3 Scope of H.264/AVC standardization [3].....	4
Figure 2-4 Basic coding structure for H.264/AVC for a macroblock [3]	6
Figure 2-5 Possible subdivisions of a picture into slices [3]	7
Figure 2-6 Possible subdivisions of a picture into slices with FMO.	7
Figure 2-7 Switching streams using I-slice and SP-slices [4].....	8
Figure 2-8 Five of nine Intra 4×4 prediction modes [3]	9
Figure 2-9 Four Intra 16×16 prediction modes [5]	9
Figure 2-10 Decomposition of macroblock for motion compensation [3]	9
Figure 2-11 Filtering for fractional-sample accurate motion compensation [3].....	10
Figure 2-12 Package of the DCT DC values [5].....	11
Figure 2-13 H.264/AVC Profiles [3].....	13
Figure 2-14 Illustration of the H.264/MPEG4-AVC FRExt profiles [6].....	14
Figure 2-15 Block diagram of an encoder with rate control [1]	15
Figure 2-16 Block diagram of H.264/AVC Rate Controller [10]	17
Figure 3-1 Relation between SAD and the number of generated bits [14].....	25
Figure 3-2 Approximated relation between SAD and the number of generated bits [14].	25
Figure 3-3 The relation between the coded coefficient bits and $1/Q_{step}$ for “news” [15].....	26
Figure 3-4 The relation between the coded coefficient bits and $1/Q_{step}$ for “foreman” [15]...	26
Figure 3-5 (a) Relation between bits and QP, (b) Relation between bits and Q_{step}	27
Figure 3-6 The relation between the number of coded bits and QP. The pink curves show the fitting of a second-order polynomial. (a)MAD=0.7695 (b)MAD=2.7383 (c)MAD=3.9297	29
Figure 3-7 The relation between the coefficient “a” and MAD in different test sequences. ...	29
Figure 3-8 The relation between the coefficient “b” and MAD in different test sequences. ...	30
Figure 3-9 The relation between the coefficient “c” and MAD in different test sequences. ...	31
Figure 3-10 The relation between MAD and the “zero point”	32
Figure 3-11 The buffer fullness and target buffer level for the “bus” sequence when the bit rates are 64K and 128K.....	34
Figure 3-12 The buffer fullness and target buffer level for the “flower” sequence when the bit rates are 64K and 128K.....	34
Figure 3-13 The buffer fullness and target buffer level for the “highway” sequence when the bit rates are 64K and 128K.	35
Figure 3-14 The buffer fullness and target buffer level for the “Stefan” sequence when the bit rates are 64K and 128K.....	35

Figure 3-15. The number of basic units have no remaining target bits when the bit rate is 64K	36
Figure 3-16 Compare the buffer fullness and the number of basic units that have no remaining target bits	37
Figure 3-17 The PSNR(Y) of each frame in this experiment.....	39
Figure 3-18 Comparison of visual quality when the bit rate is 64K,	40
Figure 3-19 The receiver timing analysis when the bit rate is 128K.....	41
Figure 3-20 Comparison of buffer fullness with RDO and without RDO as the bit rate is 64K.	43
Figure 3-21 The buffer fullness when the period of I-frame is 30	45
Figure 3-22 The PSNR of each frame when the period of I-frame is 30	47
Figure 3-23 The relation between the header bits and the MAD value of the macroblock.....	49
Figure 3-24 The relation between the macroblock header bits and the macroblock MAD for different numbers of motion vectors.	50
Figure 3-25 Proposed frame-level header bits prediction	52
Figure 4-1 The diagram of the bit allocation for picture level and basic-unit level.....	54
Figure 4-2 The relations between the number of coded bits and MAD	56
Figure 4-3 The relations between the number of coded bits and the difference between the number of 8×8 macroblocks and the number of 16×16 macroblocks.....	57
Figure 4-4 The target buffer level control, (a) the normal target buffer level, (b) non-linear target buffer level.....	59
Figure 4-5 The experiment of change frame importance. (a)flower sequence (b)mobile sequence	60
Figure 4-6 The data of the following pictures when coding the current picture with different QP's.	61
Figure 4-7 The flow chart of our strategy in QP decision	62
Figure 4-8 The relation between the value of MAD and the value of QP.....	63
Figure 4-9 The relation between the value of m and MAD	63
Figure 4-10 The PSNR of each pictures,. The sequence is “salesman” and the bit rate is 64K.	64
Figure 4-11 The PSNR of each picture.....	65
Figure 4-12 The relation between the coded bits and MAD in macroblock level.....	66
Figure 4-13 The diagram of macroblock motion.....	67
Figure 4-14 Our method to predict the new MAD value	68
Figure 4-15 The PSNR of each picture with the “silent” sequence when the bit rate is 16K.	68
Figure 4-16 The buffer fullness with the “silent” sequence when the bit rate is 16K.....	69
Figure 4-17 Compare the buffer fullness and the number of macroblock that is coded without using R-D model.....	70
Figure 4-18 The illustration of our strategy in changing bit allocation.....	71

Figure 4-19 The buffer fullness 71
Figure 4-20 The PSNR of each picture..... 72
Figure 4-21 The flow chart of our algorithm 73



Tables

Table 2-1 Summary of Symbols	20
Table 3-1 Table of experiment factors	33
Table 3-2 The delay for real time transmission	41
Table 3-3 The precision of R-D model	42
Table 3-4 The precision of R-D model when execute RDO	44
Table 3-5 The delay for real time transmission when the period of I-frame is 30	48
Table 3-6 The precision of R-D model when the period of I-frame is 30	48
Table 3-7 The precision of R-D model in high and low bit rate	48
Table 3-8 The elements of macroblock header bit in P-slice	50
Table 3-9 The experiment of the header bits prediction	53
Table 4-1 The correlation coefficients table	58
Table 4-2 The PSNR of this experiment	64
Table 4-3 The PSNR stability of this experiment	65



Chapter 1 Introduction

Rate control plays an important role in video encoders. Without rate control, the client buffer may face underflow or overflow because of the mismatch between the source bit rate and the available channel bandwidth for delivering compressed bitstreams. Hence, without rate control, it would be hard to use the video coding encoder in practice. Existing video coding standards usually have their own non-normative rate control schemes during the standardization process. For example, H.264/AVC has a rate control scheme, called JM (Joint Model).

Today, rate control has become one important research topic in the fields of video compression and transmission. In terms of the operational unit, rate control schemes can be classified into macroblock-, slice-, or frame-layer rate control. These rate control schemes usually resolve two main problems. The first problem is the bit allocation problem, which is used to predict the coded bits. The second problem is about how to properly adjust the encoder parameters, like the adjustment of the quantization parameter, to encode each unit with the allocated bits.

The rate allocation is usually associated with a buffer model, which is specified in the video coding standard. The hypothetical reference decoder (HRD) is usually a normative part to represent a set of normative requirements. An HRD-compliant bit stream must be decoded in the constant bit-rate (CBR) without overflow and underflow.

On the other hand, the quantization parameter adjustment is used to find the relation between the bitrates and the quantization parameter. This relation is usually defined as a rate-distortion (R-D) model.

In this thesis, we propose a more accurate R-D model to adjust the quantization parameter adjustment. In the bit allocation, we provide different bit allocation strategies in picture-level and macroblock-level encoding. This thesis is organized as follows. In Chapter 2, the backgrounds about H.264/AVC standard and rate control are briefly reviewed. In Chapter 3, our new R-D model is analyzed statistically and theoretically. Our bit allocation methods are discussed in Chapter 4. Finally, Chapter 5 concludes this thesis.

Chapter 2 Background

In this chapter, we'll first introduce the basic elements of a video compression system. Second we'll give an overview of the H.264/AVC video coding standard [1], which is a very efficient video codec. Our research will be built upon this video standard. Finally, we'll introduce the rate control issue of H.264/AVC in details.

2.1 Introduction to video compression systems

Why do we need video compression? Here we show a simple example. Assume a video sequence is of the QCIF size and is captured in the RGB format with the rate of 30 frames per second. Then, how many bits will a ten-minute video contain? The answer is

$$10 \times 60 \times 30 \times 176 \times 144 \times 3 \times 8 = 10,948,608,000 \text{ bits} = 1.3 \text{ GB} !!$$

On the other hand, after video compression encoding, the data size can be reduced to be around 10MB if based on the H.264/AVC standard. This example demonstrates the importance of a video compression system.

In video encoding, some typical techniques have been used. Here we list the major four techniques.

1. Prediction Coding:

This technique uses the information among frames to compress video data. The prediction coding could be classified into two major modes: intra prediction and inter prediction. The former uses the spatial information for prediction, while the latter uses the temporal information for prediction. A typical way for temporal prediction is the motion estimation operation.

2. Transform Coding:

This technique is based on the property that some suitable transformations may cause the energy of the transformed data to be more compact. At present, two most

commonly used transformations are the DCT transform and the wavelet transform.

3. Quantization:

The previous two techniques, prediction coding and transform coding, are usually designed to be lossless. This quantization operation, however, performs lossy coding. The technique converts the transformed data into quantized data. The quantization step size is usually determined by some control parameters, like the quantization parameter (QP).

4. Entropy Coding:

Entropy coding encodes the quantized data based on Shannon’s information theory. Two frequently used methods are Huffman Coding and Arithmetic Coding.

In summary, we show a general video coding scheme as below.

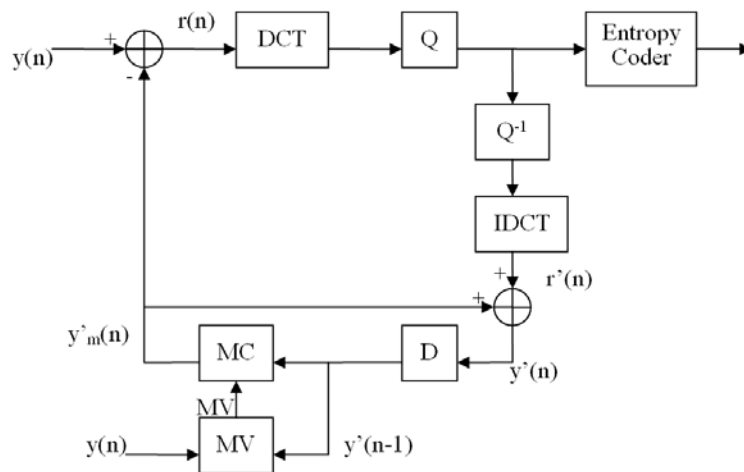


Figure 2-1 Block diagram of a typical video encoder [2]

Regarding video compression standards, there are two major organizations: ISO (International Organization for Standardization) and ITU-T (International Telecommunication Union Telecommunication Standardization Sector). In ISO, the major group working on the standardization of video coding is the MPEG (Moving Picture Experts Group) group. MPEG-I was their earliest standard, followed by MPEG-2, MPEG-4, and the latest MPEG-4 Part10 standards. The MPEG-4 Part 10 is also named as Advance Video Coding (AVC). In ITU-T, H.261 was the earliest video coding standard, followed by H.263, H.263+, and the latest H.264. Basically, the standardization of ITU-T focuses on applications in communications, while the standardization of MPEG focuses on multimedia applications.

However, in recent years, communication applications and multimedia applications get entangled together. Hence, the latest MPEG-4 Part 10 standard and the H.264 standard are actually a joint work from ISO and ITU-T. Hence, this new standard is also called H.264/AVC. In this paper, we are going to use this H.264/AVC standard as the framework of our study.

2.2 Introduction to H.264/AVC

Since 1988, the ITU-T Video Coding Experts Group (VCEG) prepared to set up a new video standard, called H.26L. Its purpose is to define an efficient coding scheme that could be twice more efficient than the state-of-the-art coding standards at that time. In December 2001, VCEG and MPEG organized a new team, called JVT (Joint Video Team), to work on the new video coding standard. As mentioned above, this new standard was called H.264/AVC [1].

There are two major layers in H.264/AVC, the Video Coding Layer (VCL) and the Network Abstraction Layer (NAL). The VCL layer contains techniques for video compression, while the NAL layer provides “network friendly” transmissions and error resilience capability.

Figure 2-2 show an illustration of the H.264/AVC coding structure.

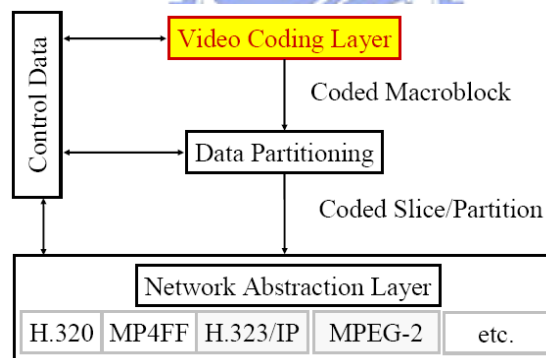


Figure 2-2 Structure of H.264/AVC [3]

The marked block in Figure 2-3 is the video coding layer. In H.264/AVC, only the decoder side is standardized. The specifications of the encoder side are left open. In this thesis, we are going to discuss a specific part of the encoder: the rate control part.

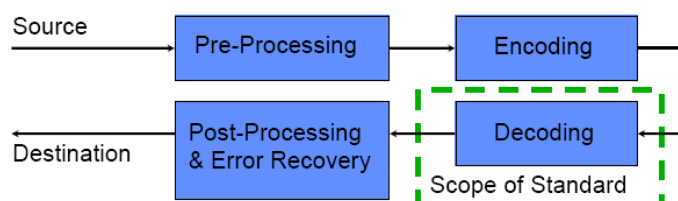


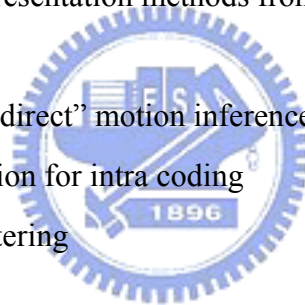
Figure 2-3 Scope of H.264/AVC standardization [3]

2.2.1 Highlights of H.264/AVC

In this section, we will describe some highlights of H.264/AVC. Some of these highlights are used to increase compression ratio, while some others are for error resilience and network-friendly transmissions. Based on their objectives, these highlights could be partitioned into three major categories [3]:

a. Improve Prediction Coding :

- Variable block-size motion compensation with small block sizes
- Quarter-sample-accurate MC
- MVs over picture boundaries
- Multiple reference picture MC
- Decoupling of referencing order from display order
- Decoupling of picture representation methods from picture referencing capability
- Weighted prediction
- Improved “skipped” and “direct” motion inference
- Directional spatial prediction for intra coding
- In-the-loop deblocking filtering



b. Increase compression rate :

- Small block-size transform : 4x4 DCT Transform
- Hierarchical block transform
- Short word-length transform : Only use 16-bits
- Exact-match inverse transform
- Arithmetic entropy coding
- Context-adaptive entropy coding

c. Help for error resilience and friendly network transmission :

- Parameter set structure :
Separate important data into different part to promise the safe in transmission
- NAL unit syntax structure :

A format of packet is used for network transmission

- Flexible slice size
- Flexible macroblock ordering (FMO)
- Arbitrary slice ordering (ASO)
- Redundant pictures
- Data Partitioning
- SP/SI synchronization/switching pictures

2.2.2 Video Coding Layer (VCL)

Figure 2-4 shows a typical structure of an H.264/AVC encoder. Intra prediction or inter motion compensation is the first step, followed by the DCT transform of the residual data, quantization of the transformed data, and entropy coding of the quantized data. Then, the coded data are sent to the NAL unit to be packed into packages for transmission. At the decoder site, reversed operations are applied for data reconstruction. In the following sections, we will describe some details of the H.264/AVC VCL layer.

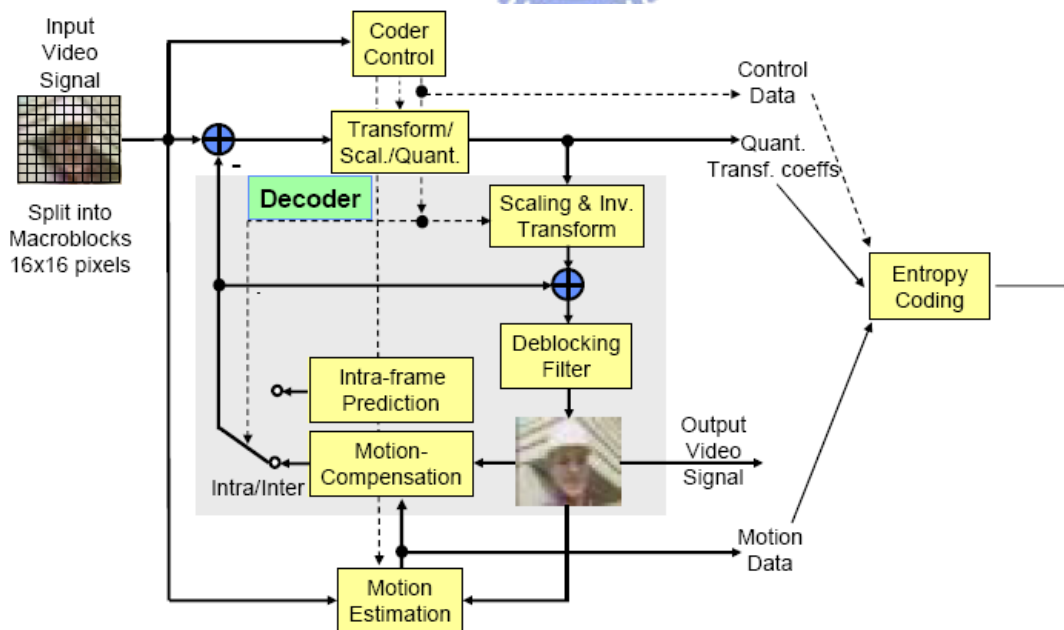


Figure 2-4 Basic coding structure for H.264/AVC for a macroblock [3]

a. Pictures, Frames, and Fields

A video sequence consists of several pictures. One Picture could be either a frame or a field. In H.264/AVC, the coding of a macroblock could be either frame coding or field coding.

b. YCbCr Color Space and 4:2:0 Sampling

The color space of H.264/AVC is YCbCr. A typical sampling pattern of YCbCr is 4:2:0. However, in the earliest profile (High Profile), which is also called Fidelity-Range Extension (FRExt), 4:4:4 sampling is supported.

c. Division of a Picture into Macroblocks

The luma macroblock size is 16x16 and the chroma macroblock size is 8x8. For the FRExt Profile, the macroblock size is decided based on the color sampling size.

d. Slices and Slice Groups

Slices are a sequence of macroblocks that are processed in the order of raster scan, if not using FMO (Flexible Macroblock Ordering). Two examples are illustrated in Figure 2-5. If using FMO, pictures are partitioned into slices. Each slice is a set of macroblocks defined by a macroblock-to-slice-group map. Two examples of FMO are illustrated in Figure 2-6.

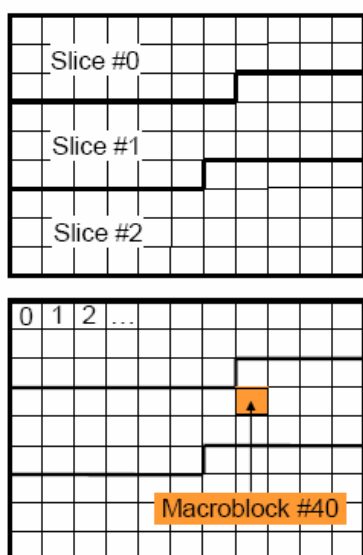


Figure 2-5 Possible subdivisions of a picture into slices [3]

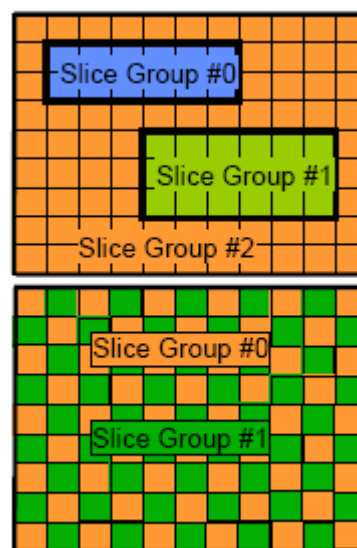


Figure 2-6 Possible subdivisions of a picture into slices with FMO.

No matter whether using FMO or not, slices are coded by one of the following ways.

- I slice: All macroblocks of a slice are coded by intra prediction.
- P slice: Slices are partitioned into several macroblocks. Some are coded based on motion-compensated prediction, while the others are coded based on intra prediction.
- B slice: similar to P slice, but some macroblocks may use bi-directional motion-compensation for prediction.

In addition to these three kinds of slices, there are two special types of slices: SI slice and SP slice. They are used in sequence switching. An illustration is shown in Figure 2-7.

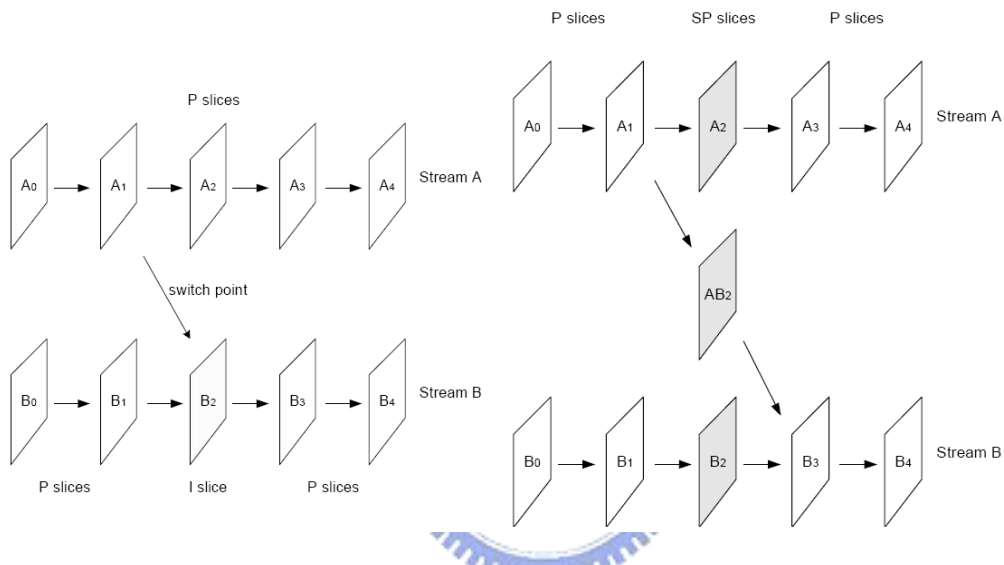


Figure 2-7 Switching streams using I-slice and SP-slices [4]

e. Intra-Frame Prediction

In Intra Prediction, there are two modes that are different in block size. One is the 4×4 prediction mode, and the other is the 16×16 prediction mode. Usually, we use the 4×4 mode in complex regions while use the 16×16 mode in smooth regions. The 4×4 prediction mode is further divided into nine different sub-modes to handle different edge directions. Some examples are shown in Figure 2-8. On the other hand, the 16×16 mode has four different sub-modes.

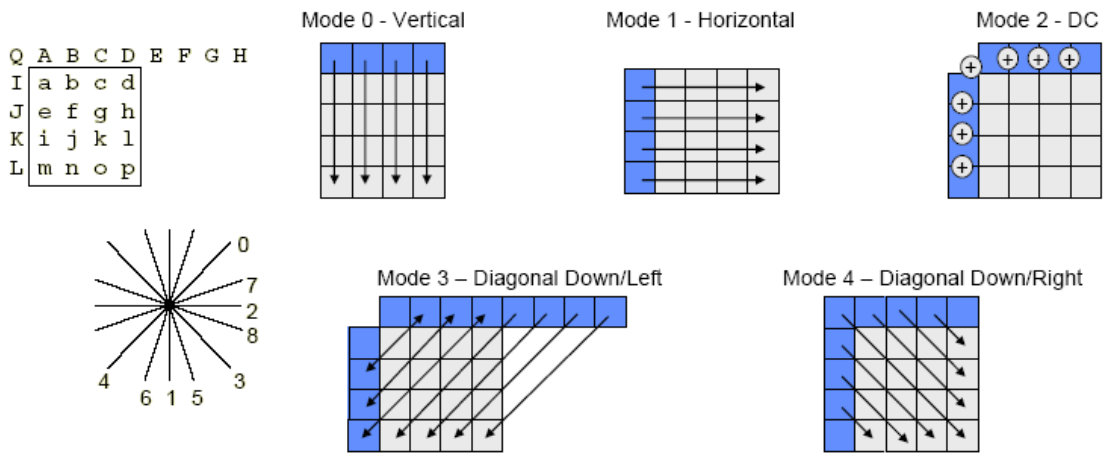


Figure 2-8 Five of nine Intra 4x4 prediction modes [3]

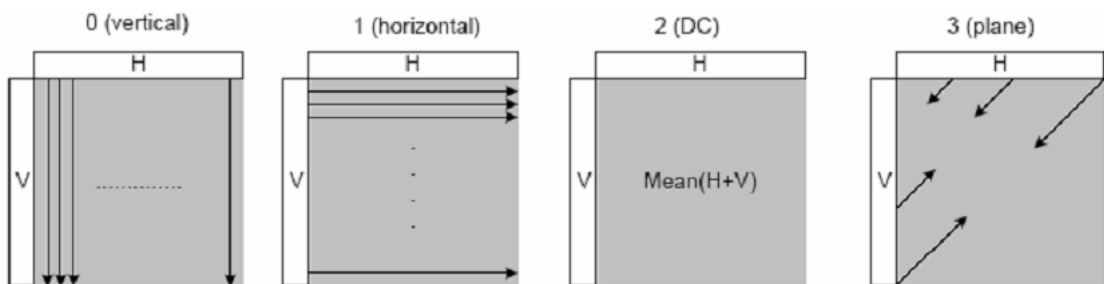


Figure 2-9 Four Intra 16x16 prediction modes [5]

f. Inter-Frame Prediction

In inter Prediction, H.264/AVC provides eight macroblock modes. They are illustrated in Figure 2-10. First, in the selection of Macroblock Types, we have four choices: 16x16, 16x8, 8x16 and 8x8. If we have chosen the 8x8 mode, each 8x8 Type has four extra modes to select: 8x8, 8x4, 4x8 and 4x4. Hence, according to the complexity of image contents, we may have different choices of modes. For example, we may have chosen the 16x16 mode for regions with a global motion, but use the 8x8 mode for regions that contain individual moving objects.

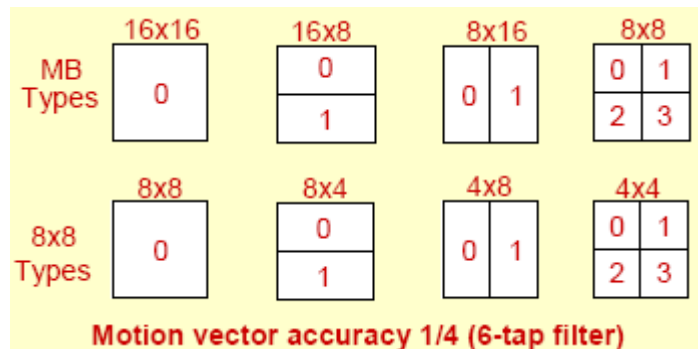


Figure 2-10 Decomposition of macroblock for motion compensation [3]

In H.264/AVC, the accuracy of motion compensation is in units of one quarter of the distance between luma samples. The prediction values at half-sample positions are obtained by applying a one-dimensional 6-tap FIR filter horizontally and vertically. Prediction values at quarter-sample positions are generated by averaging samples at integer- and half-sample positions. This is illustrated in Figure 2-11.

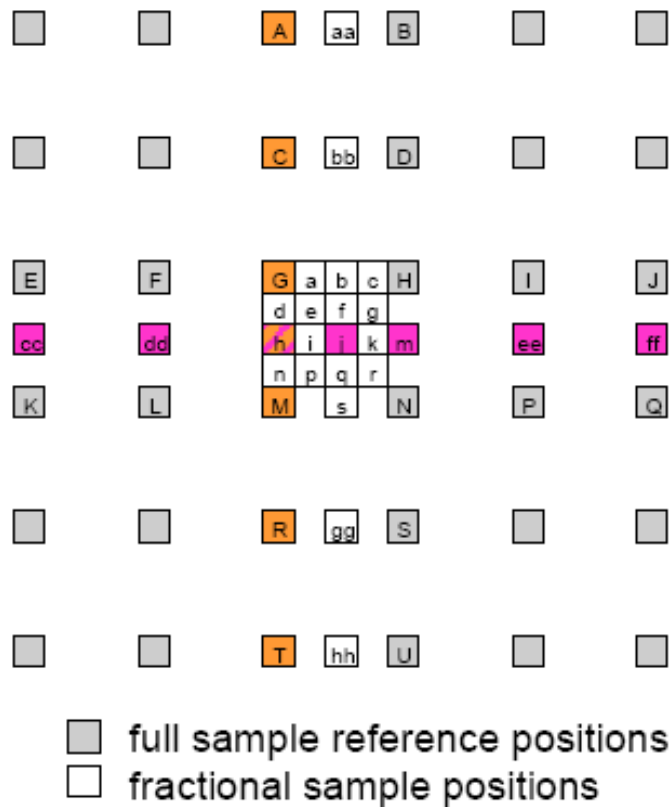


Figure 2-11 Filtering for fractional-sample accurate motion compensation [3]

g. Transform and Quantization

H.264/AVC uses 4x4 integral DCT, whose transform matrix is

$$H = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 2 & 1 & -1 & -2 \\ 1 & -1 & -1 & 1 \\ 1 & -2 & 2 & -1 \end{bmatrix}$$

This is an exact-match inverse transform. This transformation doesn't cause any miss match in the inverse transform.

Besides, after the DCT transform, the DC values of 16 luma blocks will be packed, as

shown in Figure 2-12, and then transformed by the 4x4 Hadamard transform. Similarly, the DC values of 4 chroma blocks will be packed and then transformed by the 2x2 Hadamard transform.

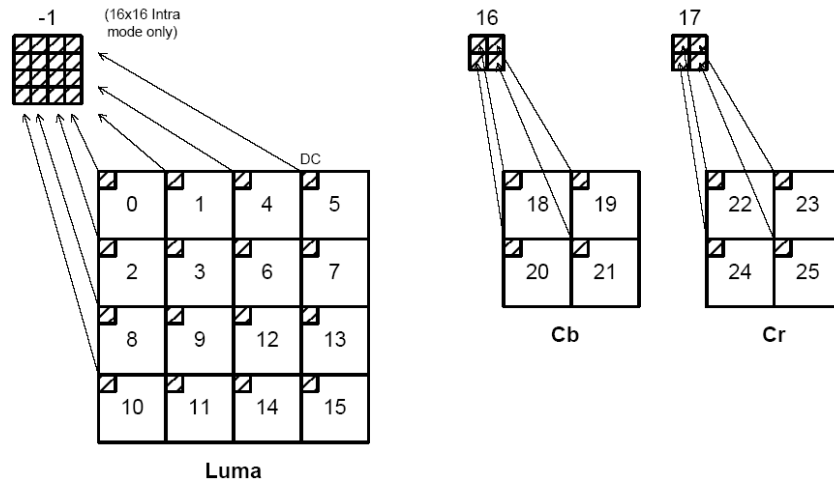


Figure 2-12 Package of the DCT DC values [5]

There are two major reasons for the use of the 4x4 DCT Transform. First, H.264/AVC has a great improvement in prediction coding. Hence, using a smaller size DCT transform may still obtain reasonable performance. Second, the computational complexity becomes lighter when a small-size DCT is used.

The QP values of H.264/AVC range from 0 to 51. An increase of 6 in the QP value will double the quantization step size.

h. Entropy Coding

There are two modes of entropy coding in H.264/AVC: Context-Adaptive Variable Length Coding (CAVLC) and Context-Adaptive Binary Arithmetic Coding (CABAC). CAVLC is a coding technique that is more efficient than VLC, but with higher complexity. On the other hand, CABAC uses Arithmetic Coding and is also more complex than VLC.

i. In-Loop Deblocking Filter

H.264/AVC is the first standard that uses in-loop deblocking filter. This means decoders would have a deblocking filter too. The use of the deblocking filter is to reduce blocking effect in the reference image. The use of the deblocking filter can improve the performance of

motion compensation and thus increase the coding efficiency.

2.2.3 H.264/AVC Profile

In previous sections, we briefly introduce the flow chart of H.264/AVC coding. Now, we will introduce the four profiles defined in H.264/AVC. These four profiles are

- **Baseline Profile:**

The main application of baseline profile is in low bit-rate transmission, e.g. cell phone transmission. This profile has low computational complexity and acceptable performance. In this paper, our algorithm is to be discussed on this Baseline Profile.

- **Main Profile:**

The difference between main profile and baseline profile is that the main profile contains interlaced coding. A main application of the main profile is for HDTV.

- **Extension Profile:**

This profile has error resilience tools and can be used in IP-TV or MOD. Figure 2-13 describes the different function units of the above three profiles.



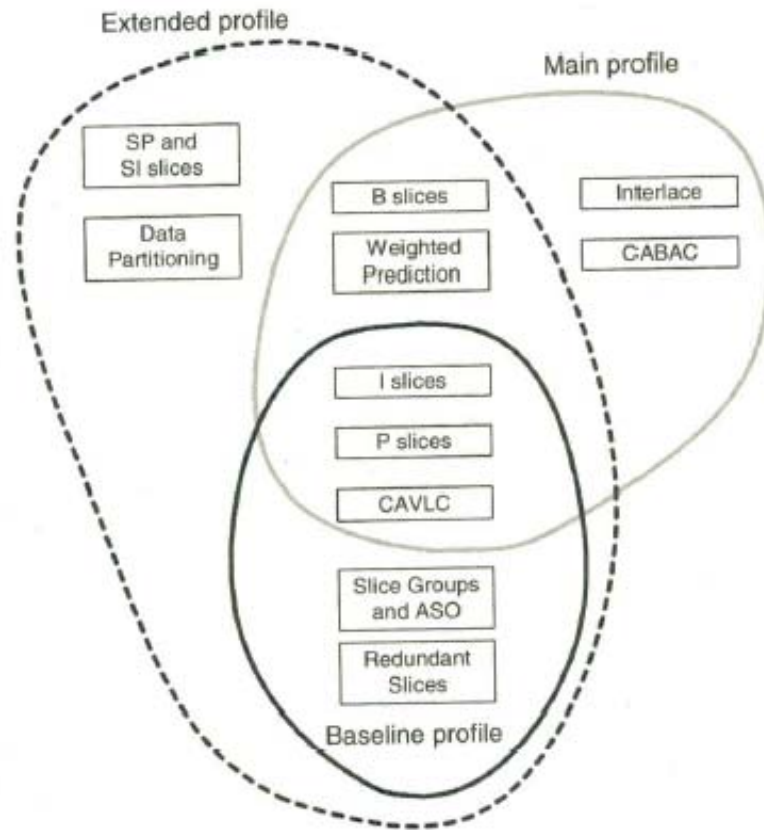


Figure 2-13 H.264/AVC Profiles [3]



- **High Profile:**

Because H.264/AVC has a rather poor compression performance in high resolution, the Fidelity-Range Extensions (FRExt) Profile is proposed in July 2004. The major purpose of this profile is to resolve the distortion of human vision in the high-resolution domain. This FRExt profile is also called the High Profile. In this profile, there are three major changes:

- 8x8 Intra Spatial Prediction
- 8x8 and 4x4 Transform Adaptive:

There are flexible selections. According to the situation, the encoder can select either the 4x4 or the 8x8 DCT transform.

- Various color spaces:

Besides the 4:2:0 chroma format, FRExt has additional 4:4:4 and 4:2:2 chroma formats. These formats perform better in high resolution.

Figure 2-14 illustrates the four different profiles of FRExt.

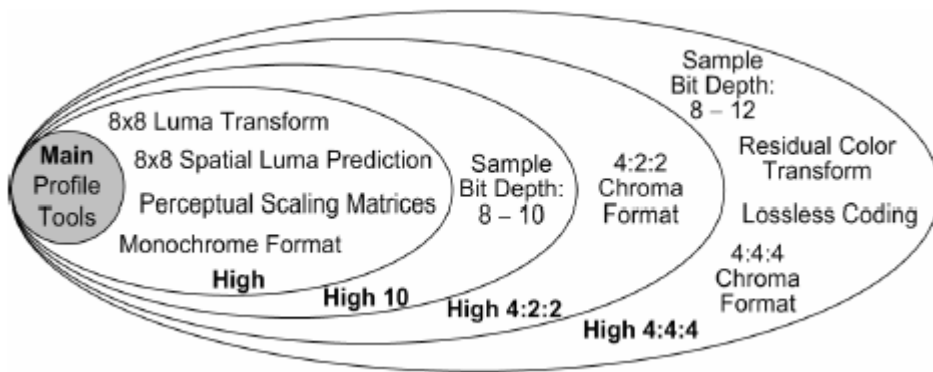


Figure 2-14 Illustration of the H.264/MPEG4-AVC FRExt profiles [6]



2.3 Introduction to Rate Control

The purpose of Rate Control is to control the encoded bits count. Figure 2-15 is a block diagram of video encoding. Generally speaking, we adjust the QP value to control the size of compressed data. The factors that may affect the determination of the QP values include channel bandwidth, buffer fullness, and video complexity.

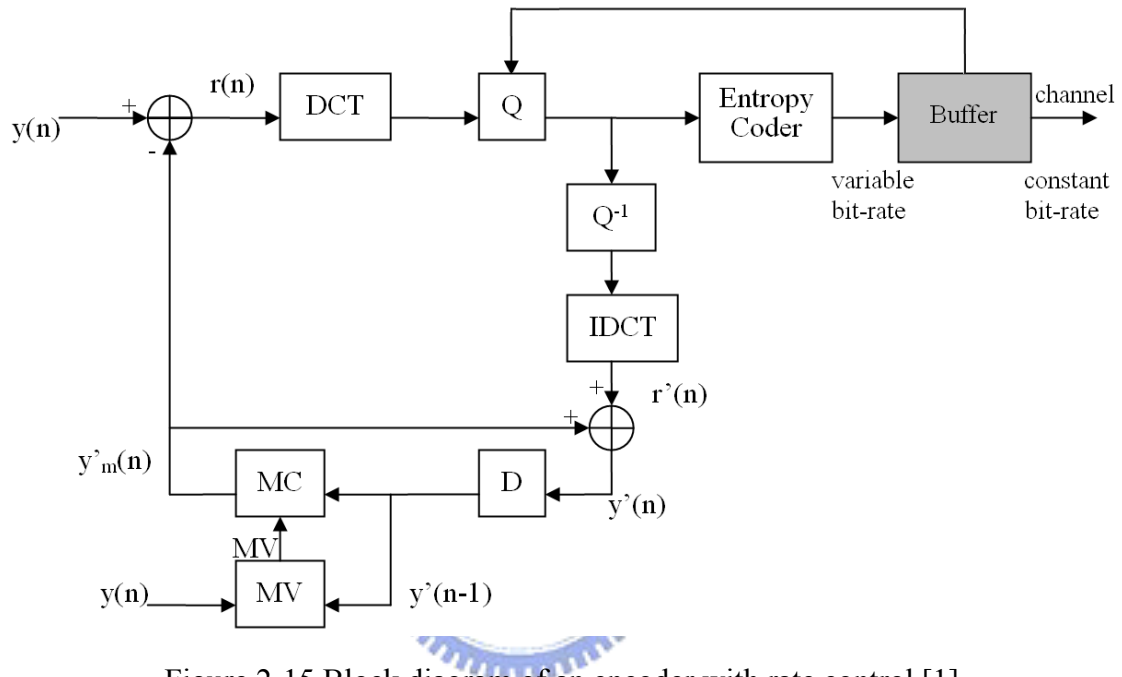


Figure 2-15 Block diagram of an encoder with rate control [1]

There exist several rate control algorithms for video coding standards, such as the TM5 [7] in MPEG-2, the TMN8 [8] in H.263, and the VM-18 [9] in MPEG-4. The scheme of rate control usually contains two parts. The first step is to determine the target bits based on buffer fullness or some other factors. The second step is to adjust the QP values to fit the target bits.

There also exist several rate-distortion models for the determination of the QP value. A commonly used model is the quadratic rate-distortion model proposed by Chiang and Zhang [1] which uses the image complexity and the target bits to decide the QP value. Besides, many other methods have been provided to better predict the QP values, hoping to provide better coding performance and video quality [12],[13],[14]. In the following sections, we will describe the rate control scheme adopted in H.264/AVC JM 8.4.

2.3.1 Rate Control of H.264/AVC

Rate-distortion optimization (RDO) is a rate control process suggested for H.264/AVC. Its purpose is to find the optimal tradeoff between rate and distortion. The motion vectors and block modes aren't decided according to the minimum distortion, but are decided by the RDO solution. The equations in RDO are as follows:

$$\begin{aligned} J(s, c, MODE | QP, \lambda_{MODE}) &= SSD(s, c, MODE | QP) + \lambda_{MODE} \cdot R(s, c, MODE | QP) \\ \lambda_{MODE, P} &= 0.85 \times 2^{QP/3} \\ J(m, \lambda_{MOTION}) &= SA(T)D(s, c(m)) + \lambda_{MOTION} \cdot R(m - p) \\ \lambda_{MOTION} &= \lambda_{MODE}^2 \end{aligned} \quad (2-1)$$

$SSD(s, c, MODE | QP)$ means the distortion between the original block s and the reconstructed block c when the value of QP is decided. $R(s, c, MODE | QP)$ means the bits after variable-length coding. $SA(T)D(s, c(m))$ is the distortion after deciding the motion vector m , and $R(m - p)$ is the bits when the motion vector is m and the predictive motion vector is p . According to the above equations, QP must be determined before executing the motion estimation. However, the determination of QP is to be performed at a later stage, after the MAD (mean of absolute difference) is calculated. Hence, this is actually a chicken and egg problem.

Figure 2-16 shows the block diagram of H.264/AVC rate controller. In the following sections, we will explain some major elements of this controller.

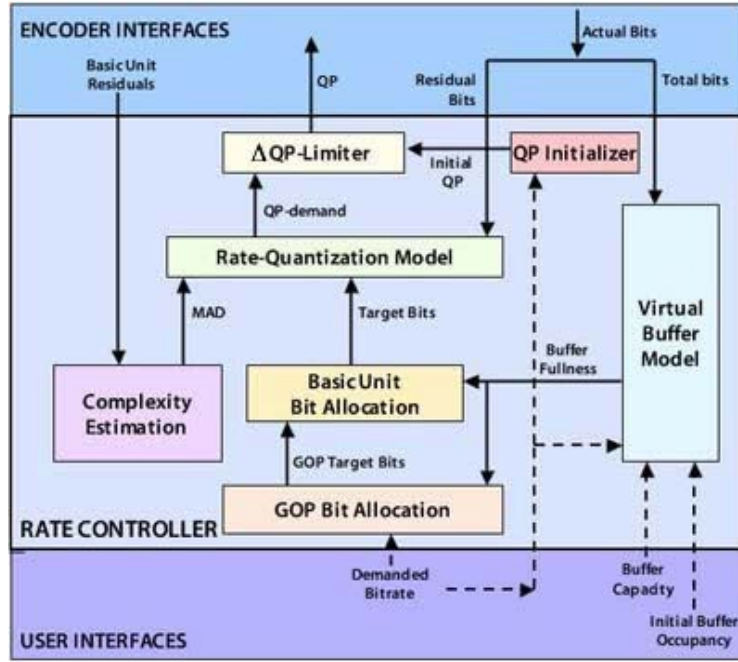


Figure 2-16 Block diagram of H.264/AVC Rate Controller [1]

- **Basic Unit**

One basic unit can be defined as one macroblock, one slice, or one frame. Assume there are N_{mbpic} macroblocks (MB) in one frame, and N_{mbuni} is the total number of MBs in one basic unit. If we denote the number of basic units as N_{unit} , then we have

$$N_{unit} = \frac{N_{mbpic}}{N_{mbunit}}. \quad (2-2)$$

- **Rate-Quantization Model**

Assume T denotes the number of target bits, which represents the predicted bits count before encoding. When the target bits had been determined, we need to decide the QP value for encoding. Using the quadratic rate-distortion model (R-D model), the QP value can be computed based on the following formula.

$$T_i(j) = c_1 \times \frac{\tilde{\sigma}_i(j)}{Q_{step,i}(j)} + c_2 \times \frac{\tilde{\sigma}_i(j)}{Q_{step,i}^2(j)} + m_{h,i}(j) \quad (2-3)$$

$T_i(j)$ means the target bits, the GOP index is I and the picture index is j .

M means the predicted header bits

σ means MAD (mean of absolute difference)

c_1 、 c_2 are coefficients

- **A linear Model for MAD Prediction**

The MAD value is used in the R-D model. However, as aforementioned, MAD value can be calculated only after the motion compensation process. Hence, we need to predict the MAD value beforehand. Here, the first-order linear model is used.

$$\tilde{\sigma}_i(j) = a_1 \times \sigma_i(j-1) + a_2, \quad (2-4)$$

where a_1 and a_2 are two coefficients for the prediction model. The initial values of a_1 and a_2 are set to 1 and 0. They are updated after the coding of each basic unit.

- **Δ QP Limiter**

The purpose of the Δ QP limiter is to avoid a dramatic change of the QP value between successive frames. A dramatic change in the QP value may cause unpleasant change of video quality. Normally, we constrain the difference between QP's to be less than 2.

- **Hypothetical Reference Decoder**

In order to place a practical limit on the size of decoder buffer, a lower bound and an upper bound for the target bits of each frame are determined by considering the hypothetical reference decoder (HRD) [12]. Compliant encoders must generate bitstreams that meet the requirements of the HRD. The lower bound and upper bounds for the n th frame are denoted as $L(n_{i,j})$ and $U(n_{i,j})$, respectively. Let $tr(n_{i,j})$ denote the removal time of the j th frame in the i th GOP. Also let $be(t)$ be the bit equivalent of the time t , with the conversion factor being the buffer arrival rate. The initial values of the upper bound and the lower bound are given as follows:

$$\begin{aligned} L(n_{i,1}) &= Tr(n_{i,0}) + \frac{u(n_{i,0})}{F_r} \\ U(n_{i,1}) &= (Tr(n_{i,0}) + be(t_r(n_{i,1}))) \times \varpi \end{aligned} \quad (2-5)$$

where $Tr(n_{i,0})$ is the remaining bits of the $(i-1)$ th GOP and $Tr(n_{1,0}) = 0$. The value of ϖ is

0.9. Then, $L(n_{i,j})$ and $U(n_{i,j})$ are computed iteratively as follows:

$$\begin{aligned} L(n_{i,j}) &= L(n_{i,j-1}) + \frac{u(n_{i,j-1})}{F_r} - b(n_{i,j-1}) \\ U(n_{i,j}) &= U(n_{i,j-1}) + \left(\frac{u(n_{i,j-1})}{F_r} - b(n_{i,j-1}) \right) \times \varpi \end{aligned} \quad (2-6)$$

2.3.2 H.264/AVC Rate Control Scheme

In this section, we will introduce the rate control scheme of H.264/AVC. The rate control of H.264/AVC has three parts. The first part is the GOP-level rate control, where the initial QP and the definition of buffer fullness of this GOP will be determined. The second part is the picture-level rate control, where the QP of each frame, such as a P frame or a B frame, will be determined. Finally, the basic-unit-level rate control will determine the QP value of each basic unit. In Table 2-1, we list the symbols used in this section.

2.3.2.1 GOP-Level Rate Control

In the GOP-level rate control, the expected available bits for GOP encoding are computed. When the j th picture had been encoded in the i th GOP, the formulae are

$$B_i(j) = \begin{cases} \frac{R_i(j) \times N_i - V_i(j)}{f} & j=1 \\ B_i(j-1) + \frac{R_i(j) - R_i(j-1)}{f} \times (N_i - j + 1) - b_i(j-1) & j=2,3,\dots,N_i \end{cases} \quad \text{and} \quad (2-7)$$

$$V_i(1) = \begin{cases} 0 & i=1 \\ V_{i-1}(N_{i-1}) & \text{other} \end{cases}$$

$$V_i(j) = V_i(j-1) + b_i(j-1) - \frac{R_i(j-1)}{f} \quad j=2,3,\dots,N_i \quad (2-8)$$

The initial QP value of each GOP is determined as follows. If this is the first GOP of the video sequence, then the initial QP is determined according to channel bandwidth. The relation between QP and bandwidth is formulated as following.

$$QP_i(1) = \begin{cases} 40 & bpp \leq l1 \\ 30 & l1 < bpp \leq l2 \\ 20 & l2 < bpp \leq l3 \\ 10 & bpp > l3 \end{cases} \quad \text{where } bpp = \frac{R_i(1)}{f \times N_{pixel}} \quad (2-9)$$

If the GOP is not the first GOP of the sequence, the initial QP value is determined according to the average QP value of the previous GOP. The formula is as follows.

$$QP_i(1) = \max \left\{ QP_{i-1}(1) - 2, \min \left\{ QP_{i-1}(1) + 2, \frac{SumQP(i-1)}{N_p(i-1)} - \min \left\{ 2, \frac{N_{i-1}}{15} \right\} \right\} \right\} \quad (2-10)$$

Table 2-1 Summary of Symbols

Parameter Name		Definition
Initialization	i	The index of GOP
	j	The index of Picture in each GOP
	f	Frame rate
	L	The number of successive non-stored pictures between two stored pictures.
	$R_i(j)$	Channel bit rate
	N_i	Total number of pictures in the i th GOP
	$N_p(i-1)$	Total number of stored pictures in the $(i-1)$ th GOP
	N_{pixel}	The number of pixels in a picture
Target Bit Estimation	$B_i(j)$	The bits for the rest pictures in this GOP
	$\bar{W}_{p,i}(j)$	The average complexity weight of stored pictures
	$\bar{W}_{b,i}(j)$	The average complexity weight of non-stored pictures
	$\tilde{T}_i(j)$	The delta target bits
	$\hat{T}_i(j)$	The hat target bits
	$T_i(j)$	Target bits
	$N_{p,r}$	The number of the remaining stored pictures
	$N_{b,r}$	The number of the remaining non-stored pictures
	$Z_i(j)$	The lower bound of HRD requirement
	$U_i(j)$	The upper bound of HRD requirement
	$t_{r,1}(1)$	The removal time of the first picture from the coded picture buffer
Buffer Control	$V_i(j)$	Buffer fullness
	$S_i(j)$	Target buffer level
Encoding, Post-Encoding	$b_i(j-1)$	Actual coded bits
	$QP_i(j)$	The j th picture's QP of i th GOP
	$Q_{step,i}(j)$	The j th picture's quantization step size
	$SumPQP(i-1)$	The sum of average picture QP for all stored picture in the $(i-1)$ th GOP
	$\tilde{\sigma}$	The predictive MAD
	σ	The actual MAD
	$m_{h,i}(j)$	The total number of header bits and motion vector bits

2.3.2.2 Picture-Level Rate Control

This level is divided into two stages, pre-encoding stage and post-encoding stage. They are explained as follows.

A. Pre-encoding stage

This stage contains two different cases: stored pictures (P frames) and non-stored pictures (B frames). Here, we describe them individually.

(a) Stored picture

Step 1: Determine the target bits of this picture

Step 1.1: determine the target buffer level of each picture

Assume $S_i(j)$ is the target buffer level. Then,

$$S_i(2) = V_i(2)$$

$$S_i(j+1) = S_i(j) - \frac{S_i(2)}{N_p(i) - 1} + \frac{\bar{W}_{p,i}(j) \times (L+1) \times R_i(j)}{f \times (\bar{W}_{p,i}(j) + \bar{W}_{b,i}(j) \times L)} - \frac{R_i(j)}{f} \quad (2-11)$$

L means the number of the non-stored pictures between two stored pictures. The complexity weight of stored picture and non-stored picture are expressed as followings.

$$\bar{W}_{p,i}(j) = \frac{W_{p,i}(j)}{8} + \frac{7 \times \bar{W}_{p,i}(j-1)}{8}$$

$$\bar{W}_{b,i}(j) = \frac{W_{b,i}(j)}{8} + \frac{7 \times \bar{W}_{b,i}(j-1)}{8}$$

$$W_{p,i}(j) = b_i(j) \times QP_{p,i}(j)$$

$$W_{b,i}(j) = \frac{b_i(j) \times QP_{b,i}(j)}{1.3636}$$

$\bar{W}_{p,i}(j)$ the average complexity weight of stored pictures

$\bar{W}_{b,i}(j)$ the average complexity weight of non - stored pictures (2-12)

Step 1.2: compute the target bits of each picture

The target bits are predicted in two aspects. One is according to buffer fullness and channel bit rate; the other is determined by the remaining bits of this GOP.

$$\tilde{T}_i(j) = \frac{R_i(j)}{f} + \gamma \times (S_i(j) - V_i(j)) \quad \gamma = 0.5 \quad (2-13)$$

$$\hat{T}_i(j) = \frac{W_{p,i}(j-1) \times B_i(j)}{W_{p,i}(j-1) \times N_{p,r} + W_{b,i}(j-1) \times N_{b,r}} \quad (2-14)$$

The real target bits are computed by combining $\tilde{T}_i(j)$ and $\hat{T}_i(j)$.

$$T_i(j) = \beta \times \hat{T}_i(j) + (1 - \beta) \times \tilde{T}_i(j) \quad \beta = 0.5 \quad (2-15)$$

Finally, the target bits will satisfy the limit of HRD.

$$T_i(j) = \max\{Z_i(j), T_i(j)\}$$

$$T_i(j) = \min\{U_i(j), T_i(j)\}$$

$$Z_i(j) = \begin{cases} B_{i-1}(N_{i-1}) + \frac{R_i(j)}{f} & j=1 \\ Z_i(j-1) + \frac{R_i(j)}{f} - b_i(j) & \text{other} \end{cases}$$

$$U_i(j) = \begin{cases} (B_{i-1}(N_{i-1}) + t_{r,1}(1)) \times \varpi & j=1 \\ U_i(j-1) + (\frac{R_i(j)}{f} - b_i(j)) \times \varpi & \text{other} \end{cases} \quad (2-16)$$

Here, ϖ is a constant with a typical value of 0.9.

Step 2: Compute the QP value and perform the RDO process

First the MAD value is estimated based on a linear model.

$$\tilde{\sigma}_i(j) = a_1 \times \sigma_i(j-1-L) + a_2 \quad (2-17)$$

After this, t

$$T_i(j) = c_1 \times \frac{\tilde{\sigma}_i(j)}{Q_{step,i}(j)} + c_2 \times \frac{\tilde{\sigma}_i(j)}{Q_{step,i}^2(j)} + m_{h,i}(j)$$

Finally, the Δ QP Limiter will limit the difference between the current QP and the previous QP to be less than two.

(b) Non-stored picture

The mainly idea is to use the interpolation of the previous and the subsequent stored pictures.

B. Post-encoding

In this stage, the coefficients such as a_1 and a_2 in the MAD linear model will be updated.

They will be updated by the linear regression method. The coefficients c_1 and c_2 of the R-D model are also updated. Finally, this stage will add the actual encoded bits into the buffer and make sure whether the buffer overflows.

2.3.2.3 Basic-Unit-Level Rate Control

At this stage, we need to determine the QP value for each basic unit.

Step 1: Predict the MAD of each basic unit according to the linear mode.

Step 2: Compute the target bits of each basic unit.

Step 3: Determine the QP value of each basic unit. There are three different cases.

Case 1: If this is the first basic unit in this picture, the QP is the average value of the previous picture QP.

Case 2: If the remaining bits of this picture become negative, the QP value is determined by adding a small value onto the QP value of the previous basic unit.

Case 3: Apart from the above two cases, the QP value of the remaining basic units is determined according to the quadratic R-D model.

Step 4: Execute RDO in each macroblock and encode.

Step 5: The coefficients in the linear prediction model and in the quadratic R-D model are updated.

Chapter 3

Modified Rate-Distortion Model for H.264/AVC

In this chapter, we discuss the relation between the rate-distortion (quantization) model and rate control. First, we will describe some existing R-D models. Then, we formulate a new R-D model according to the relation between bits and quantization parameters. Since the header bits are also important in the R-D model, we also find the relation between the header bits and the macroblock mode. Finally, we build a modified R-D model to provide an improved performance in rate control.



3.1 Previous R-D Models

There are several kinds of R-D models for different video standards. Chiang and Zhang proposed a quadratic rate-distortion model that is used in MPEG-4 [1]. In H.264/AVC, the Joint Model (JM) also uses this model [11]. However, since MPEG-4 and H.264/AVC do not use the same quantization, transform coding and entropy coding, this quadratic R-D model may not be suitable for H.264/AVC.

Tsai and Leou provided several choices for the estimation of QP [13]. These choices could be based on a quadratic formula, a logarithmic formula, or an exponential formula. These formulas are expressed as follows.

$$\begin{aligned} B_i &= X_1 \times Q_i^{-2} + X_2 \times \delta_i \times Q_i^{-1} + X_3 \\ B_i &= Y_1 \times \delta_i \times (\ln Q_i)^{-1} + Y_2 \\ B_i &= Z_1 \times \delta_i \times 2^{-Q_i} + Z_2 \end{aligned} \tag{3-1}$$

where X_1 , X_2 , X_3 , Y_1 , Y_2 , Z_1 , and Z_2 are the model parameters, δ_i is the SAD of the i -th frame,

and B and Q are the number of bits and the QP value. The QP value is determined from one of these three equations that can have the minimum error.

Satoshi Muyaaji and Yasuhiro Takishima observed the relation between SAD (Sum of Absolute Difference) and the number of generated bits [14]. Figure 3-1 shows that these two factors are highly correlated and the relation can be well approximated by the following quadratic polynomial:

$$bits = a \times S^2 + b \times S + c, \quad (3-2)$$

where “bits” is the number of generated bits and S is the average SAD in the macroblock. These coefficients a, b and c are calculated by the least squares method. Finally, based on (3-2), Muyaaji and Takishima deduced a formula for the R-D model.

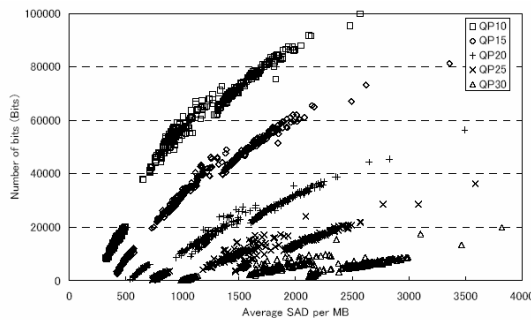


Figure 3-1 Relation between SAD and the number of generated bits [14].

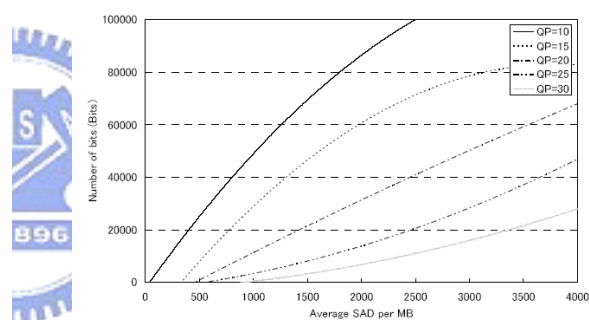


Figure 3-2 Approximated relation between SAD and the number of generated bits [14].

On the other hand, Siwei Ma and Wen Gao draw a conclusion that the relation between R and $1/Q_{step}$ in H.264/AVC can be described by a linear model [15], as shown in Figure 3-3 and Figure 3-4. Hence, they model the relation between rate and quantization stepsize (R- Q_{step} model) as

$$R_i^t = \frac{K^t SAD_i}{Q_{step,i}} + C^t \quad t = I, P, B \quad (3-3)$$

where R is the estimated number of coded bits of a macroblock; SAD_i is the sum of absolute difference (SAD) of a motion-compensated macroblock; C is the bits used to code the header information of a macroblock; and K is a coefficient of the model.

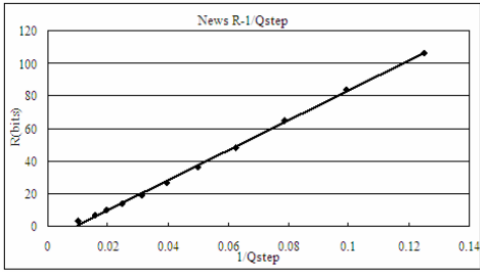


Figure 3-3 The relation between the coded coefficient bits and $1/Q_{step}$ for “news” [15]

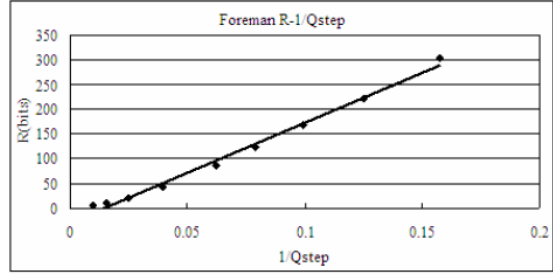


Figure 3-4 The relation between the coded coefficient bits and $1/Q_{step}$ for “foreman” [15]

In the following section, we will formulate a new R-D model that can better fit the relation between the number of coded bits and the quantization parameter.



3.2 Our Rate-Distortion Model

First, we are interested in modeling the relation between the number of coded bits and the quantization parameter when the residual MAD of the macroblock is fixed. Figure 3-5 shows the plots of the number of generated bits with respect to QP, Q_{step} , $1/QP$, and $1/Q_{step}$, respectively. Here, the relation between QP and Q_{step} is expressed as follows. It can be easily deduced that as the QP value increases by six, the step size is doubled.

$$Q_{step} = 2^{(QP-4)/6} \quad (3-4)$$

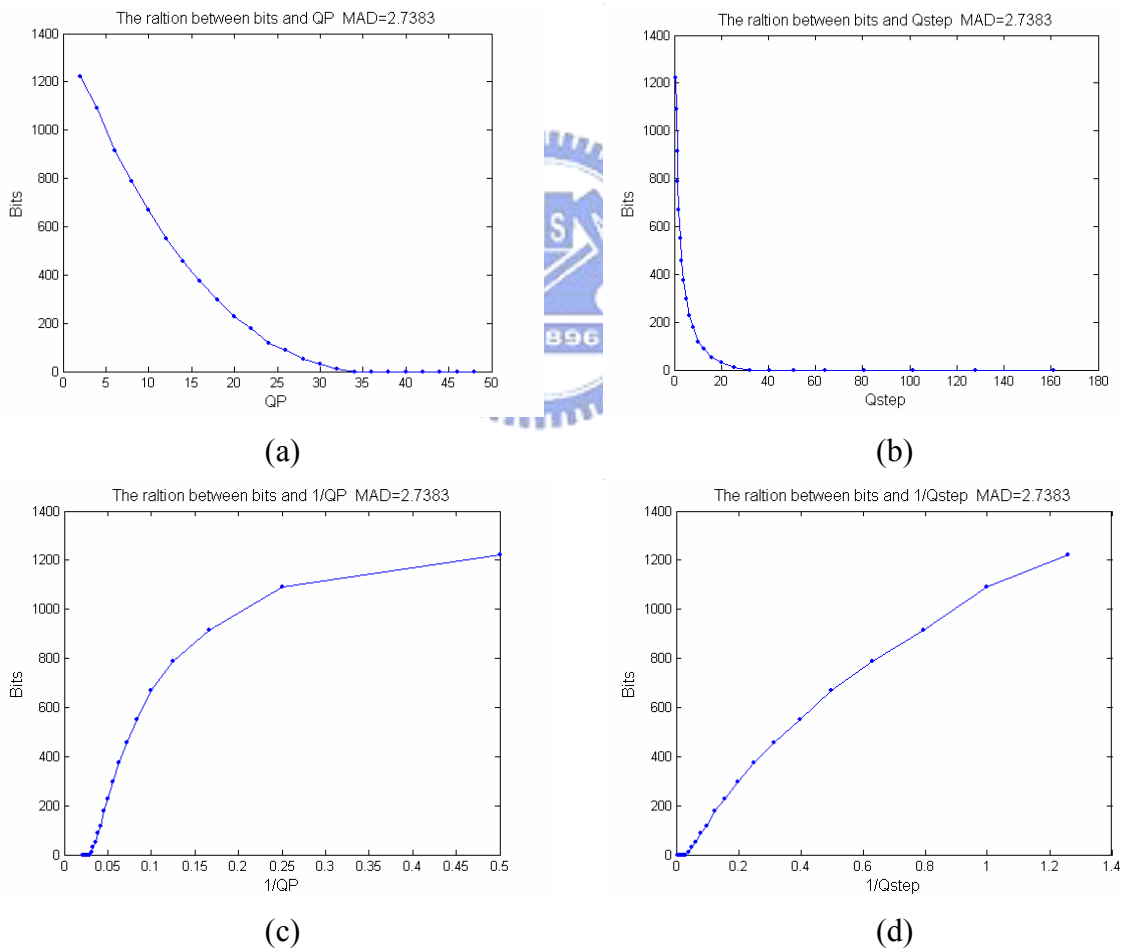
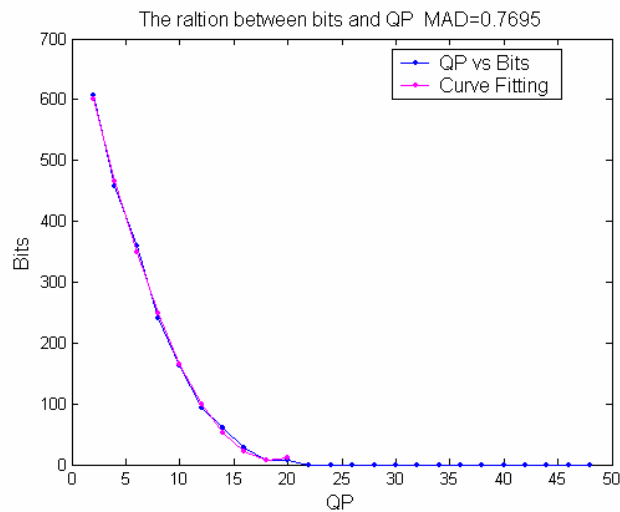


Figure 3-5 (a) Relation between bits and QP, (b) Relation between bits and Q_{step}
(c) Relation between bits and $1/QP$, (d) Relation between bits and $1/Q_{step}$

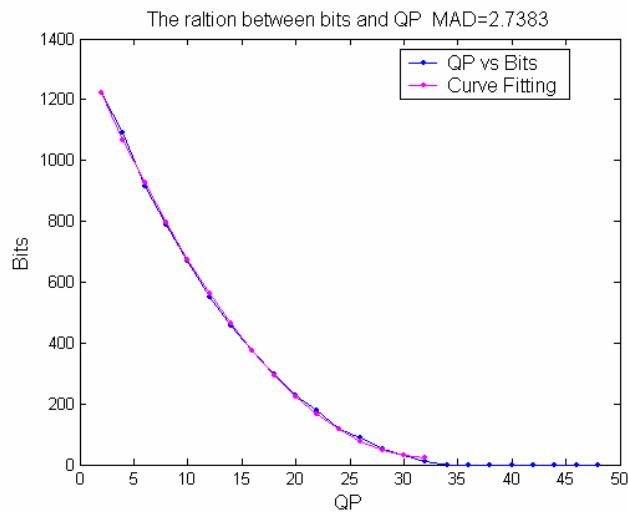
It can be seen in Figure 3-5 that the relation between the number of generated bits and $1/Q_{\text{step}}$ doesn't actually follow the linear model. Hence, we adopt the second-order polynomial instead. That is,

$$\text{Bits} = a \times QP^2 + b \times QP + c \quad (3-5)$$

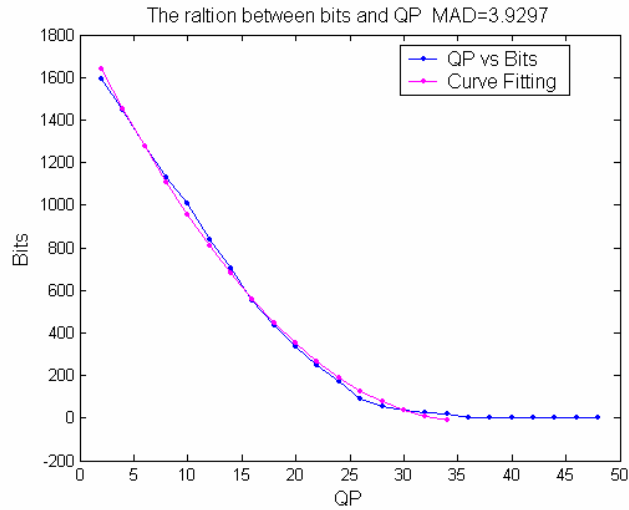
Figure 3-6 describes the curve fitting results based on the second order polynomial. It can be seen that the second order polynomial fits the relationship pretty well.



(a)



(b)



(c)

Figure 3-6 The relation between the number of coded bits and QP. The pink curves show the fitting of a second-order polynomial. (a)MAD=0.7695 (b)MAD=2.7383 (c)MAD=3.9297

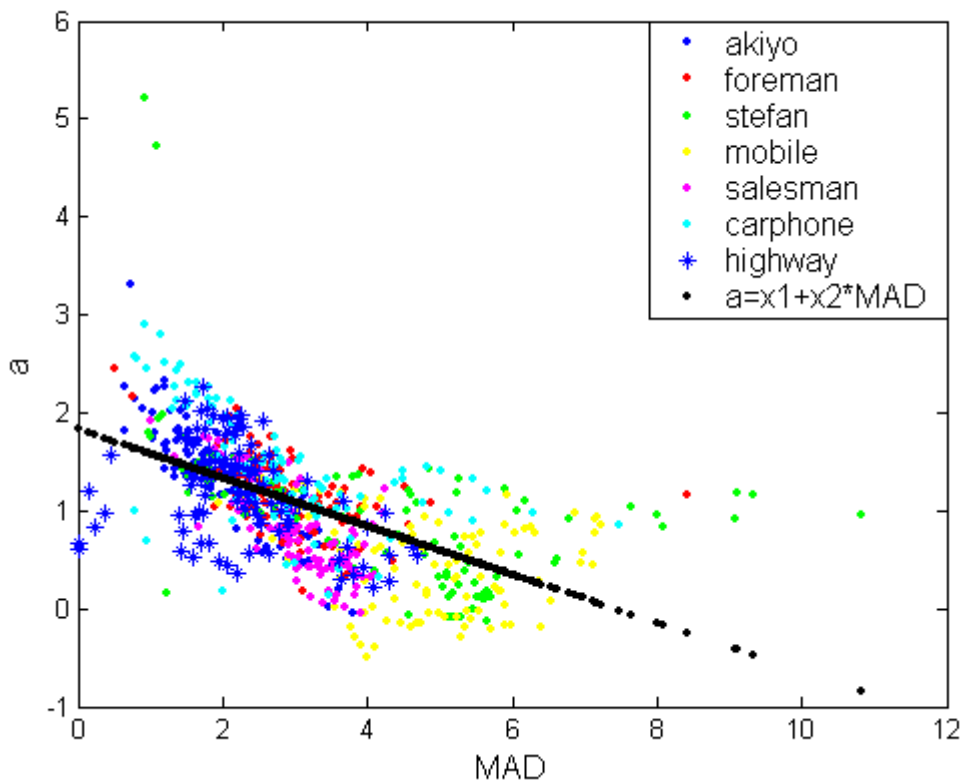


Figure 3-7 The relation between the coefficient “a” and MAD in different test sequences.

After this, we will find the relation between MAD and the coefficients in (3-5). Figure 3-7 plots the relation between the coefficient “a” and MAD. Based on this experiment, we model the relation to be linear. The linear model is expressed as

$$a = \max(0, x_1 + x_2 \times MAD). \quad (3-6)$$

The lower bound 0 is to avoid a negative “a”, which does not match our assumed model.

Similarly, we can find the relationship between the coefficient “b” and MAD. Figure 3-8 shows that b is approximately equal to the constant -90 for various values of MAD. Even though the variance of b is actually not very small, we still treat “b” as a constant in our model to simplify the problem.

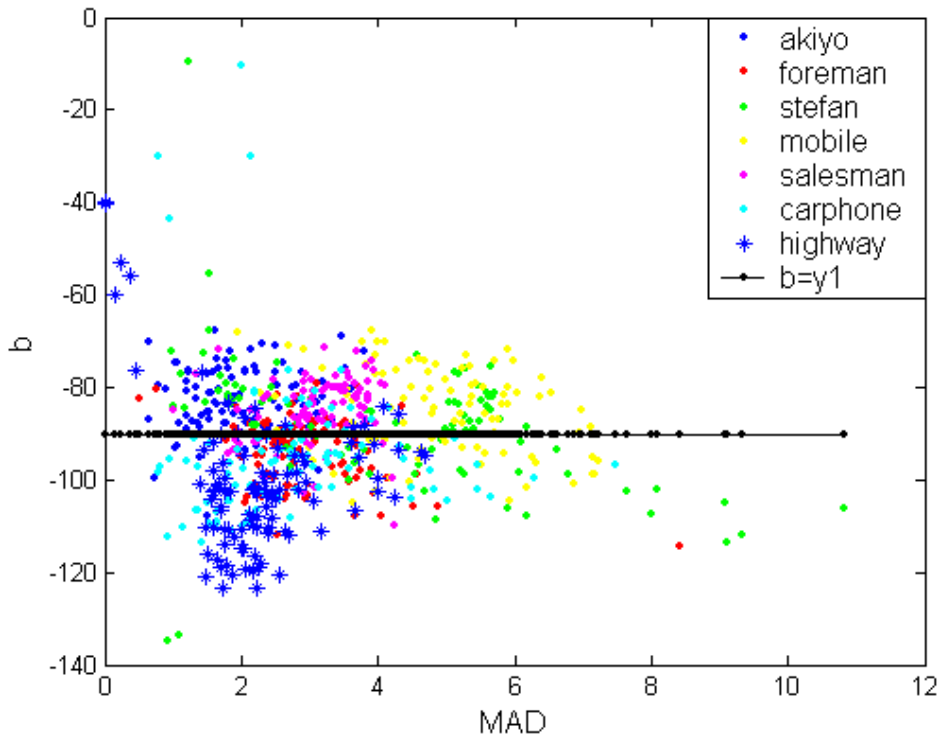


Figure 3-8 The relation between the coefficient “b” and MAD in different test sequences.

Regarding the coefficient “c”, we can see in Figure 3-9 that there is an apparent relationship between “c” and MAD. We have tried several curve models to fit this relationship and finally find the simple model expressed in Equation (3-7). The fitting result is shown in Figure 3-9.

$$c = z_1 \times MAD + z_2 \times \sqrt{MAD} \quad (3-7)$$

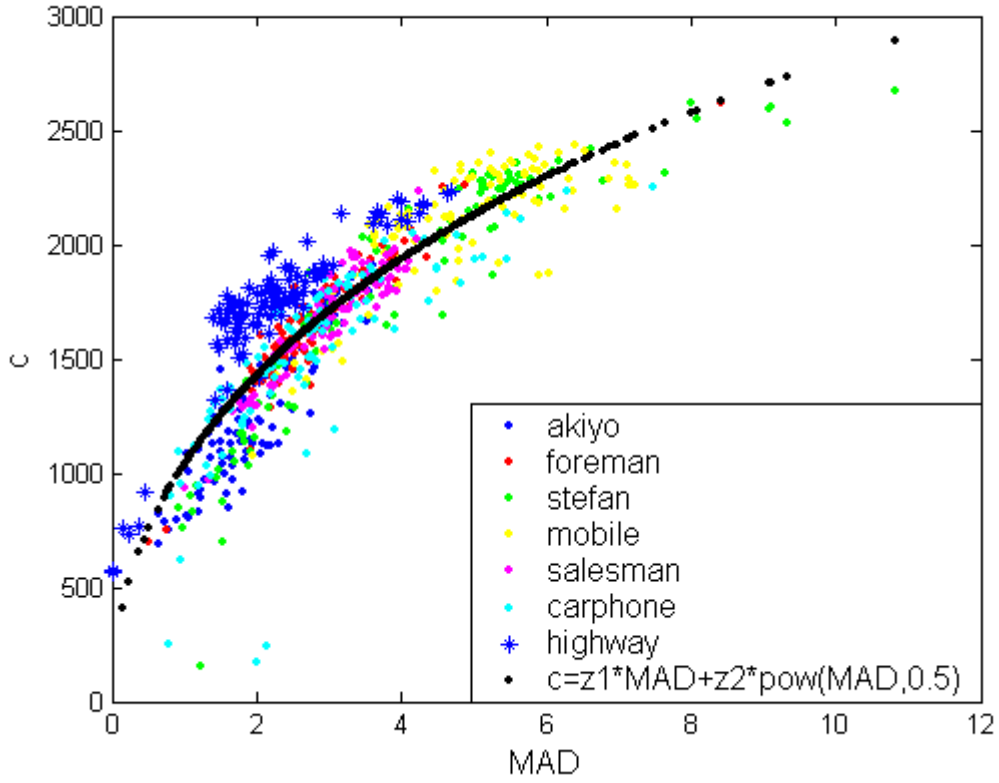


Figure 3-9 The relation between the coefficient “c” and MAD in different test sequences.

Based on Equations (3-6) and (3-7), we can rewrite the equation (3-5) as

$$\begin{aligned}
 Bits &= a \times QP^2 + b \times QP + c \\
 &= (x_1 + x_2 \times MAD) \times QP^2 + y_1 \times QP + z_1 \times MAD + z_2 \times \sqrt{MAD} \\
 &= x_2 \times MAD \times QP^2 + z_1 \times MAD + x_1 \times QP^2 + z_2 \times \sqrt{MAD} + y_1 \times QP \\
 &= k_1 \times A^2 B^2 + k_2 \times A^2 + k_3 \times B^2 + k_4 \times A + k_5 \times B
 \end{aligned}$$

where $A = \sqrt{MAD}$ and $B = QP$

$$Bits = k_1 \times MAD \times QP^2 + k_2 \times MAD + k_3 \times QP^2 + k_4 \times \sqrt{MAD} + k_5 \times QP \quad (3-8)$$

The equation (3-8) needs one extra constraint. This is because this model is used only for non-zero coded bits. Observing Figure 3-6, we can find that when the QP value is large enough, the number of coded bits become 0. For example, the number of coded bits become 0 when QP is 24 in Figure 3-6 (a). We name this turning point as the “zero point”. In Figure

3-10, we show the relationship between MAD and the “zero point”..

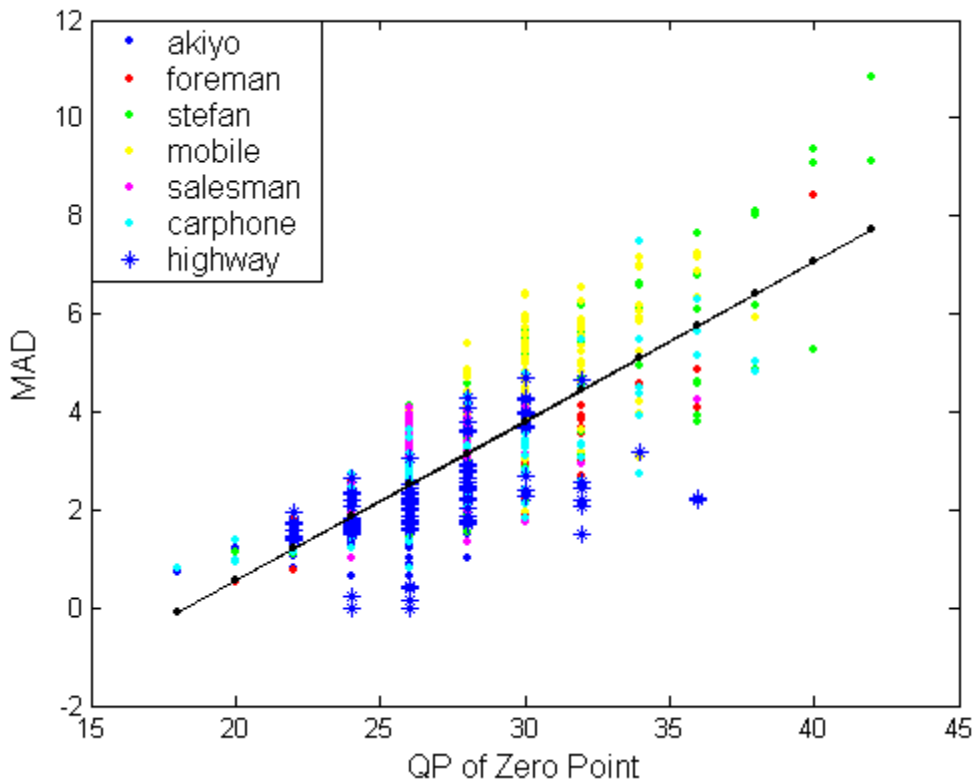


Figure 3-10 The relation between MAD and the “zero point”

Based on Figure 3-10, we describe the relation between MAD and the “zero point” as a linear model and express the relationship as

$$MAD = h_1 + h_2 \times QP \quad (3-9)$$

Then, we add on the constraint that MAD has to be larger than some value. The equation (3-8) is then rewritten as

$$Bits = k_1 \times MAD \times QP'^2 + k_2 \times MAD + k_3 \times QP'^2 + k_4 \times \sqrt{MAD} + k_5 \times QP'$$

$$QP' = \begin{cases} QP' & QP' \leq \frac{MAD - h_1}{h_2} \times \sigma \\ QP' - 1 & QP' > \frac{MAD - h_1}{h_2} \times \sigma \end{cases} \quad (3-10)$$

where σ is a parameter that must be larger than one. In this new model, we'll have to calculate

the model coefficients k_1 , k_2 , k_3 , k_4 and k_5 . Here, we may use a linear regression technique to find these coefficients. Based on this model, we then formulate our R-D model for H.264/AVC at the macroblock level. In the following experiments, we will demonstrate the performance of the proposed R-D model. In Table 3-1, we first list some major factors in our experiments.

Table 3-1 Table of experiment factors

Factor	Selections
Sequence	Bus, Flower, Highway, Stefan
Bit Rate(Kilo-bits per sec)	64, 128
Size of GOP	Number of total frames, 30
Frame Size	QCIF
Frame Rate (frame/sec)	30
Buffer Size	0.5 Bit rate
Basic Unit	One Macroblock
RDO	Off and On

Here, we choose four different sequences. The available bit rate corresponds to the channel bandwidth. Because the tested frame size is QCIF, the PSNR of the QCIF sequences will be high enough when the bandwidth is wider than 128k bits/sec. Hence, we test only at two different bit rates. Moreover, the size of GOP corresponds to the period of I-frames and the basic unit is chosen to be one macroblock.

Case A: The size of GOP is the number of total frames

Figure 3-11, Figure 3-12, Figure 3-13 and Figure 3-14 shows the situation of buffer fullness during the encoding of various sequences. The x-axis indicates the number of frames and the y-axis indicates the situation of buffer fullness. In theory, the buffer fullness must be close to the target buffer level. However, in low bit rate, there could be a large difference between the buffer fullness and the target buffer level. It can be seen that our model can reduce this difference. That is, in Figure 3-11, Figure 3-12, Figure 3-13 and Figure 3-14, we can see that our buffer fullness curve is closer to the expected curve.

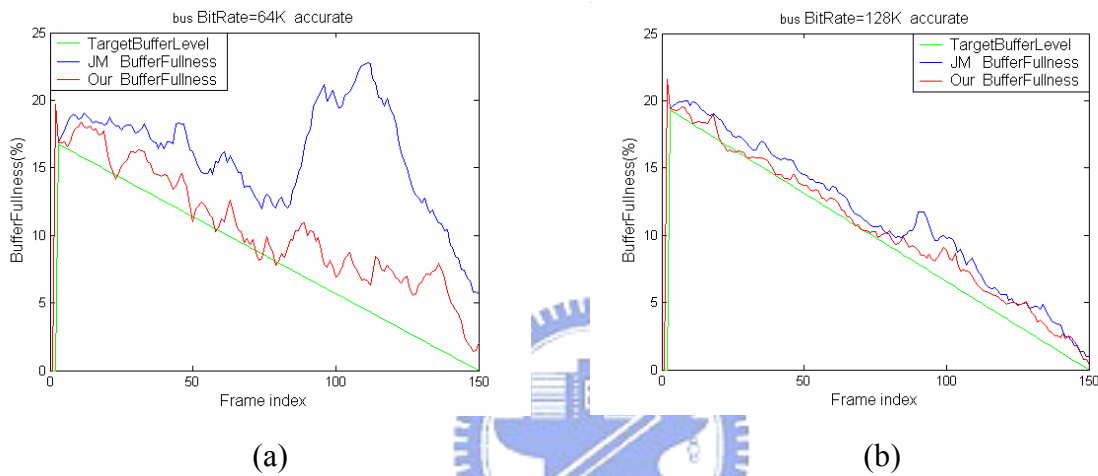


Figure 3-11 The buffer fullness and target buffer level for the “bus” sequence when the bit rates are 64K and 128K.

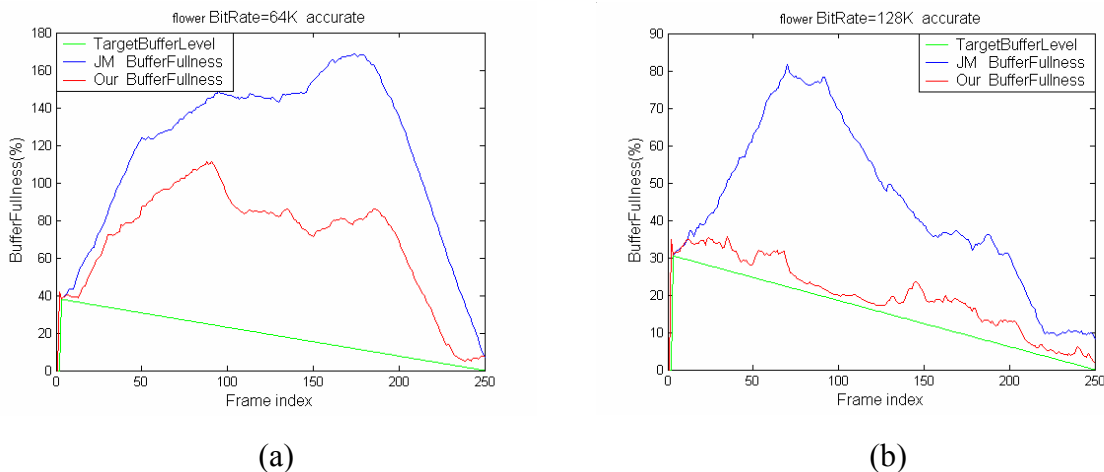
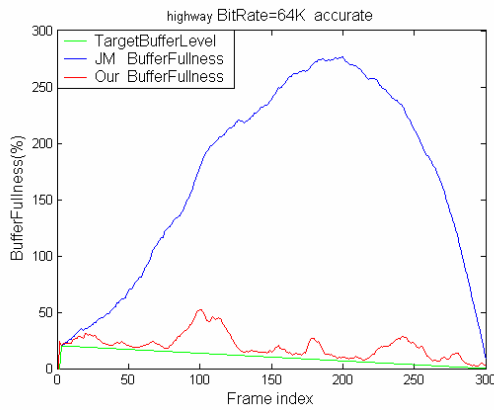
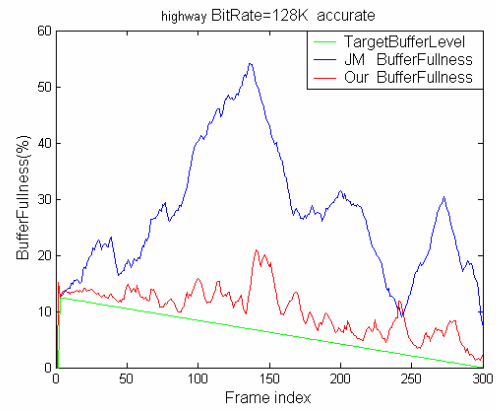


Figure 3-12 The buffer fullness and target buffer level for the “flower” sequence when the bit rates are 64K and 128K.

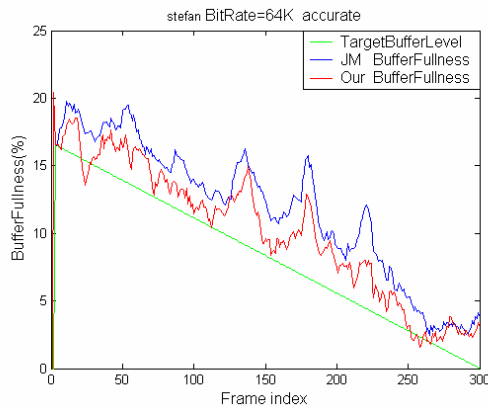


(a)

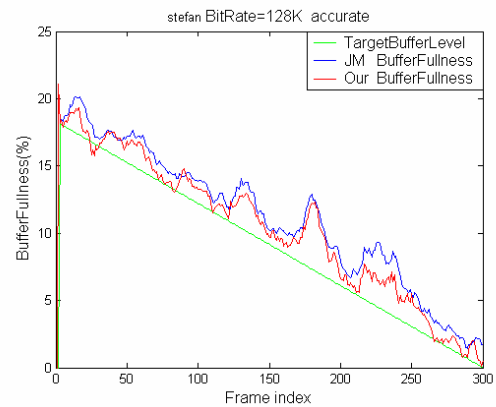


(b)

Figure 3-13 The buffer fullness and target buffer level for the “highway” sequence when the bit rates are 64K and 128K.



(a)



(b)

Figure 3-14 The buffer fullness and target buffer level for the “Stefan” sequence when the bit rates are 64K and 128K.

An inaccurate R-D model in rate control will cause some serious problems. In Figure 3-12 and Figure 3-13, the buffer fullness is over 100%. It means the size of buffer is not large enough. In real transmission, the “frame skip” operation will happen to avoid buffer overflow and this will cause the decrease of video quality. Besides this, the visual quality of each frame may be decreased. When there are no remaining target bits in the current frame, the rest basic unit cannot use the R-D model to determine its QP value. Instead, the QP value will be determined by adding a positive delta value on the previous QP value. Hence, the rest basic units will have a decreased visual quality. Figure 3-15 describes the number of basic units that

have no remaining target bits. In theory, a larger number of this kind of basic units will cause a more serious overflow of the encoded bits. In Figure 3-11, the rising of the buffer fullness curve is because the coded bits are larger than the expected target bits.

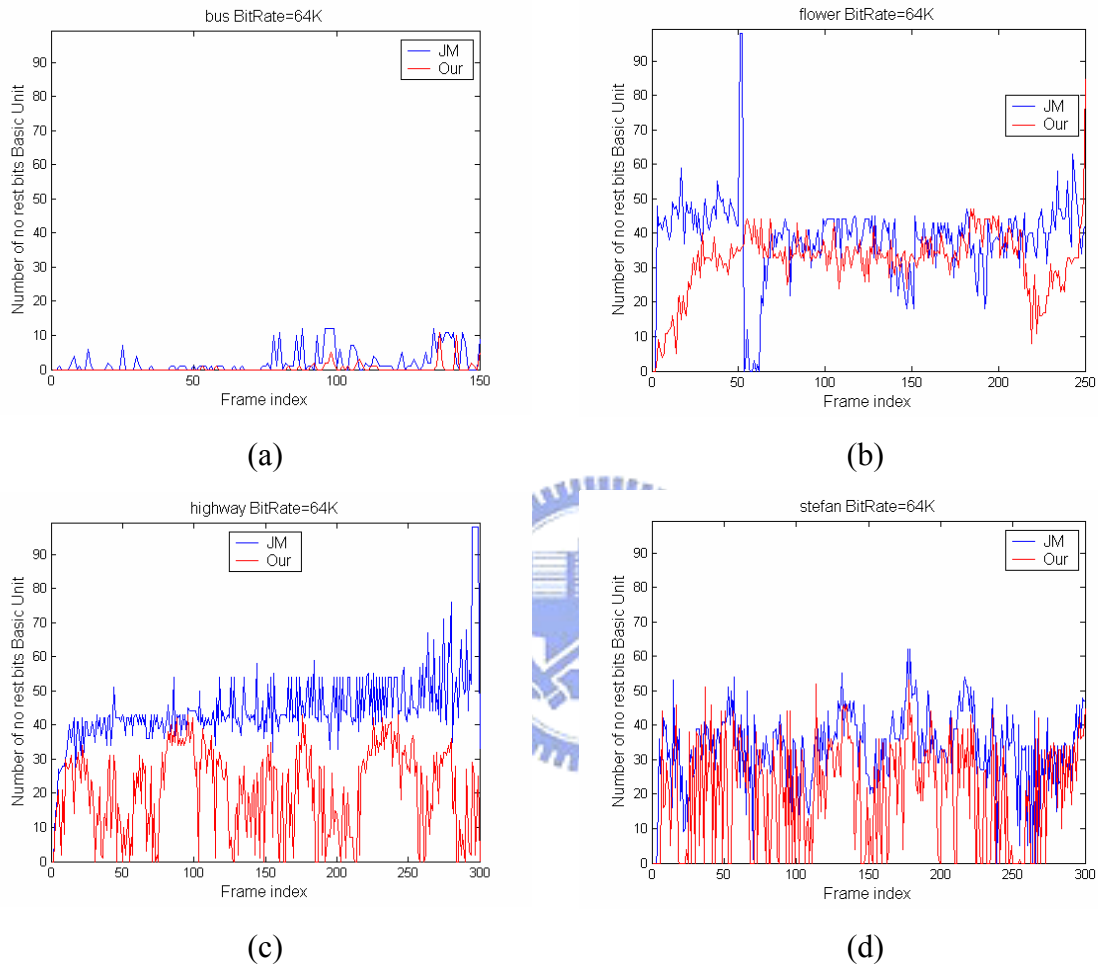


Figure 3-15. The number of basic units have no remaining target bits when the bit rate is 64K

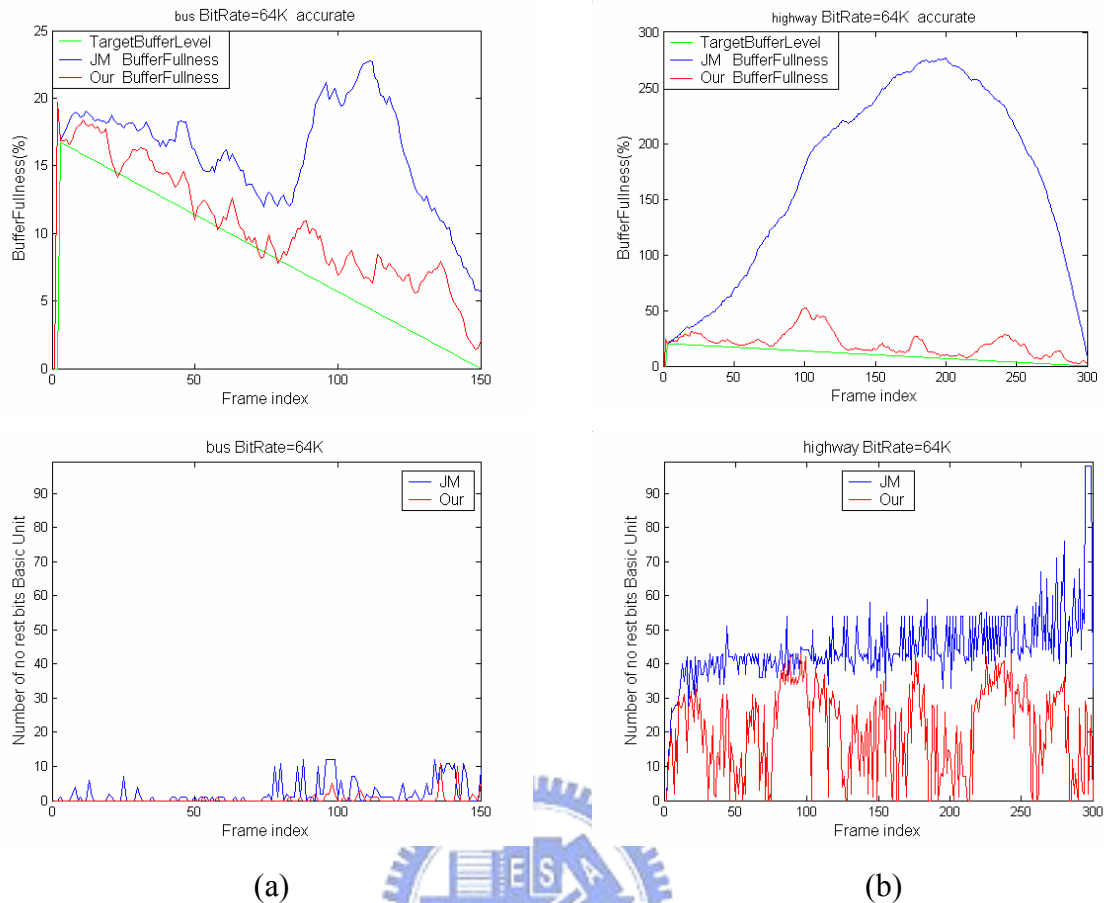
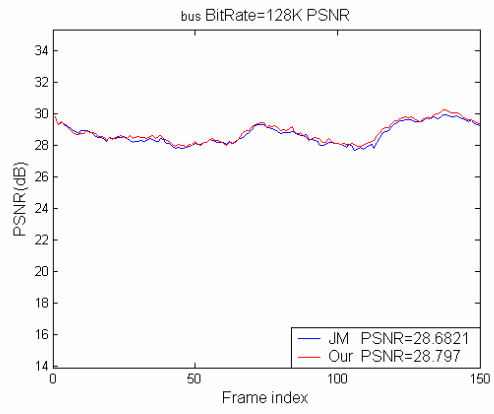
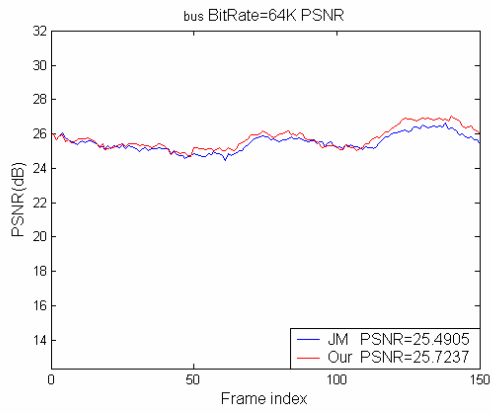
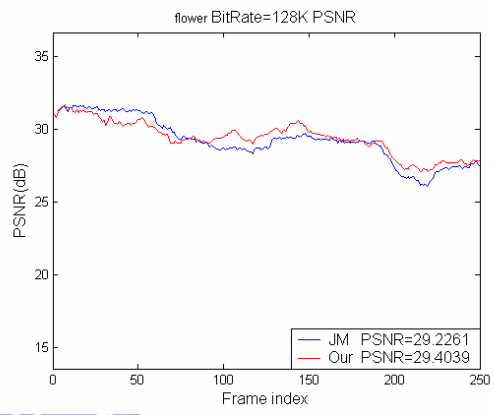
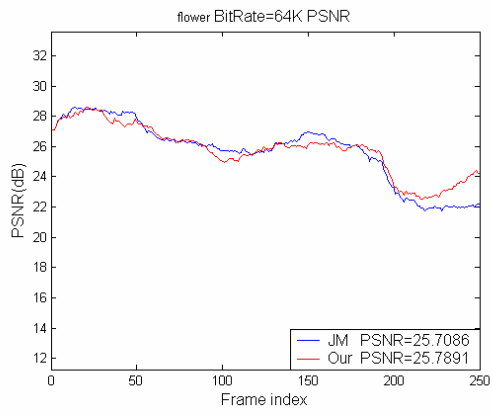


Figure 3-16 Compare the buffer fullness and the number of basic units that have no remaining target bits

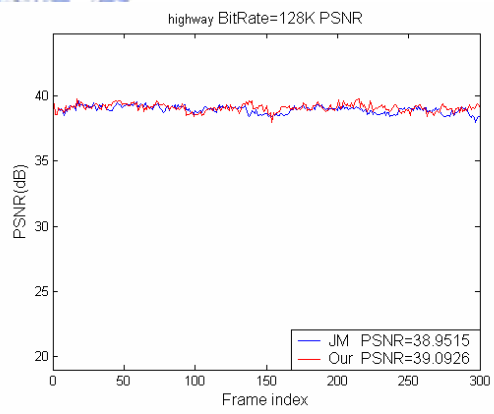
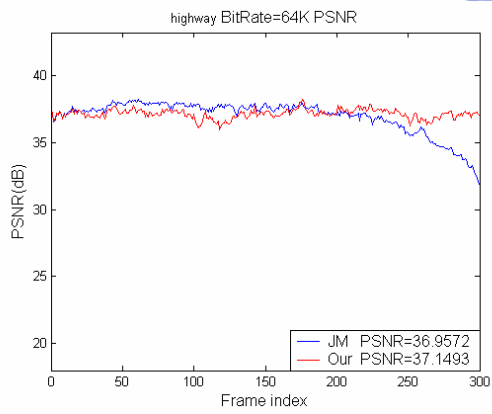
We can see in Figure 3-16 that the number of basic units that have no remaining target bits is proportional to the slope of the buffer fullness curve. That is because the overly coded bits will cause the rise of buffer fullness. In Figure 3-17, we compare the PSNR (Peak Signal to Noise Ratio) of each frame. The left figure in Figure 3-17 corresponds to the bit rate of 64K, while the right figure corresponds to 128K.



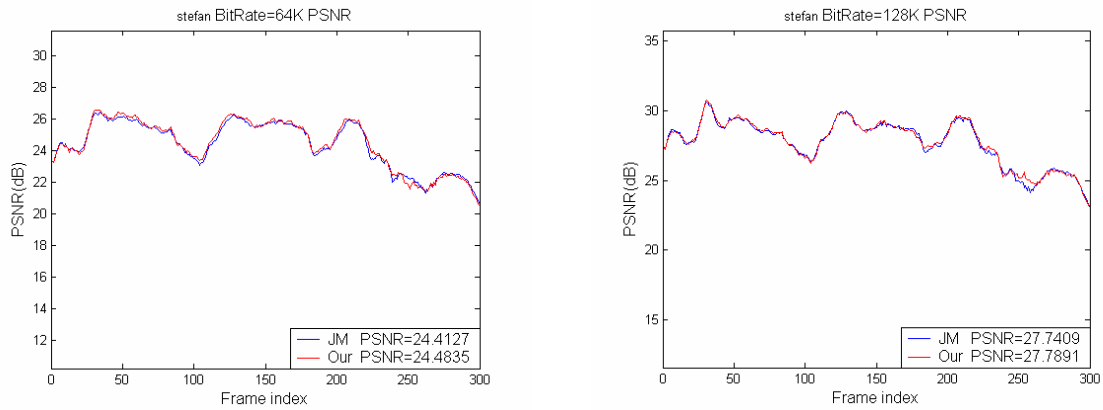
(a)



(b)



(c)



(d)

Figure 3-17 The PSNR(Y) of each frame in this experiment.

In Figure 3-17(a), we can see our PSNR is flatter than the original PSNR. This is because ours buffer fullness curve is closer to the target buffer level. Besides the improvement of PSNR, there are also improvements on the visual quality. In Figure 3-18, we can find the obvious differences between the original images and our images. The differences usually happen at bottom half of the frame. The reason of this difference is caused by the fact that an incorrect model may cause the macroblocks in the bottom half frame to have no remaining target bits.



Original

Our

(a)



Original



Our

(b)



Original



Our

(c)



Original



Our

(d)

Figure 3-18 Comparison of visual quality when the bit rate is 64K,

(a) The 145-th frame of bus, (b) The 220-th frame of flower,

(c) The 300-th frame of highway, (d) The 50-th frame of stefan

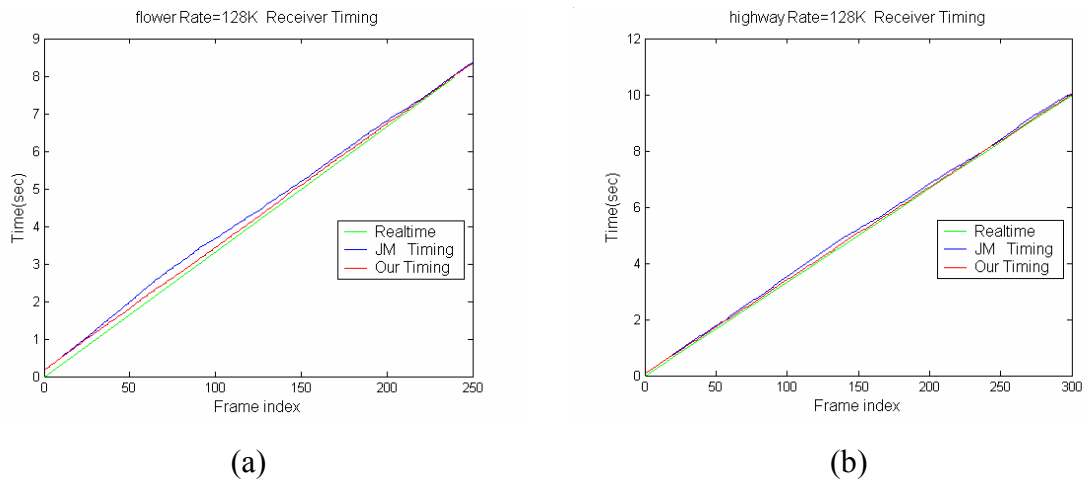


Figure 3-19 The receiver timing analysis when the bit rate is 128K

Finally, we simulate the decoder delay in real-time transmission. A longer delay means that the decoder needs a larger buffer. In Figure 3-19, the green line denotes the expected receiver timing in real-time transmission. If the line is above the green line, it means the decoder needs a delay time to display this image. The experiments of the transmission delay is described in Table 3-2.

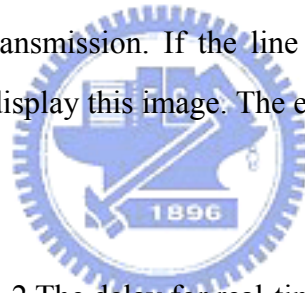


Table 3-2 The delay for real-time transmission

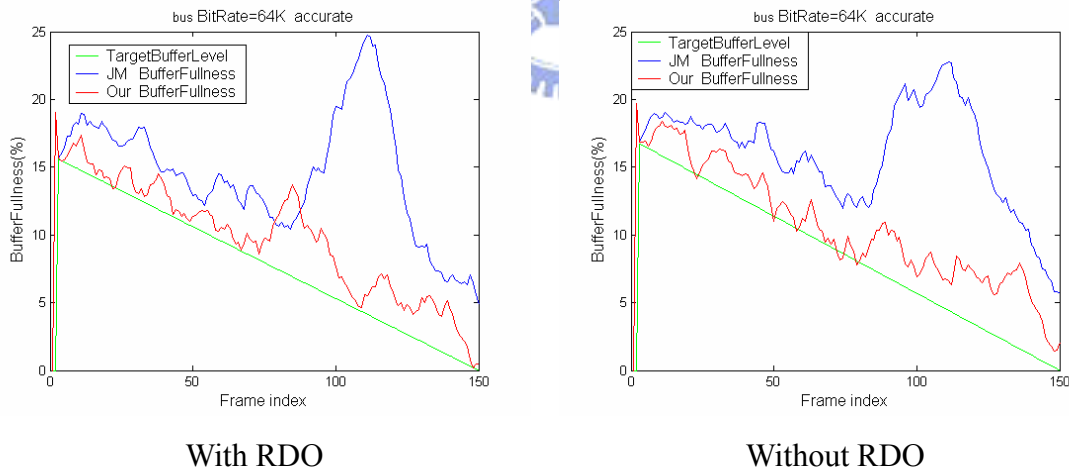
Sequence	Bit Rate	Transmission Delay (sec)		Improvement
		JM	Our	
Bus	64K	0.1138	0.0984	0.0154
	128K	0.1082	0.1082	0
Flower	64K	0.8438	0.5564	0.2874
	128K	0.4082	0.1780	0.2302
Highway	64K	1.3820	0.2629	1.1191
	128K	0.2709	0.1050	0.1659
Stefan	64K	0.1020	0.1020	0
	128K	0.1054	0.1054	0

Finally, in Table 3-3, we compare the accuracy between the original JM R-D model and our model. Here, “error” means the mean of the absolute difference between the target bits and the actually coded bits for each frame.

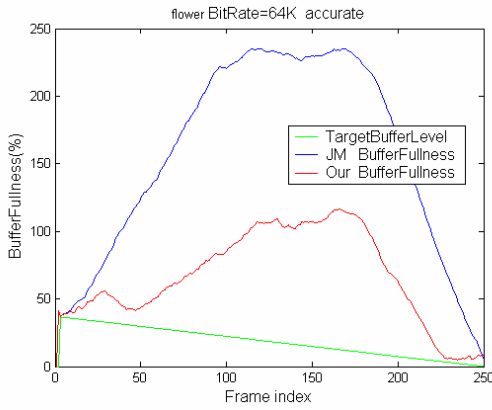
Table 3-3 The accuracy of R-D model

Sequence	Bit Rate	Error (MAD)		Improvement (JM-Our)/JM
		JM	Our	
Bus	64K	569.7	193.7	0.66
	128K	224.2	150.5	0.33
Flower	64K	1185.9	995.2	0.16
	128K	1965.9	700.7	0.64
Highway	64K	1320.2	572.9	0.57
	128K	1947.4	624.3	0.68
Stefan	64K	245.2	163.28	0.33
	128K	298.54	223.49	0.25

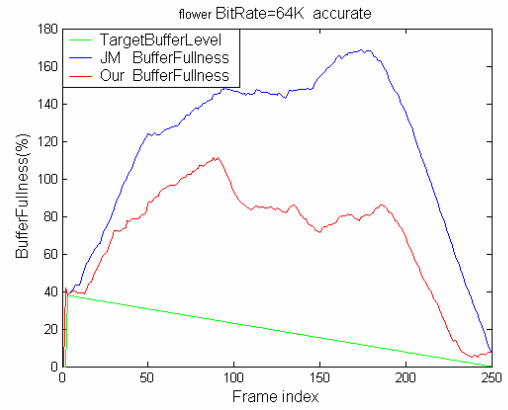
In the above experiments, videos are coded without RDO (Rate-Distortion Optimization). In the following experiments, we compare the situation of buffer fullness with RDO or without RDO. In Figure 3-20 we can find that with RDO our proposed model still make improvements. Table 3-4 describes the accuracy of the R-D model when executing RDO.



(a)

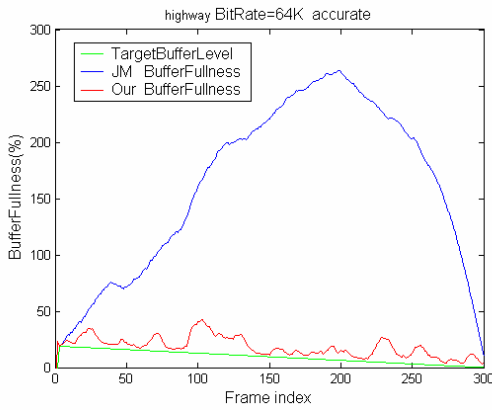


With RDO

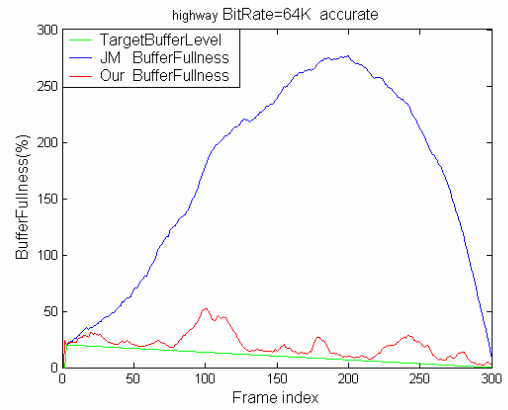


Without RDO

(b)

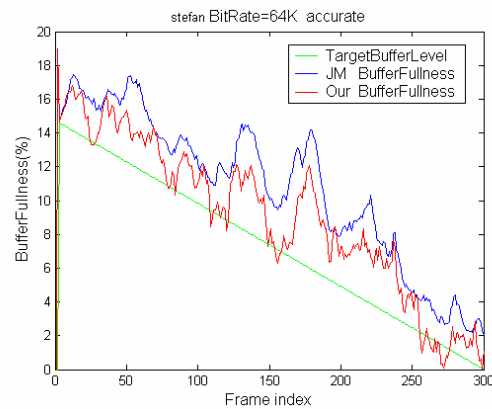


With RDO

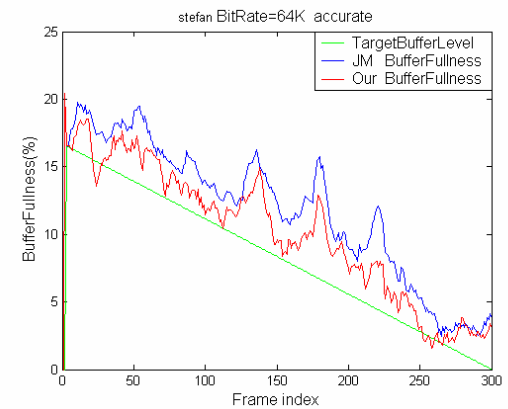


Without RDO

(c)



With RDO



Without RDO

(d)

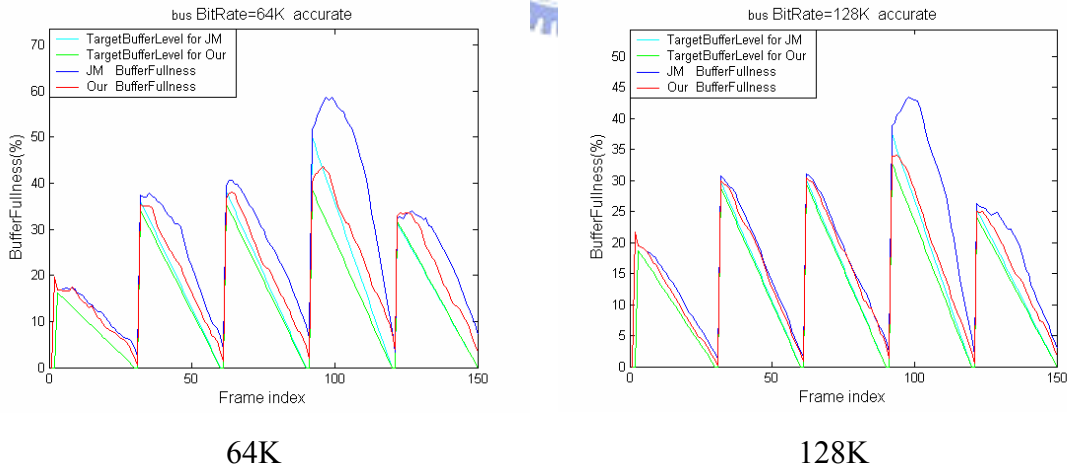
Figure 3-20 Comparison of buffer fullness with RDO and without RDO as the bit rate is 64K.

Table 3-4 The accuracy of R-D model when executing RDO

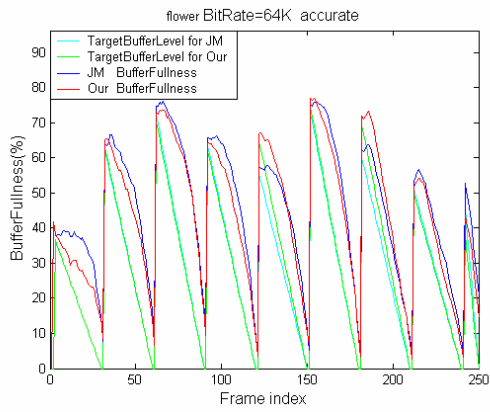
Sequence	Bit Rate	Error (MAD)		Improvement (JM-Our)/JM
		JM	Our	
Bus	64K	569.7	193.7	0.66
	128K	224.2	150.5	0.33
Flower	64K	1185.9	995.2	0.16
	128K	1965.9	700.7	0.64
Highway	64K	1320.2	572.9	0.57
	128K	1947.4	624.3	0.68
Stefan	64K	245.2	163.28	0.33
	128K	298.54	223.49	0.25

Case B. The size of GOP is 30

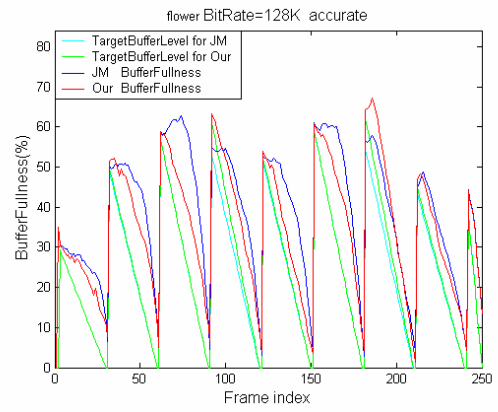
In the following experiments, we compare four aspects when encoding without RDO. These aspects are buffer fullness, PSNR, maximum delay for real time transmission, and the accuracy of R-D mode. As shown in Figure 3-21, our model has made improvements in avoiding buffer fullness. The improvements also cause the change of PSNR. As shown in Figure 3-22, the PSNR of our model is flatter than the original model.



(a) bus

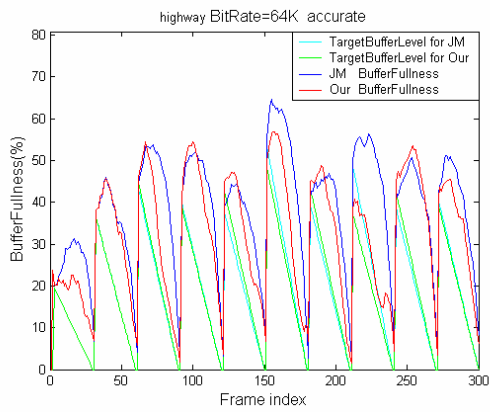


64K

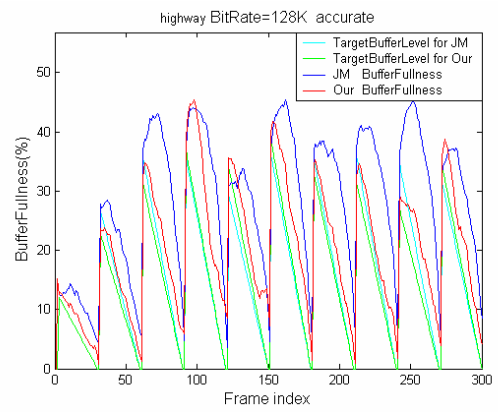


128K

(b) flower

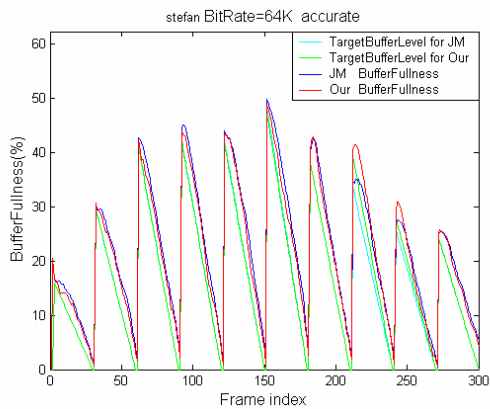


64K

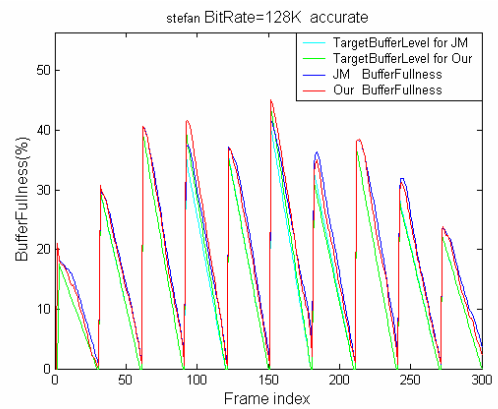


128K

(c) highway



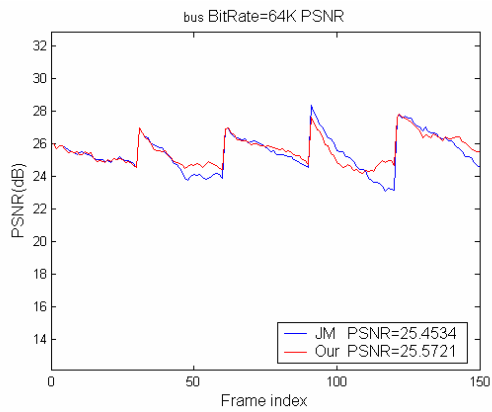
64K



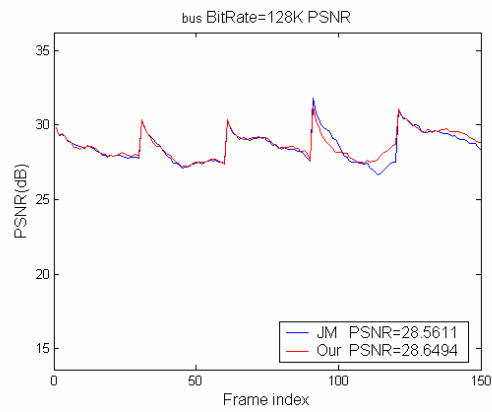
128K

(d) stefan

Figure 3-21 The buffer fullness when the period of I-frame is 30

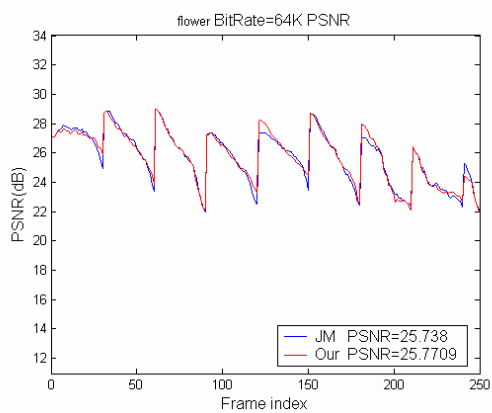


64K

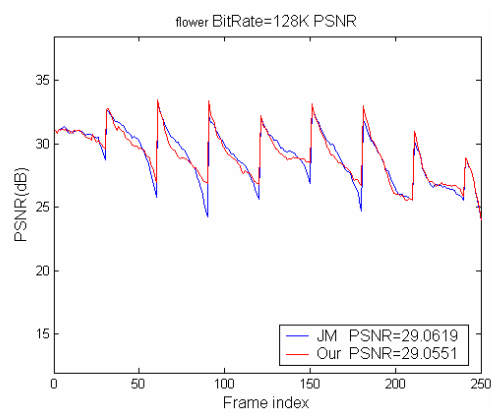


128K

(a) bus

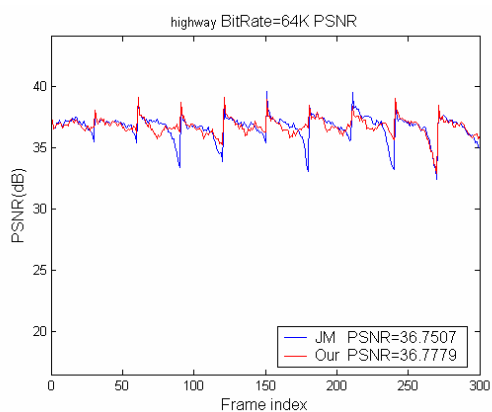


64K

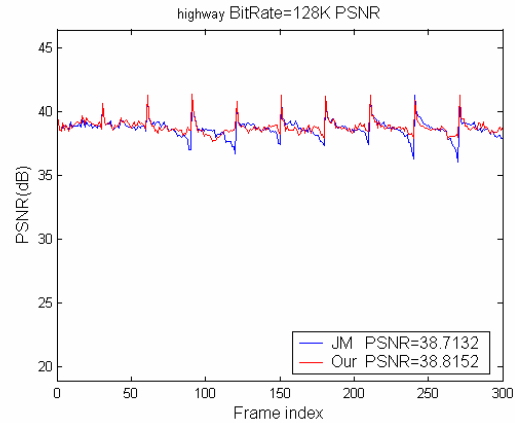


128K

(b) flower

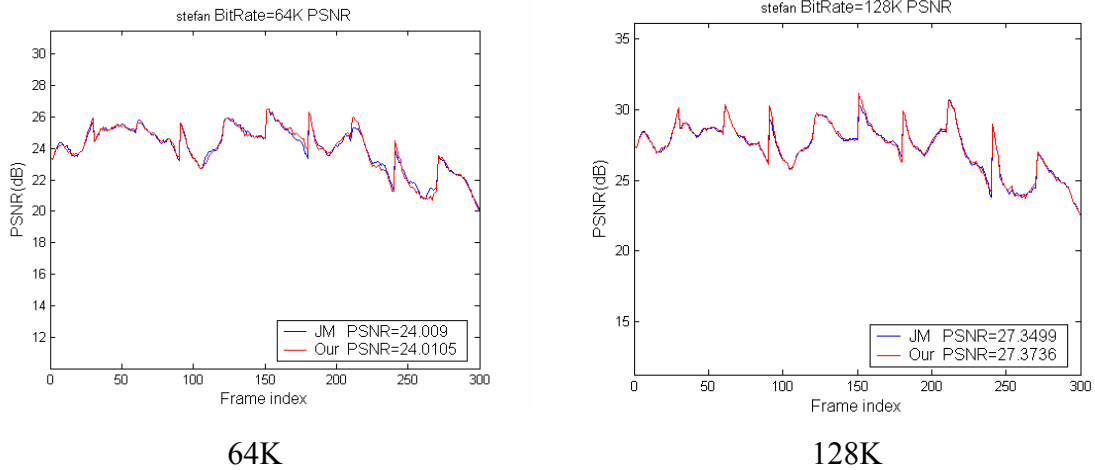


64K



128K

(c) highway



(d) stefan

Figure 3-22 The PSNR of each frame when the period of I-frame is 30

Table 3-5 shows the delay time of real-time transmission. Although the improvements are not as obvious as that in Table 3-2, we may still see the improvements. The accuracy of the R-D model is shown in Table 3-6. The improvement on accuracy is also obvious in this case. Finally we compare the accuracy of the R-D model from low bitrates to high bitrates. This verifies that our model is adaptive to different QP values. Here, when the basic unit is not a macroblock, we calculate the average MAD and target bits in macroblock size to determine the QP value for each basic unit.

Table 3-5 The delay for real-time transmission when the period of I-frame is 30

Sequence	Bit Rate	Transmission Delay (sec)		Improvement
		JM	Our	
Bus	64K	0.2934	0.2180	0.0754
	128K	0.2172	0.1704	0.0468
Flower	64K	0.3801	0.3849	-0.0048
	128K	0.3138	0.3363	-0.0225
Highway	64K	0.3232	0.2841	0.0391
	128K	0.2271	0.2267	0.0004
Stefan	64K	0.2491	0.2433	0.0058
	128K	0.2087	0.2257	-0.017

Table 3-6 The accuracy of R-D model when the period of I-frame is 30

Sequence	Bit Rate	Error (MAD)		Improvement (JM-Our)/JM
		JM	Our	
Bus	64K	990.6	614.2	0.38
	128K	1163.0	743.7	0.36
Flower	64K	1579.1	1453.1	0.08
	128K	2895.6	2436.5	0.16
Highway	64K	1496.4	1210.4	0.19
	128K	2530.4	1396.4	0.45
Stefan	64K	636.7	570.1	0.10
	128K	1079.4	955.9	0.11

Table 3-7 The accuracy of R-D model in high and low bit rates

Sequence	Bit Rate	Error (MAD)		Improvement (JM-Our)/JM
		JM	Our	
Bus	16K	602.4	600.4	0.003
	32K	479.5	371.3	0.23
	64K	990.6	614.2	0.38
	128K	1163.0	743.7	0.36
	256K	1047.2	989.3	0.06
	512K	1353.3	1327.3	0.02

3.3 The Relation between Header Bits and Macroblock Mode

In this section, we will analyze the header bits in different macroblock modes of P-frame. In H.264/AVC, inter-prediction coding has several macroblock modes. The number of macroblock header bits is variable in different macroblock modes. In Figure 3-23, we can see the header bits of 16×16, 16×8 and 8×16 macroblock modes are basically irrelative to MAD, but the header bits of the 8×8 mode are proportional to the MAD value of the macroblock. The variances of 16×16, 16×8 and 8×16 header bits are very small, but it is not the case in the 8×8 mode. Table 3-8 lists the elements of macroblock bits.

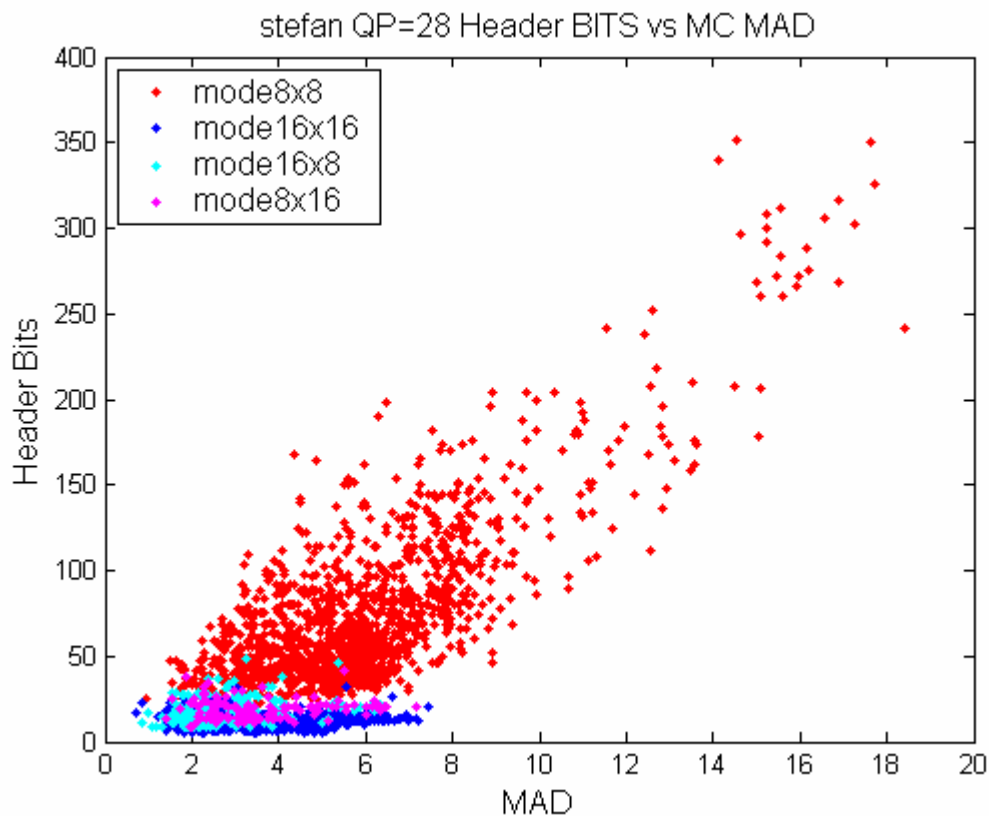


Figure 3-23 The relation between the header bits and the MAD value of the macroblock

Table 3-8 The elements of macroblock header bits in P-slice

Elements	Explanation
Macroblock skip run	The flag if this macroblock is skipped
Macroblock type	The type of the macroblock mode
Motion vector	Motion vector data
CBP	Coded Block Patterns to prevent the need for transmitting EOB (End Of Block) symbols in these zero coded block.
Delta QP	The QP difference with respect to the previous macroblock

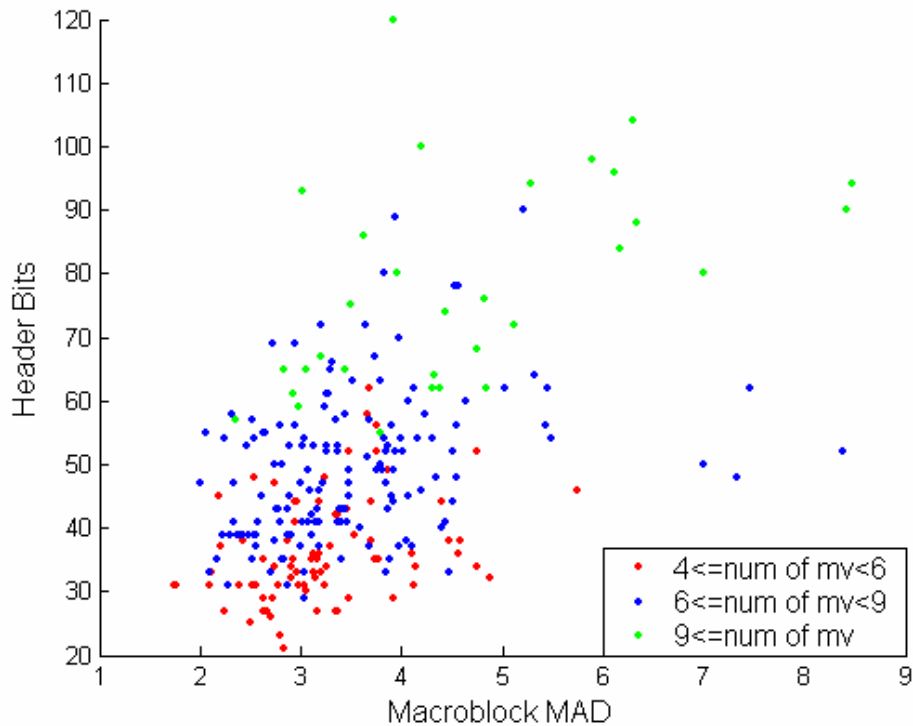


Figure 3-24 The relation between the macroblock header bits and the macroblock MAD for different numbers of motion vectors.

In the 8x8 mode, the motion vector data are variable. This is because that an 8x8 block can be divided into 8x4, 4x8, or 4x4 modes. A smaller block size causes more motion vectors

and thus more header bits are coded. In , we can observe such a relation. Hence, a larger MAD has a larger chance to encode more motion vectors. Based on this observation, we model the relation between the 8×8 header bits and the MAD value of the macroblock as a linear model (3-11).

$$Hbits_{8 \times 8} = a \times MAD + b \quad (3-11)$$

where a and b are the coefficients of this model. Besides the 8×8 mode, we take the header bits of other modes, such as 16×16, 16×8, and 8×16 modes, as the average values of previous macroblock. In the next section, we will use this relation to revise and experiment the R-D model.

3.4 Frame Level Header Bits Prediction

The header bits prediction is needed in the R-D model. In Equation (2-3), the quadratic R-D model needs the predicted number of header bits. In JM, a simple way is used to estimate the number of head bits. It takes the number of the previous macroblock header bits as the predicted number of header bits $m_{h,i}(j)$. Hence, in JM, we have

$$T_i(j) = c_1 \times \frac{\tilde{\sigma}_i(j)}{Q_{step,i}(j)} + c_2 \times \frac{\tilde{\sigma}_i(j)}{Q_{step,i}^2(j)} + m_{h,i}(j)$$

In this section, we change the predicted number of header bits based on the observation discussed in Section 3.3. The header bits of a frame are actually a combination of the header bits from several macroblocks that can have different modes. Hence, we can divide the header bits in a frame into several parts, as shown in Figure 3-25. After dividing, we predict the number of each macroblock mode in the current frame and compute the average bits of each macroblock mode. Equation (3-12) formulates our header bits prediction.

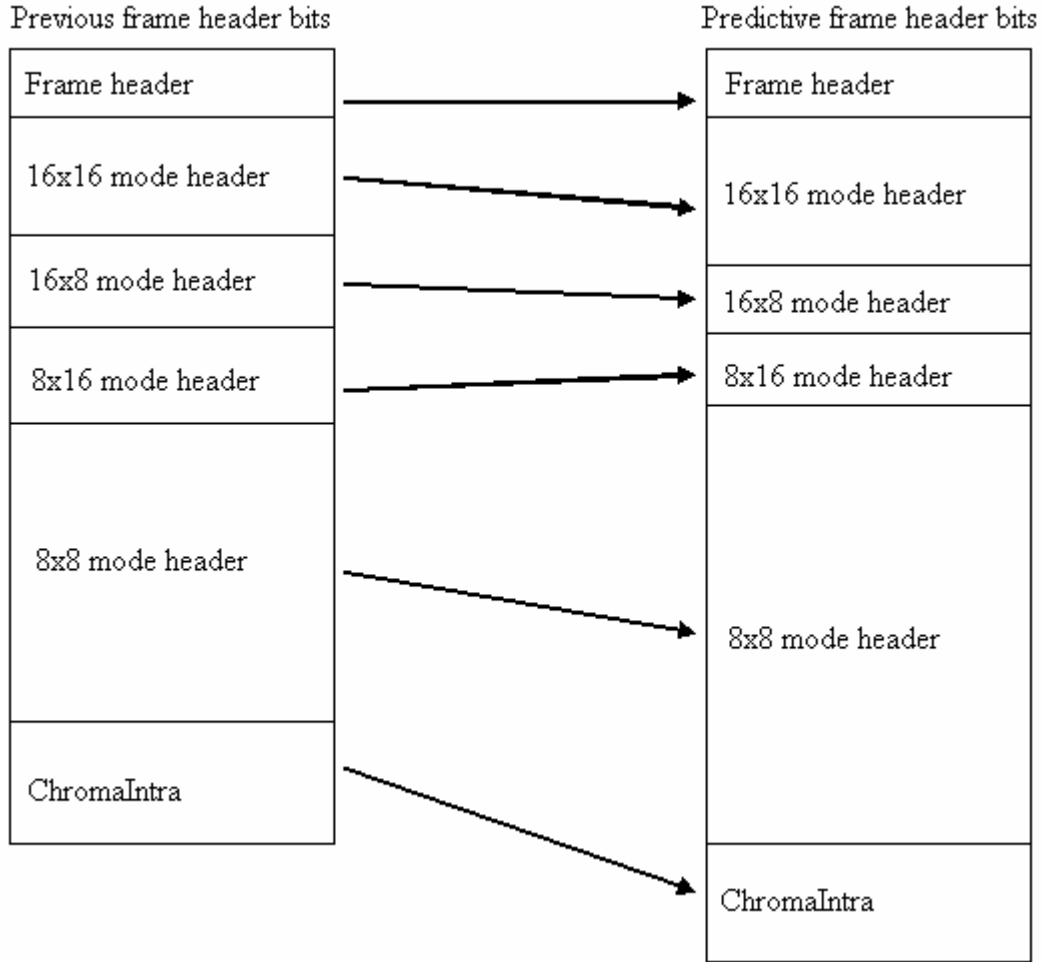


Figure 3-25 Proposed frame-level header bits prediction

$$\tilde{h}(j)^t = \frac{\tilde{N}(j)^t}{N(j-1)^t} \times h(j-1)^t$$

$$\tilde{h}_{frame}(j) = \sum_t \tilde{h}(j)^t \quad t = 16 \times 16, 16 \times 8, 8 \times 16, 8 \times 8, chromaintra \quad (3-12)$$

where \tilde{h} and h are the predicted and actual numbers of coded header bits, and j means the j -th pictures. The symbol t denotes the type of macroblock mode. N and \tilde{N} are the actual and predicted macroblock number. Hence, we take a linear model to predict the number of macroblock mode and compute the total predicted number of frame-level header bits. In the following experiments, we compare the precision of header bits prediction. The error means the mean of absolute difference between the predicted number header bits and the actual number of coded header bits, with the GOP format being IPPPP...

Table 3-9 Experiments of the header bits prediction

Sequence	Bit Rate	Error (MAD)		Improvement (JM-Our)/JM	PSNR(Y)	
		JM	Our		JM	Our
Bus	32K	102.28	118.88	-0.16	22.81	22.80
	64K	230.50	195.20	0.15	25.85	25.84
	128K	448.91	307.23	0.32	28.9	28.89
	256K	455.74	341.97	0.25	32.35	32.35
	512K	618.10	435.25	0.30	36.50	36.50
Flower	32K	149.36	117.82	0.21	23.03	23.00
	64K	346.00	220.79	0.36	26.25	26.23
	128K	466.33	303.87	0.34	29.65	29.63
	256K	487.02	370.80	0.24	33.36	33.36
	512K	437.03	327.30	0.25	37.54	37.54
Highway	32K	246.28	217.30	0.12	34.98	35.04
	64K	406.40	308.47	0.24	37.14	37.15
	128K	521.44	354.06	0.32	39.22	39.22
	256K	661.20	423.24	0.36	40.86	40.86
	512K	332.83	324.61	0.02	42.13	42.12
Stefan	32K	213.48	263.36	-0.23	21.55	21.54
	64K	370.37	279.81	0.24	24.62	24.63
	128K	520.11	390.53	0.25	27.95	27.95
	256K	498.20	456.32	0.08	31.58	31.56
	512K	609.49	562.51	0.08	35.73	35.74

In Table 3-9, we can see that there is an expectable improvement in common cases. However, the improvement of PSNR is not obvious, and sometimes we may even have worse results. In this experiment, we know that a complicated prediction may not necessarily provide a better result for rate control. Hence, we maintain the original header bits prediction in our rate control scheme and provide an option if an accurate prediction is desired.

Chapter 4

Bit allocation of H.264/AVC

In Chapter 3, we discussed the R-D model for the rate control of H.264/AVC. Before applying the R-D model to the estimation of the QP value, we need to determine the target bits. Target bit means the expected coded bits as we code a frame or a macroblock. The allocation of target bits can affect the buffer fullness and video quality. In this chapter, we focus on the bit allocation of target bits in frame level and macroblock level. In frame level, we will determine the target bits for each frame (picture). In the following, we'll first discuss the target bits in frame level.

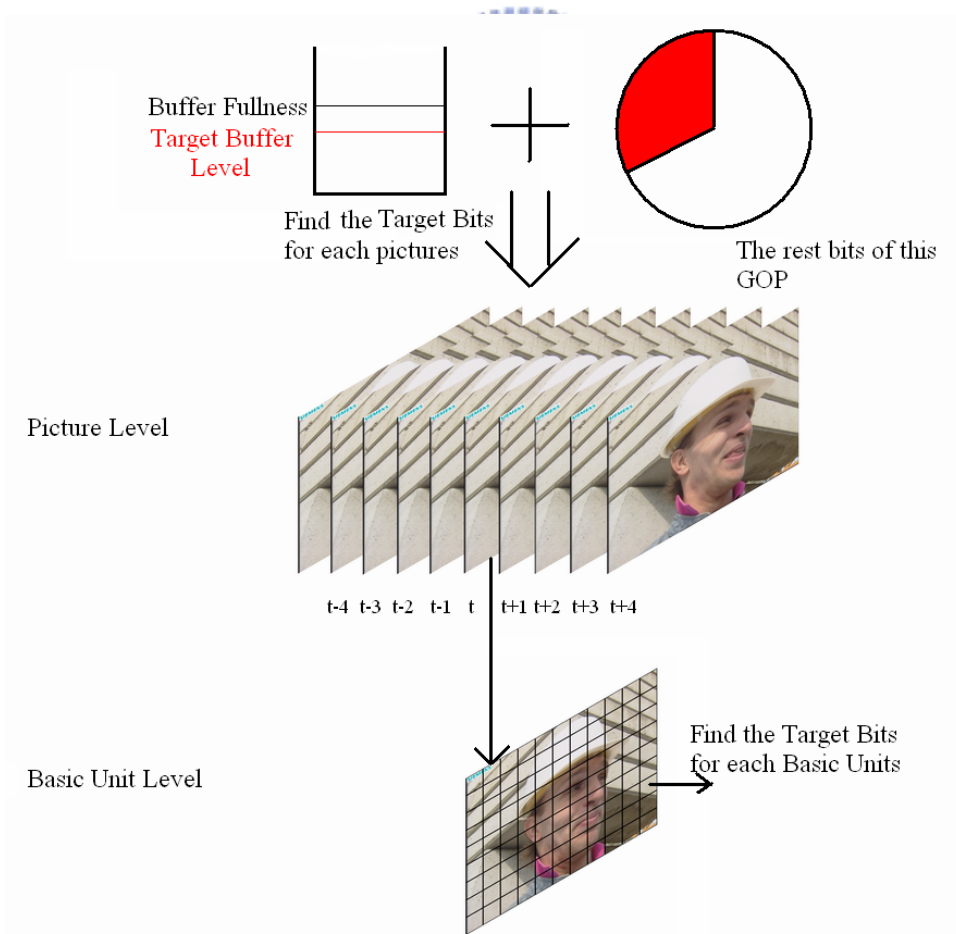


Figure 4-1 The diagram of the bit allocation for picture level and basic-unit level

4.1 Frame Level

In JM model of H.264/AVC, the target bits are determined by two parts, as expressed in Equations (2-13) and (2-14). Equation (2-13) uses the buffer fullness, target buffer level, and average picture bit rate to determine the target bits. The strategy is that when buffer fullness is higher than the target bits, we use fewer bits to encode the data and thus reduce the level of buffer fullness. Equation (2-14) determines the target bits according to the remaining bits of this GOP and averages the remaining bits by checking the picture complexity.

$$\tilde{T}_i(j) = \frac{R_i(j)}{f} + \gamma \times (S_i(j) - V_i(j)) \quad \gamma = 0.5$$

$$\hat{T}_i(j) = \frac{W_{p,i}(j-1) \times B_i(j)}{W_{p,i}(j-1) \times N_{p,r} + W_{b,i}(j-1) \times N_{b,r}}$$

Apart from the above two strategies, there are other methods that are proposed for bit allocation. For example, Jiang and Lin provide the PSNR-based frame complexity estimation to determine the target bits [16]. In this section, we propose some ideas that can be used in bit allocation of frame target bits. In Section 4.1.1, we provide a parameter that can represent the frame complexity. In Section 4.1.2, we raise the importance of the beginning pictures of each GOP and add this importance into the consideration of bit allocation. Finally in Section 4.1.3, we indicate the fact that more bits coded in this frame can cause the decrease of the residual of the next frame. However, as too many bits are consumed in the coding of the current frame, there could be inadequate remaining target bits for the coding of the following frames. Hence, we will discuss the trade-off of this relation in that section.

4.1.1 Frame Complexity

When encoding video with a constant quantization value, the numbers of coded bits for each frame are different. That is because that the frame complexity in each frame is different. Traditionally, the MAD (mean absolute difference) value is usually used to represent the frame complexity. In theory, a larger MAD will correspond to the encoding of more bits. Here, we simulate the relation between the MAD value and the number of coded bits with a constant quantization value. Figure 4-2 describes the relation between the number of coded bits and the MAD value for a sequence of frames when the QP=6, 16, 26, 36 and 46. We can see the number of coded bits is directly proportional to the MAD value when the QP is small, like QP=6, 16 and 26. However, this proportional property is not found when QP = 36 and 46. When the QP value is large, the error propagation will cause the MAD value to rise dramatically. Hence, using MAD to represent the frame complexity may be not suitable when the QP value is large. Here, we try to find a factor that can represent the frame complexity.

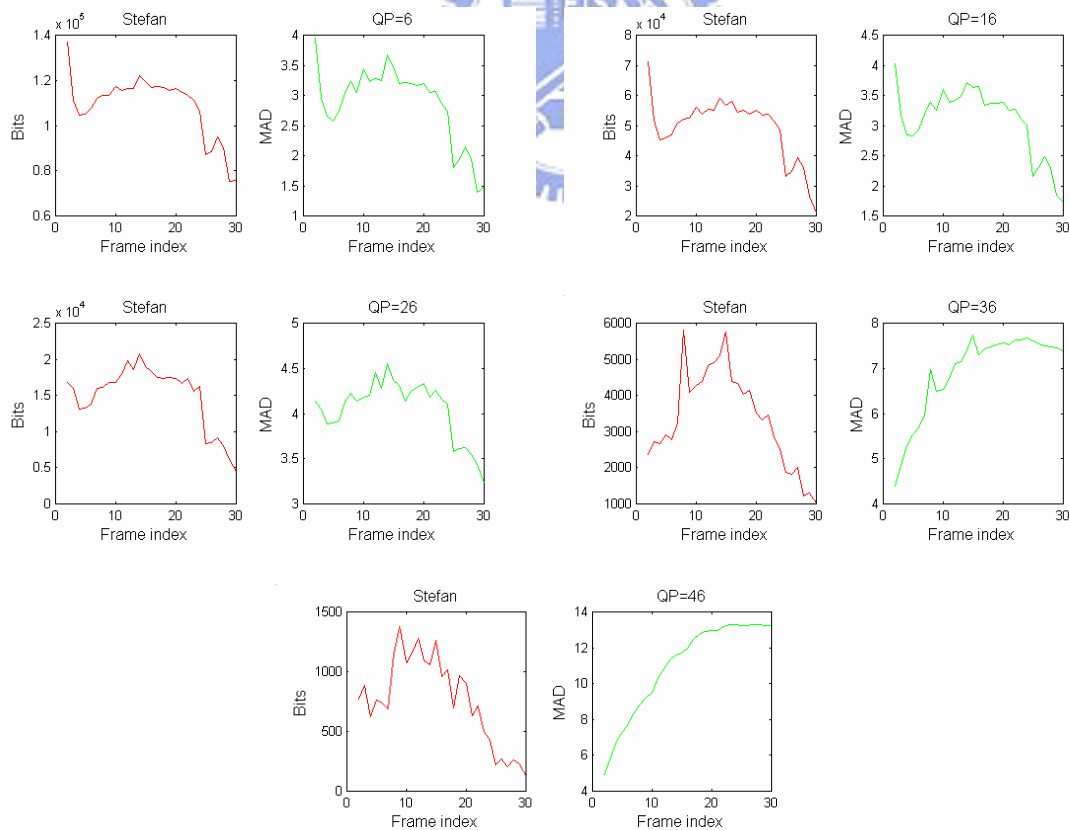


Figure 4-2 The relations between the number of coded bits and MAD

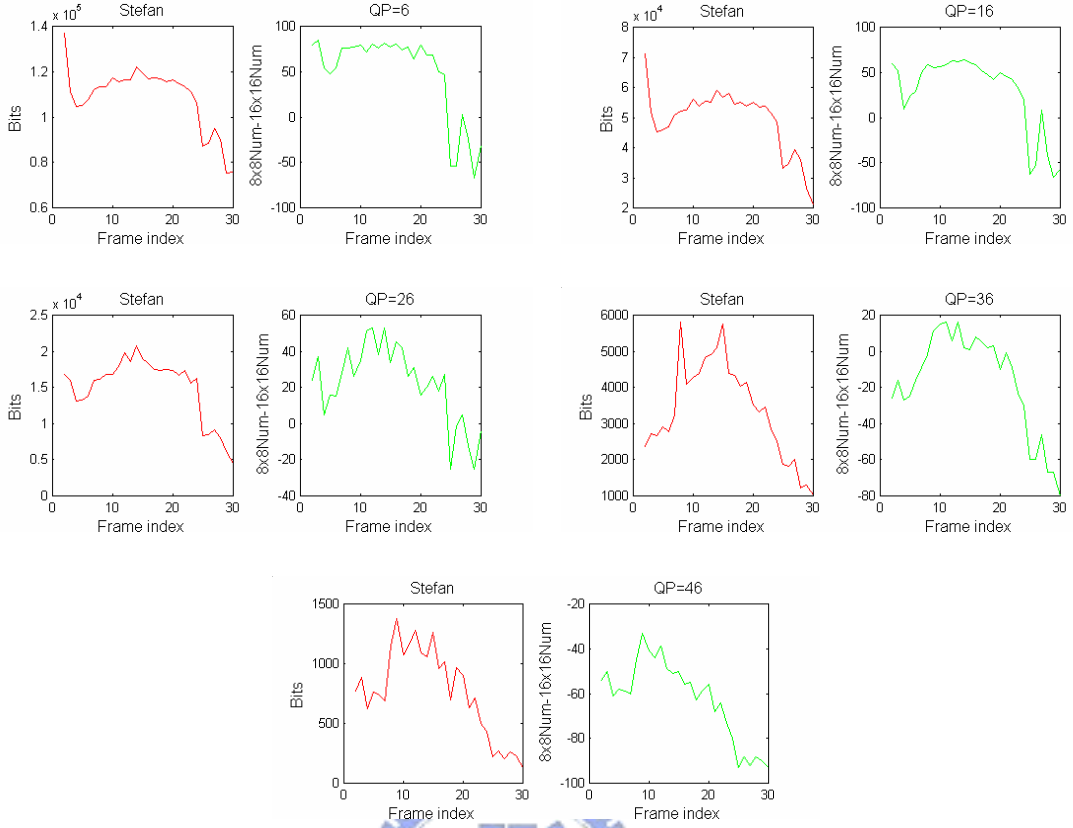


Figure 4-3 The relations between the number of coded bits and the difference between the number of 8×8 macroblocks and the number of 16×16 macroblocks.

In Figure 4-3, we can observe that the difference between the number of the 8×8 macroblocks and the number of 16×16 macroblocks is proportional to the number of coded bits. This is because when the QP value is large, the number of 16×16 macroblocks will increase. Hence, using the number of 8×8 or 16×16 macroblocks could be more helpful than using the MAD value, especially when the bit rate is low. Hence, after the target bits are determined by Equation (2-15), we add another equation (4-1) to re-estimate the target bits.

$$T_i(j) = (1-c) \times T_i(j) + c \times b_i(j-1) \times w$$

$$w = \begin{cases} \frac{\tilde{N}_{8 \times 8}(j) - \tilde{N}_{16 \times 16}(j)}{N_{8 \times 8}(j-1) - N_{16 \times 16}(j-1)} & N_{8 \times 8}(j-1) - N_{16 \times 16}(j-1) > 0 \\ \frac{N_{8 \times 8}(j-1) - N_{16 \times 16}(j-1)}{\tilde{N}_{8 \times 8}(j) - \tilde{N}_{16 \times 16}(j)} & N_{8 \times 8}(j-1) - N_{16 \times 16}(j-1) < 0 \end{cases} \quad (4-1)$$

where $N_{8 \times 8}$ is the actual number of 8×8mode macroblocks and $\tilde{N}_{8 \times 8}$ is the predicted number of 8×8mode macroblocks. The coefficient c is generally a small value, such as 0.1 or 0.2. In the

following, we compare the coded bits with rate control and a constant quantization value. If X is the array of the coded bits of each frame with rate control and Y is the array of the coded bits with a constant QP, the correlation coefficient between X and Y arrays can represent their correlation. The correlation coefficient and the average PSNR are listed in Table 4-1.

Table 4-1 The correlation coefficients table

Sequence	Bit Rate	Coefficient c	Correlation Coefficient	PSNR(Y)
Bus	64K	Original	0.143	25.85
		0.1	0.180	25.83
		0.2	0.097	25.89
	128K	Original	0.024	28.90
		0.1	0.002	28.91
		0.2	0.056	28.91
Highway	64K	Original	0.272	37.14
		0.1	0.265	37.18
		0.2	0.246	37.18
	128K	Original	0.315	39.22
		0.1	0.312	39.23
		0.2	0.182	39.20
Stefan	64K	Original	0.037	24.62
		0.1	0.032	24.58
		0.2	0.012	24.63
	128K	Original	0.061	27.95
		0.1	0.041	27.98
		0.2	0.053	27.9

In Table 4-1, we can see the improvement is not obvious. Sometime, the correlation coefficient is even reduced. This is because different QP values will cause the change of the number of each macroblock mode. Hence, the relation between the number of macroblock

and frame complexity will not be obvious.

4.1.2 Frame Importance

In Joint Model, each P-frame of the GOP has the same importance. Hence, the plot of the target buffer level is like Figure 4-4 (a). After coding the first I-frame, the target buffer level rises up to 50% and a normal target buffer level hopes to decrease the buffer level in a uniform way. In stead, our idea is that the first few pictures in a GOP are more important than the other pictures. This is because the quantization errors occurring in the first few pictures will have a longer propagation. To reduce the quantization errors in the first few frames, we raise the importance of the beginning frames and allow the buffer level to decrease slowly there. In Figure 4-4(b), we test four different curves for the target buffer level. When the value p is small, the decreases are small in the beginning pictures. However, this arrangement may cause the decrease of the buffer level to be too large when we code the remaining pictures. When the bit rate is constant and the profile is the baseline profile, the equation (2-11) can be rewritten as following.

$$S_i(2) = V_i(2)$$

$$S_i(j+1) = S_i(j) - \frac{S_i(2)}{N_p(i) - 1} \quad (4-2)$$

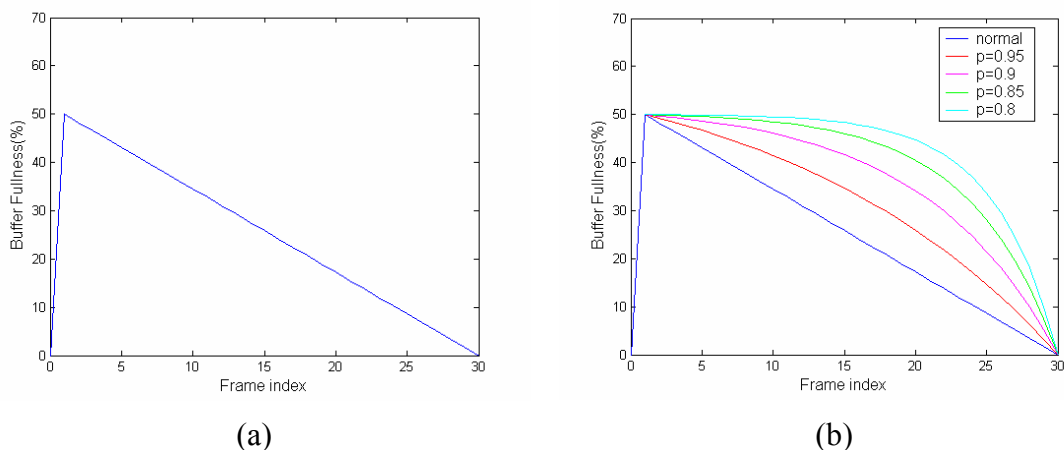


Figure 4-4 The target buffer level control, (a) the normal target buffer level, (b) non-linear target buffer level.

Then, we change the target buffer level to be a nonlinear curve. The equation is written as Equation (4-3).

$$S_i(2) = V_i(2)$$

$$S_i(j+1) = S_i(j) - S_i(2) \times \frac{(1-p)}{(1-p^{N_p(i)-1})} \times p^{N_{p,r}(i)} \quad (4-3)$$

where the N_p is the number of P-frame in a GOP and $N_{p,r}$ is the remaining number of P-frame in this GOP. Generally, we decide the p value to be 0.95 or 0.9 to avoid a dramatic decrease in the buffer level. This is because a large decrease in the buffer level may cause a serious degrading of video quality. After applying (4-3), we simulate how the video PSNR is changed by the changing of the target buffer level. In Figure 4-5, we can find the PSNR of the beginning pictures in each GOP are higher than that in the original case. However, this change may cause the drop of PSNR for the remaining pictures in the same GOP. Hence, this may cause a violent change of the video quality. Moreover, it also become easier to have a buffer overflow.

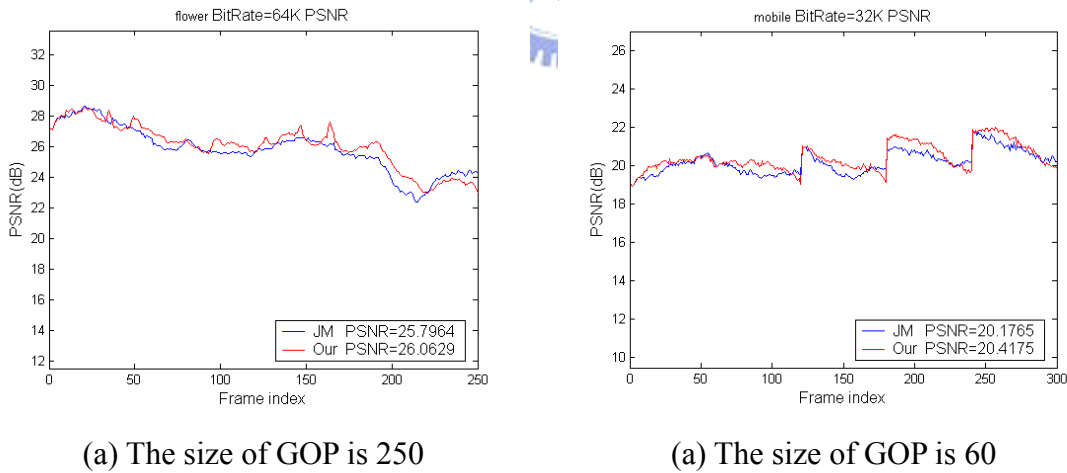


Figure 4-5 The experiment of change frame importance. (a)flower sequence (b)mobile sequence

4.1.3 Selection different QP causes the MAD

change of next frame

Inter-prediction coding compresses the difference data between two pictures. In the rate control scheme, the inter-prediction coding is the most important part. In inter-prediction coding, if a larger compression distortion is generated in the current picture, the residual in the next picture will increase. In other words, coding more bits in the current picture may cause a reduce of data in the following pictures. In Figure 4-6, show an experiment of this property.

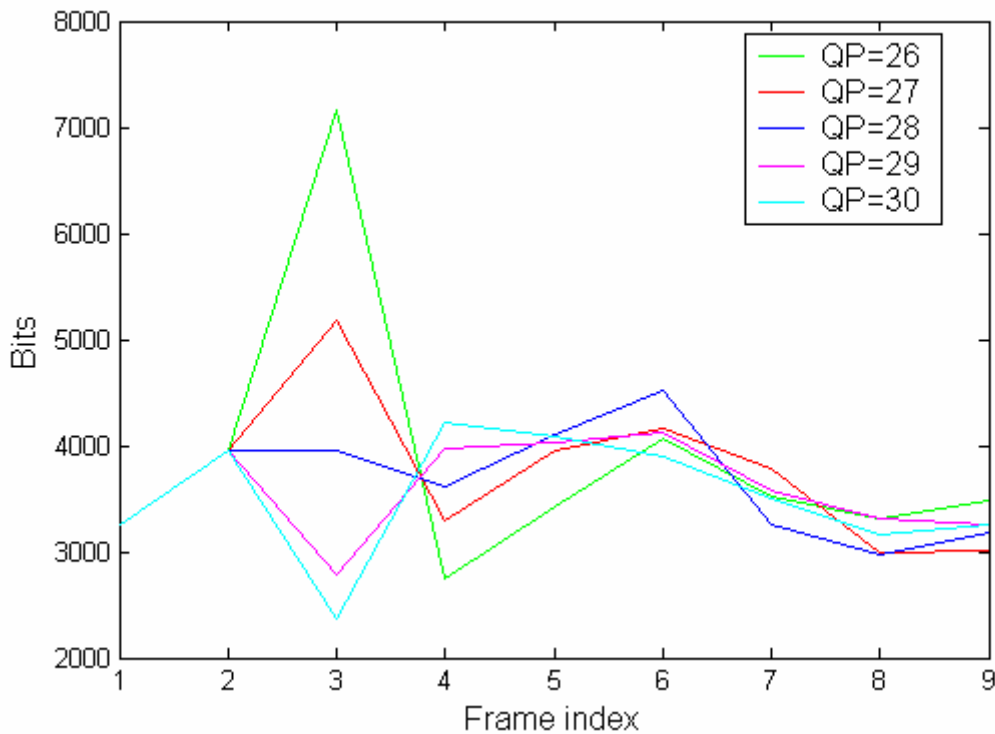


Figure 4-6 The data of the following pictures when coding the current picture with different QP's.

In Figure 4-6, all pictures, except the 3rd picture, are coded with $QP = 28$. We can see that if we encode the 3rd picture with more bits, it will cause the reduce of the residuals in the following pictures. In rate control, we determine the QP value according to buffer fullness and frame complexity. Now, our strategy is that after determining the QP value q , we can predict

the situations of buffer fullness when coding with $q-1$, q and $q+1$, respectively. Then, we can determine the QP value of the next picture according to our predicted buffer fullness and the predicted MAD. The flow chart is plotted as follows.

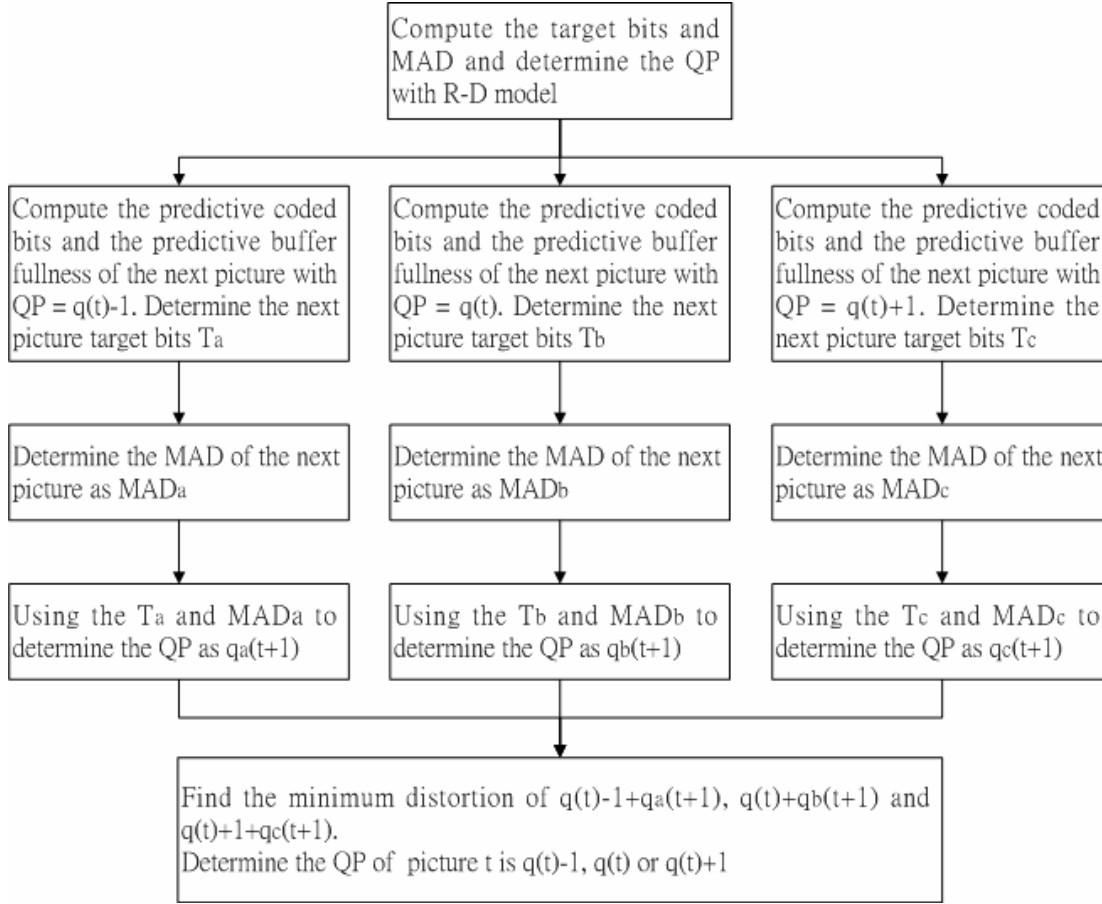


Figure 4-7 The flow chart of our strategy in QP decision

In the above flow chart, the predicted coded bits are computed by the R-D model. After that, we can compute the predicted target bits of the next picture. In the following, we will discuss how to compute the MAD of the next picture. First, we simulate the change of the MAD values of the next picture when encoding the current picture with different QP values. Figure 4-8 shows the relation between the MAD of the next picture and the QP of the current picture. There are the MAD differences when the values of QP are 27, 28 and 29, respectively, with the “foreman” sequence. The differences of MAD from 29 to 28 and from 28 to 27 are very similar. Hence, we define the symbol m as the difference between two QP’s. The relation of these MAD’s when the values of QP are 27, 28 and 29, is as following.

$$\begin{aligned}
 MAD_{QP=27} &= MAD_{QP=28} - m \\
 MAD_{QP=29} &= MAD_{QP=28} + m
 \end{aligned}
 \tag{4-4}$$

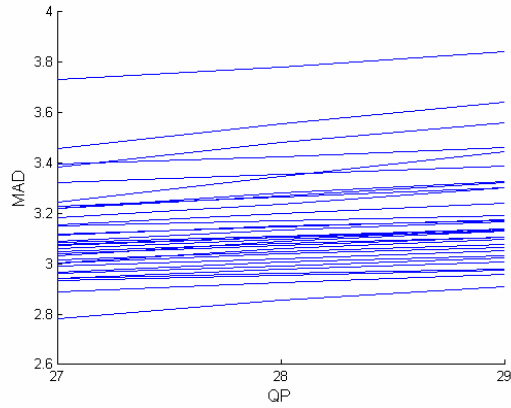


Figure 4-8 The relation between the value of MAD and the value of QP

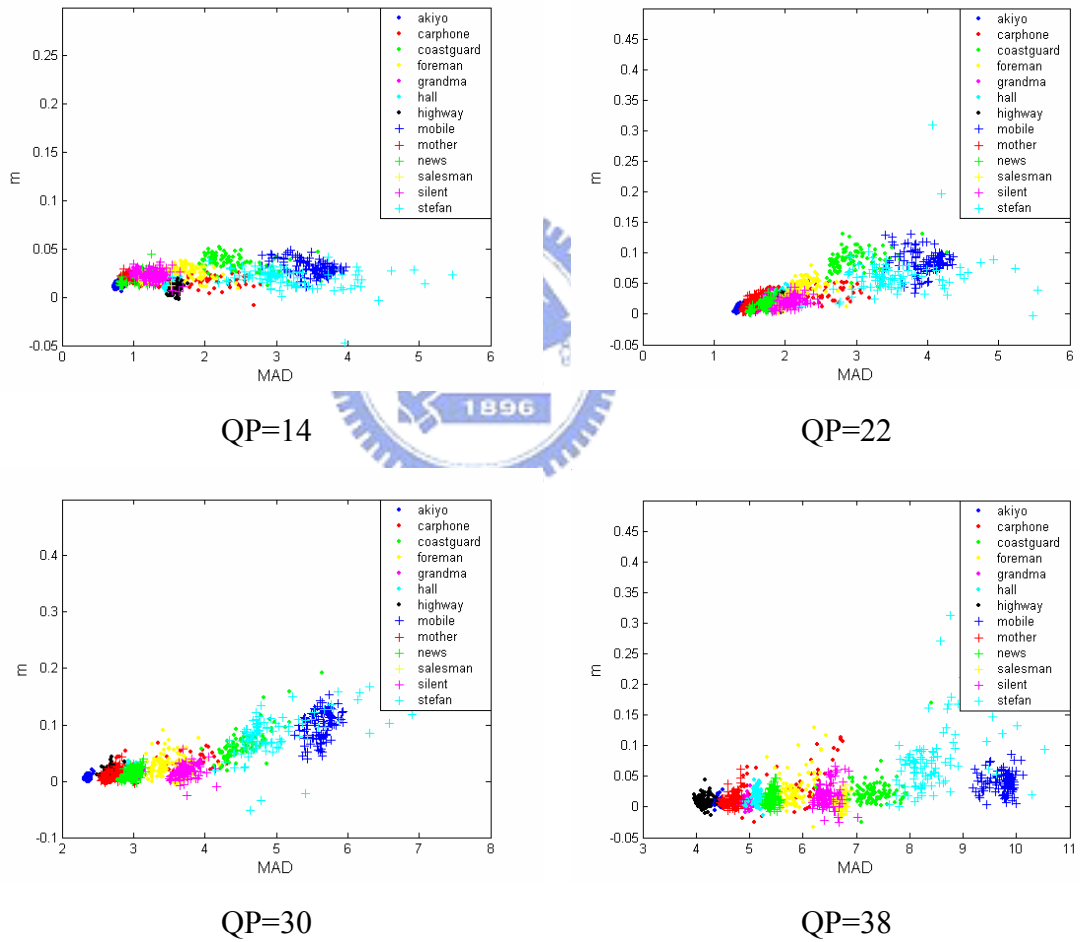


Figure 4-9 The relation between the value of m and MAD

After defining the symbol m , we observe that the value of m and MAD are related. In , we can express this relation as a linear model when QP is 14, 22, 30, 38 or other values. Hence, the MAD value of the next picture can be predicted according to this relation. Our strategy is

that after having determined the QP value of the current picture, we can predict the QP value of the next picture. This criterion can be either the minimum distortion or a stable visual quality. In the minimum distortion criterion, we can calculate the QP values of all cases and find the one which has the smallest distortion. On the other hand, in the stable visual quality criterion, we choose the one such that QP values are the most similar.

In the following experiment, we compare the performance in two aspects. One is the PSNR of each pictures, while the other is the stability of the PSNR.

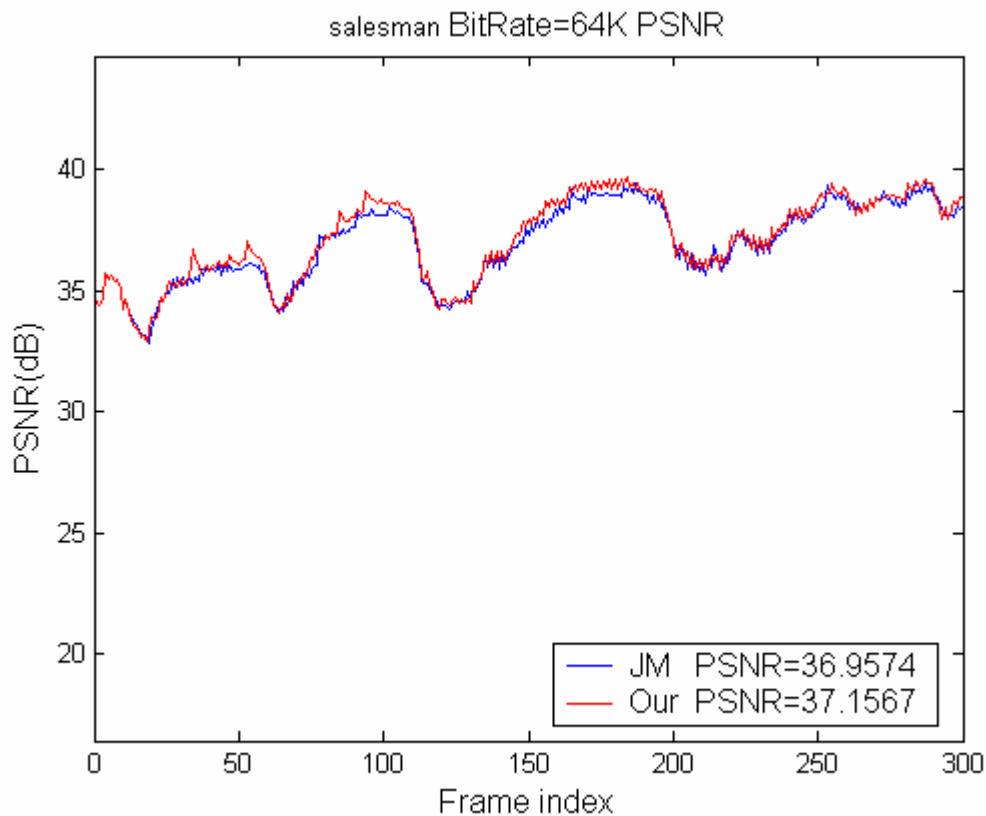


Figure 4-10 The PSNR of each pictures,. The sequence is “salesman” and the bit rate is 64K.

In Figure 4-10, the PSNR of each picture is improved. The average improvement of PSNR is about 0.5dB. The other improvements are list in the Table 4-2

Table 4-2 The PSNR of this experiment

Sequence	Bit Rate	PSNR		Improvement
		JM	Our	
Akiyo	64K	42.75	42.85	0.1
Grandma	64K	39.38	39.50	0.12
Hall	64K	38.31	38.37	0.06

In the stability of the PSNR experiment, the performance is defined to be the absolute difference between the original PSNR of each picture and the PSNR after passing through the median filter. A smaller value means the PSNR is more stable. We can find an improvement of the PSNR stability in Table 4-3. In Figure 4-11, we can see there is a little, but not obvious, improvement.

Table 4-3 The PSNR stability of the experiment

Sequence	Bit Rate	PSNR		Improvement
		JM	Our	
Hall	32K	0.1555	0.1443	0.0112
Highway	128K	0.3133	0.2672	0.0461
Mother	128K	0.3352	0.2709	0.0643
Coastguard	512K	0.3134	0.2914	0.0220

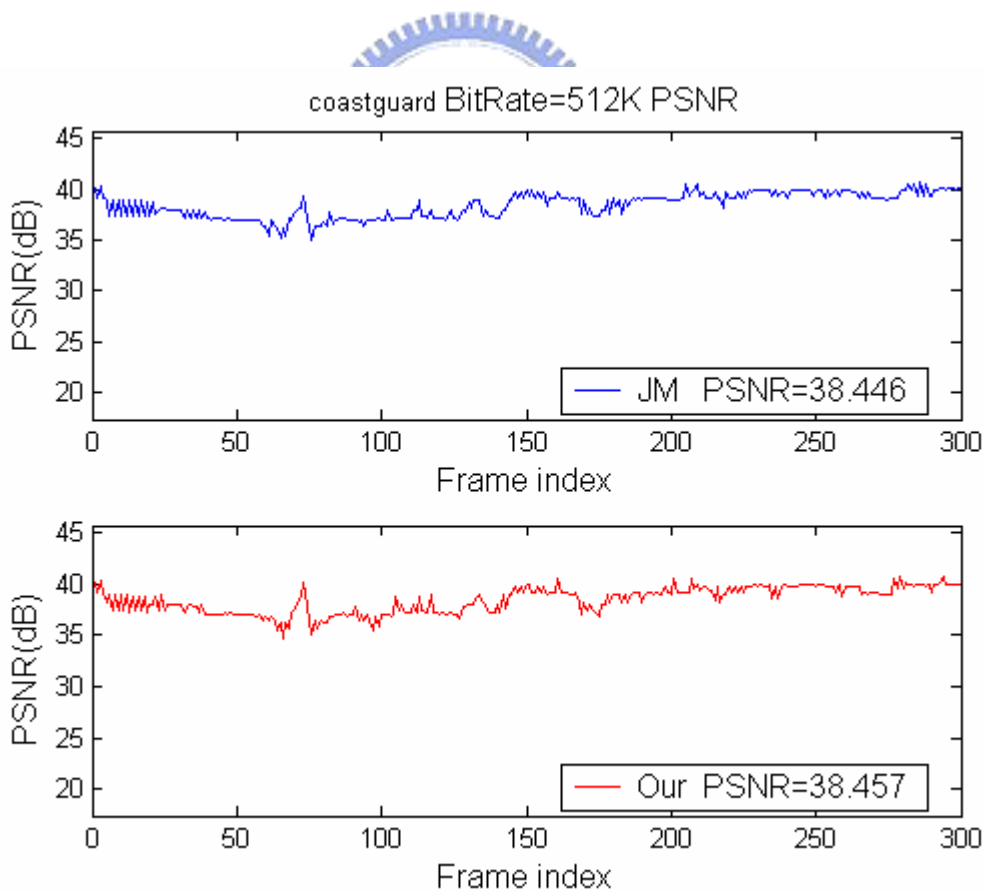


Figure 4-11 The PSNR of each picture

4.2 Macroblock Level

In the JM of H.264/AVC, if the number of basic unit in one picture is one, we need to determine the bit allocation for each basic unit. If the basic unit is a macroblock, we'll determine the bit allocation for each macroblock. In JM, the original method is defined as

$$T_{mb}(j,k) = \frac{pMAD(j,k)^2}{\sum_k pMAD(j,k)^2} \times T(j) \quad (4-5)$$

where the T_{mb} is the target bits of the macroblock, T is the target bits of the picture, j means the index of the picture, and k means the index of the encoded macroblock. The symbol $pMAD$ means the predicted MAD of a macroblock.

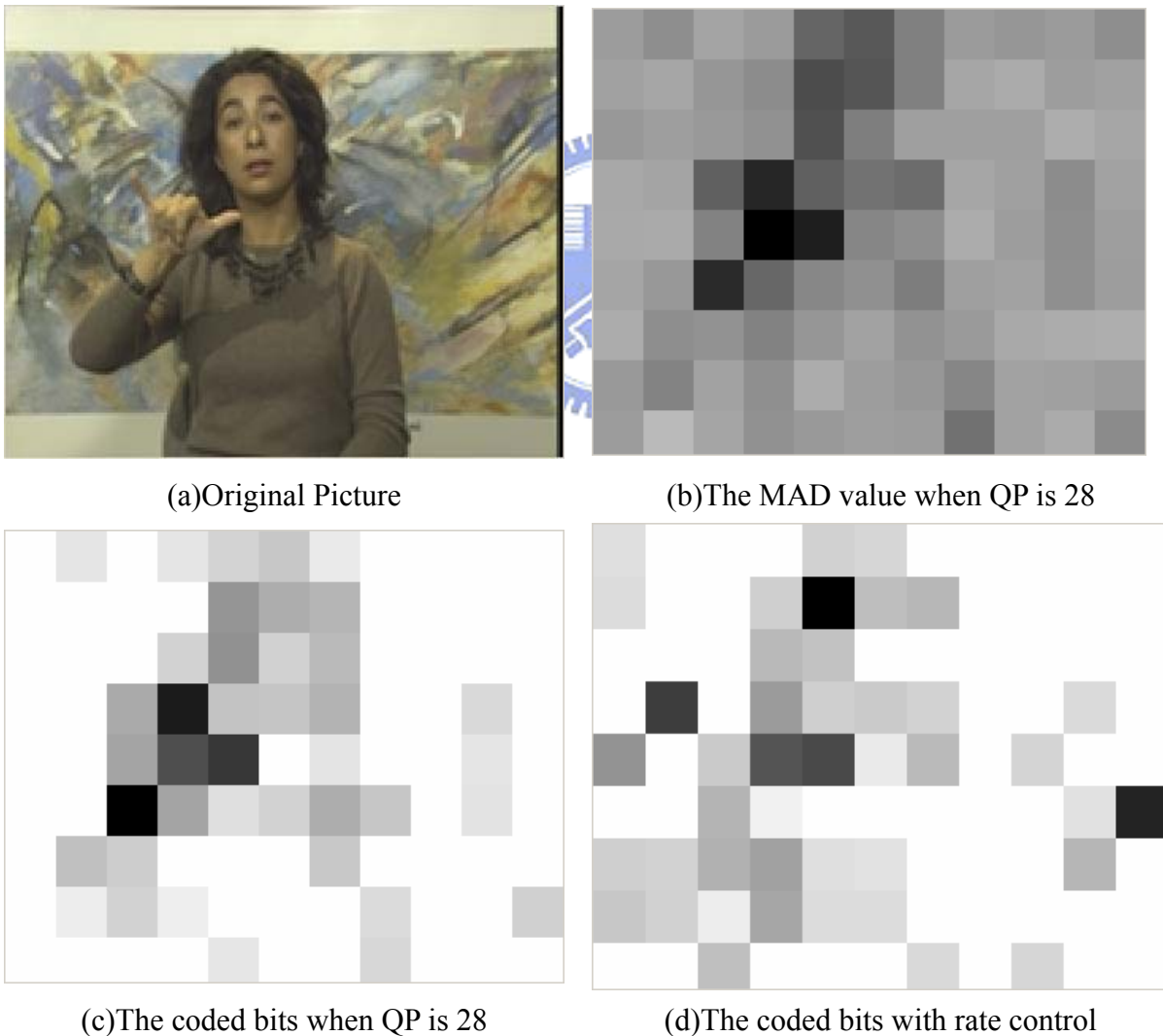


Figure 4-12 The relation between the coded bits and MAD in macroblock level

The concept of this equation is that a macroblock with a larger MAD needs more bits to be coded. In Figure 4-12, a darker macroblock means that the MAD of this macroblock is higher and the coded bits are more than a brighter macroblock. The distribution in Figure 4-12 (d) needs to be similar to the distribution in Figure 4-12 (c). In this section, we modify the bit allocation at the macroblock level to improve the coding performance.

4.2.1 MAD prediction from forward frame (add motion vector)

We have an idea that the residual data of each macroblock will move when the moving vector exists. It means the MAD of macroblock will move with motion vector. Figure 4-13 describes that the motion will cause the original MAD of this macroblock to move to other macroblocks.

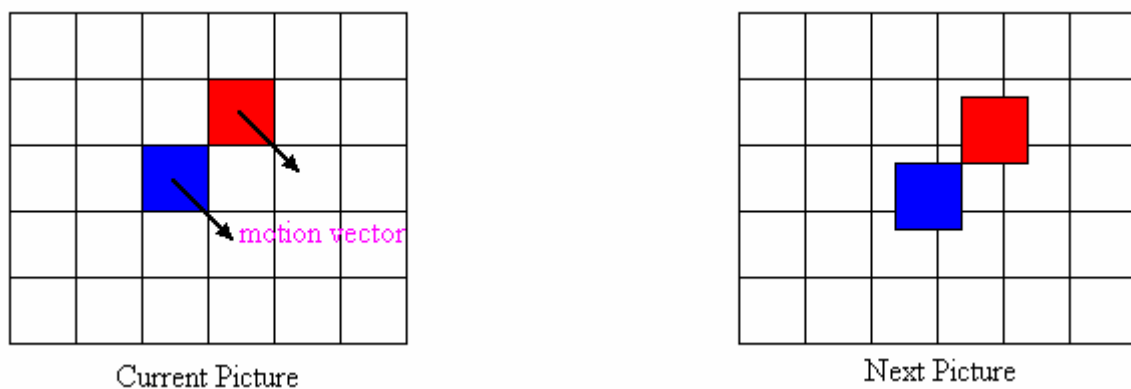


Figure 4-13 The diagram of macroblock motion

Hence, we try to modify the MAD prediction in the macroblock level. We use the motion vector of each macroblock and move the MAD of each macroblock to the new location. After moving all the macroblocks, we calculate the average MAD to represent the predicted MAD value. Like the case the Figure 4-14, the MAD of macroblock M in the next picture can be computed as:

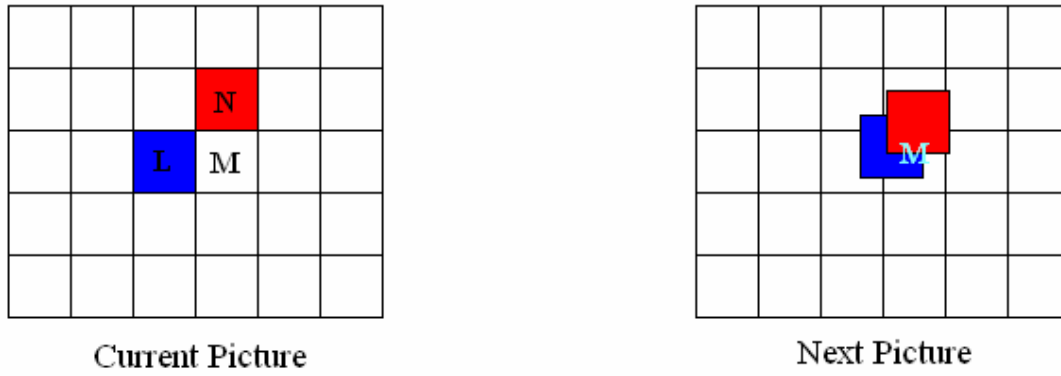


Figure 4-14 Our method to predict the new MAD value

$$pMAD_{t+1}(M) = \frac{w_N \times MAD_t(N) + w_L \times MAD_t(L) + w_M \times MAD_t(M)}{w_N + w_L + w_M} \quad (4-6)$$

where the pMAD means the predicted MAD, the w is used to represent each weighted MAD in the macroblock. Employing this method can improve the video quality in theory. This is because a more accurate MAD prediction can generate a more accurate bit allocation and avoid a mismatch in bit allocation. This can also improve the situation of buffer fullness.

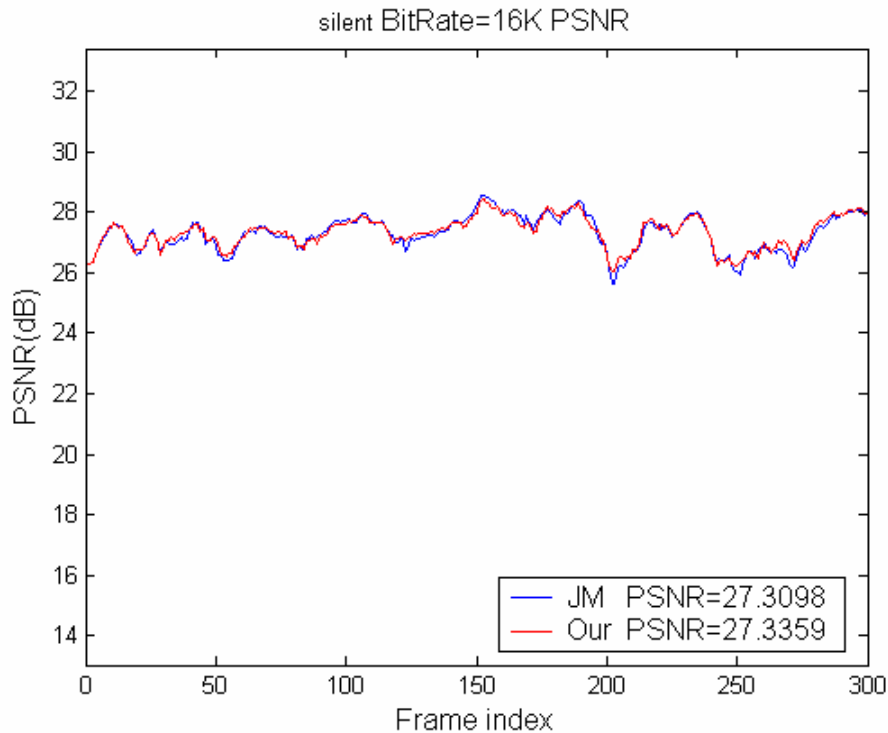
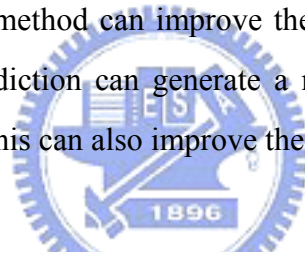


Figure 4-15 The PSNR of each picture with the “silent” sequence when the bit rate is 16K

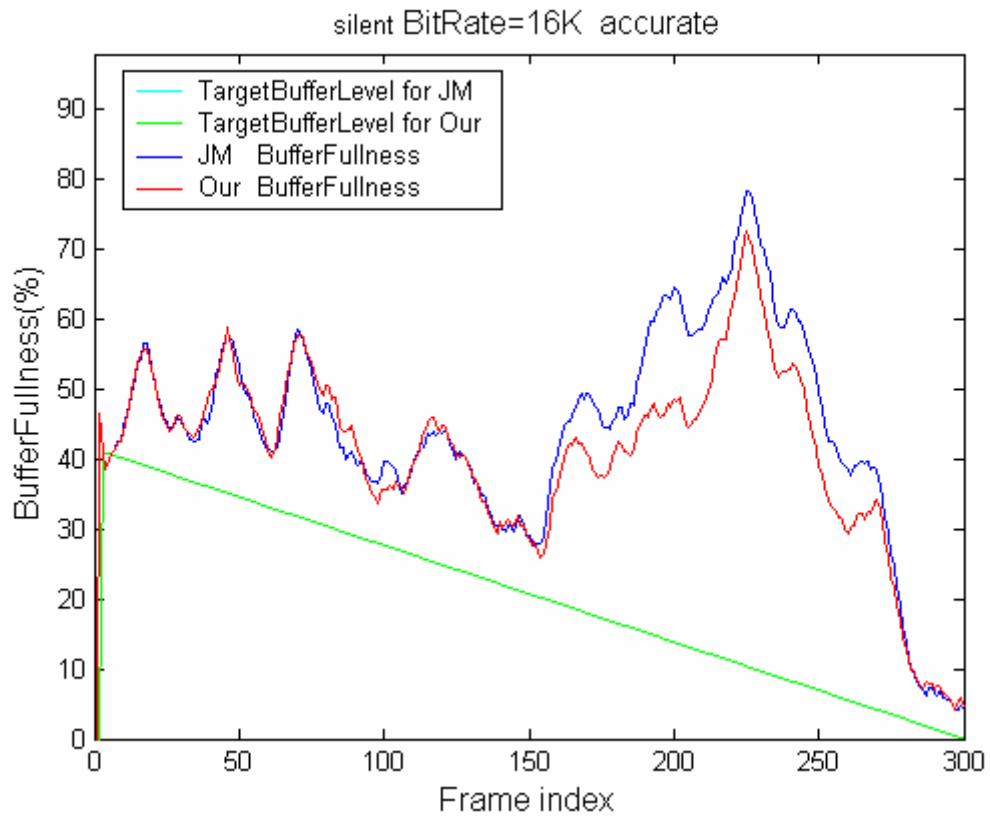


Figure 4-16 The buffer fullness with the “silent” sequence when the bit rate is 16K.

In Figure 4-15, we can find a little improvement in PSNR. There is also an improvement for accurate coded bits of picture in Figure 4-16. However, the improvement of our strategy is not very obvious.

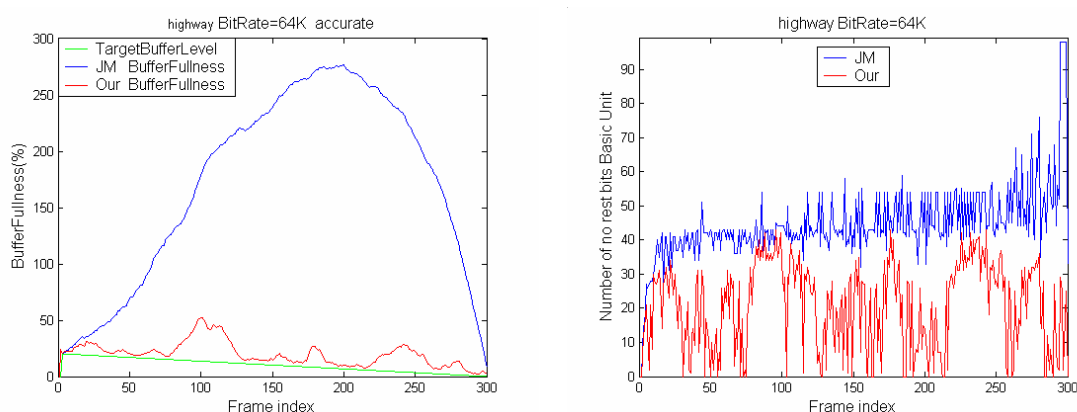
4.2.2 A solution of coding too many bits in picture level

In Section 3.2, we discussed the accuracy of the R-D model. We provide an accurate model to improve the situation. In this section, we use another approach. In a good rate control method, the coded bits of each picture must match the target bits. If the number of macroblock which is coded without the R-D model is large, it means the target bits have been used up long before we finish the encoding of all the macroblocks. Hence, we may change the target bits of each macroblock to improve this situation. The strategy is that we predict the location of these macroblocks that have no target bits left. Then, we change the target bits of these macroblocks which are before the predicted location and move the actual location backward. In Figure 4-18, we describe our strategy and formulate it as follows

$$T_{mb}(m) = \gamma \times T_{mb}(m) \quad \text{if } m < \tilde{m}$$

$$0 \leq m < \text{Total Macroblock Number} \quad (4-7)$$

where T_{mb} is the target bits for each macroblock and γ is a coefficient which is between 0 and 1. Generally a smaller γ can improve the situation more obviously. In the following, we perform some experiments and compare the results between the original approach and the proposed modification.



Buffer Fullness

The number of macroblock that is coded without using R-D model

Figure 4-17 Compare the buffer fullness and the number of macroblock that is coded without using R-D model

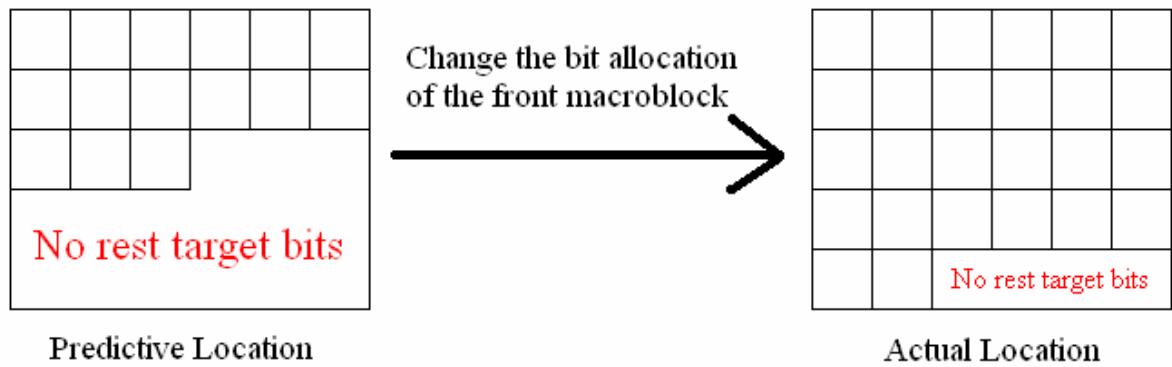


Figure 4-18 The illustration of our strategy in changing bit allocation

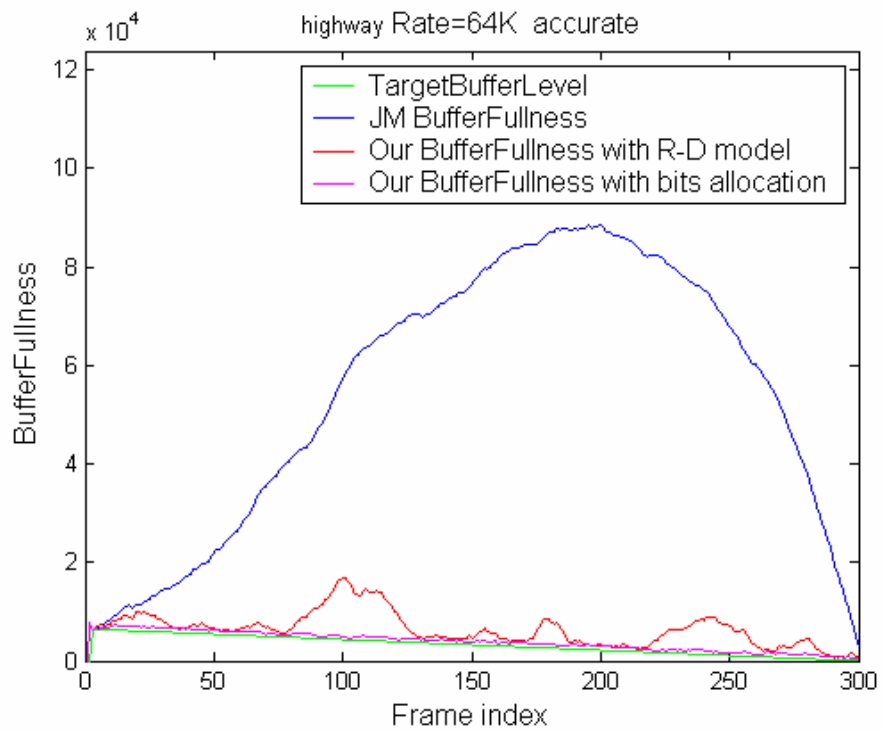


Figure 4-19 The buffer fullness



Figure 4-20 The PSNR of each picture

In Figure 4-19, the buffer fullness after the change of the bit allocation is very close to the target buffer level. This is because we can suppress the number of the macroblocks that have no remaining target bits. However, there is still a side effect. In Figure 4-20, the PSNR of each picture after the change of the bit allocation is lowered. This is because reducing the target bits of the previous macroblocks may cause the mismatch between the target bits and the complexity of each macroblock. Hence, although this modification can control the buffer fullness well, it is still not a good solution.

In Chapter 3 and 4, we proposed several approaches in some issues of rate control for H.264/AVC. Finally we use a flow chart to represent the summary of our algorithm. Figure 4-21 is our final algorithm for the rate control of H.264/AVC.

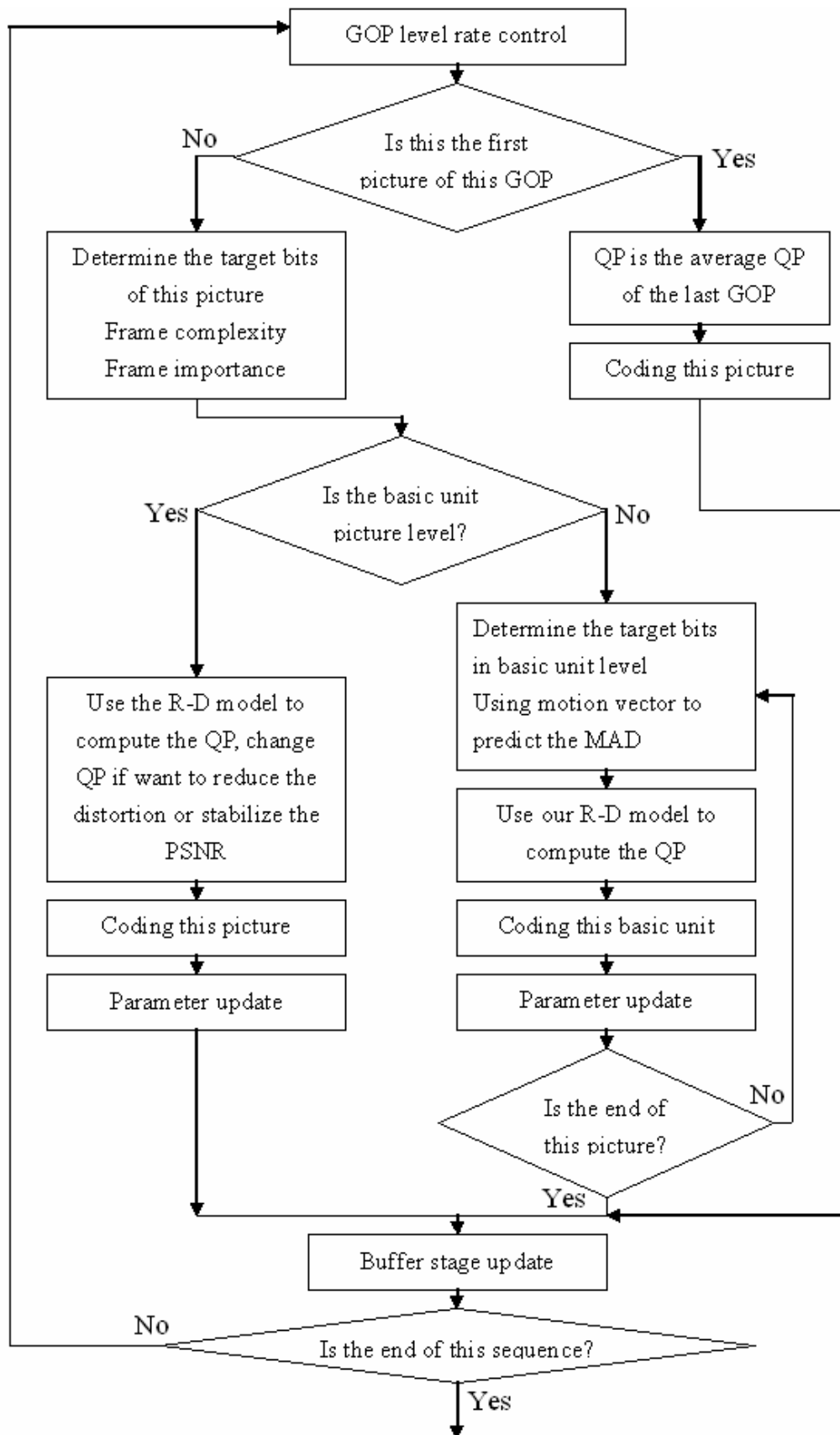


Figure 4-21 The flow chart of our algorithm

Chapter 5 Conclusion

In this thesis, we study some rate control issues of H.264/AVC. Here, we conclude our accomplishments as below:

1. We propose a new rate-distortion model for the baseline profile of H.264/AVC. Using the R-D model can improve the accuracy of rate control. The improvement can also get good visual quality, especially in low bitrates.
2. The relation between the header bit and the MAD value has been found. We use this relation to predict the number of header bits, which can be used in our R-D model.
3. In the picture level, we can adjust the quantization parameter according to the frame complexity or importance. The number of coded bits can affect the coded bits of the next picture. We also use the relation to adjust the quantization parameter in our method.
4. In the basic unit level, we use an instinctive method to predict the MAD value. Utilizing the bit allocation of basic unit level can improve the accurate target bits of picture level.

Bibliography

- [1] “Draft ITU-T recommendation and final draft international standard of joint video specification (ITU-T Rec. H.264/ISO/IEC 14496-10 AVC,” in Joint Video Team (JVT) of ISO/IEC MPEG and ITU-T VCEG, JVT-G050, 2003
- [2] King N. Ngan, Thomas Meier, and Zhenzhong Chen, “Improved Single-Video-Object Rate Control for MPEG-4,” *IEEE Trans. on Circuits. Syst. Video Technol.*, vol. 13, No. 7, pp.385-393, May 2003
- [3] Thomas Wiegand, Gary J. Sullivan, Gisle Bjontegaard, and Ajay Luthra, “Overview of the H.264/AVC Video Coding Standard,” *IEEE Trans. on Circuits. Syst. Video Technol.*, vol.13, No.7, pp.560-576, July 2003
- [4] M. Karczewicz and R. Kurçeren, “The SP and SI frames design for H.264/AVC,” *IEEE Trans. on Circuits. Syst. Video Technol.*, vol. 13, No. 7, pp. 637–644, July 2003.
- [5] Iain E. G. Richardson, “H.264 and MPEG-4 Video Compression: Video Coding for Next Generation Multimedia.”
- [6] Detlev Marpe, Stephen Gordon, and Thomas Wiegand, “H.264/MPEG4-AVC Fidelity Range Extensions: Tools, Profiles, Performance, and Application Areas,” *Proc. ICIP 2005*, vol.1, pp.I-593-6, Genova, Italy, September 11-14, 2005.
- [7] ISO/IEC JTC1/SC29/WG11, *Test Model 5*, 1993.
- [8] ITU-T/SG15, *Video Codec Test Model*, TMN8, Portland, June 1997.
- [9] MPEG-4 Video Verification Model V18.0, Coding of Moving Pictures and Audio N3908, ISO/IEC, JTC1/SC29/WG11, Jan. 2001.

- [10] H. J. Lee and T. H. Chiang and Y.-Q. Zhang, "Scalable rate control for MPEG-4 video," *IEEE Trans. on Circuits. Syst. Video Technol.*, vol.10, No.6, pp. 878-894, Sept. 2000.
- [11] Z. Li et al., "Adaptive Basic Unit Layer Rate Control for JVT," *JVT-G012, 7th Meeting*: Pattaya, Thailand, March 2003.
- [12] Siwei Ma, Wen Gao, Feng Wu and Yan Lu, "Rate Control for JVT Video Coding Scheme with HRD Considerations," *IEEE International Conference on Image Processing*, vol. 3, pp.793-796, Sept. 2003.
- [13] Thung-Hiung Tsai and Jin-Jang Leou, "A Rate Control scheme for H.264 Video Transmission," *IEEE International Conference on Multimedia and Expo (ICME)*, vol.2, pp.1359-1368, June 2004.
- [14] Satoshi Miyaji, Yasuhiro Takishima and Yoshinori Hatori, "A Novel Rate Control Method for H.264 Video Coding," *IEEE International Conference on Image Processing*, vol. 2, pp.II-309-12, 11-14 Sept. 2005
- [15] Siwei Ma and Wen Gao, "Rate-Distortion Analysis for H.264/AVC Video Coding and its Application to Rate Control." *IEEE Trans. on Circuits and Syst. Video Technol.*, vol.15, No.12, pp.1533-1544, Dec. 2005
- [16] Mingiang Jiang and Nam Lin, "On Enhancing H.264/AVC Video Rate Control by PSNR-based Frame Complexity Estimation," *IEEE Transactions on Consumer Electronics*, vol.51, No.1, pp.281-286, Feb. 2005



簡歷

蔣宗翰，民國71年2月5日出生於台北市。86年到89年就讀於國立高雄師範大學附屬高級中學，之後進入國立成功大學電機工程學系。大學畢業後進入國立交通大學電子研究所修讀碩士學位至今。碩士論文題目為「H.264/AVC之碼率控制技術研究」。

聯絡地址：高雄市三民區黃興路437巷5號5樓

電子郵件：jkc.ee93g@nctu.edu.tw