REGION OF INTEREST (ROI) BASED RATE CONTROL FOR H.263 COMPATIBLE VIDEO CONFERECING

by

LIN TONG

Presented to the Faculty of the Graduate School of

The University of Texas at Arlington in Partial Fulfillment

of the Requirements

for the Degree of

DOCTOR OF PHILOSOPHY

THE UNIVERSITY OF TEXAS AT ARLINGTON

December 2005

ACKNOWLEDGEMENTS

First of all, I would like to thank my research advisor Prof. K.R.Rao for teaching me how to conduct serious research and how to clearly and effectively present my work. He was always a source of encouragement for my Ph.D. studies. I also would like to thank Prof. Qilian Liang, Prof. Soontorn Oriantara, Prof. Rao, Prof. Manry and Prof. Devarajan, from whom I got invaluable training in the courses they taught at UTA.

I enjoyed technical discussions with many group members in MP lab and MSP lab and Prof. Zhou Wang. I would like to thank the group members in MP lab for their consistent support.

My husband, Zhipeng Zhu, gave me peaceful life to go through all the troubles I met. I am greatly indebted to him for all he has done for me.

Finally, I would like to thank my family for their unconditional support.

October 30, 2005

ABSTRACT

REGION OF INTEREST (ROI) BASED RATE CONTROL FOR H.263 COMPATIBLE VIDEO CONFERECING

Publication No. _____

Lin Tong, PhD.

The University of Texas at Arlington, 2005

Supervising Professor: K.R. Rao

This dissertation presents a region of interest (ROI) based H.263 [10] [13] compatible video codec, which integrates the idea of object-based coding from MPEG-4 Visual [9] [15] into the traditional block-based H.263 codec. A face detection and tracking scheme with very low complexity is proposed to segment human face region from video conferencing sequences in real-time. For intra frames, the proposed detection method is a hybrid skin color and mosaic-rule based face detection and for

inter frames the face tracking method uses motion vectors only. With the segmentation information, the ROI based codec and its associated rate control schemes are designed.

By analyzing quadratic rate models in frame layer, VOP layer and macroblock layer extracted from the test data, a quadratic rate model at macroblock layer with a modified physical meaning is proposed to improve the model accuracy. The basic idea is to use a group of un-coded macroblocks in the current VOP instead of individual macroblocks to achieve the rate model and update model parameters. Based on this proposed rate model, some new features are designed to use both average statistics of a VOP and individual statistics of MB at the same time in order to achieve more efficient rate control performance.

For CBR video, a joint VOP layer and MB layer rate control algorithm is proposed. The performance is compared with conventional TMN8 rate control as a baseline, and object based VM8 rate control as well. The comparisons show that the proposed algorithm can achieve more accurate rate control and better PSNR for ROI based on several video sequences.

The proposed rate control algorithm is applied for variable bit rate (VBR) video case also, together with a frame layer rate control scheme. The efficiency of the proposed rate control is proved by simulation.

TMN8 [6] is adopted as the platform, and the Modified Quantization Mode in Annex T of H.263 [69] is adopted to achieve flexibility in assigning quantization parameters among different macroblocks.

TABLE OF CONTENTS

LIST OF ILLUSTRATIONS

LIST OF TABLES

Table Page

CHAPTER 1

INTRODUCTION

1.1 Introduction to Rate Control

Multimedia applications over communication networks are highly popular with the increasing bandwidth and the well established compression techniques. Low bit rate video coding is important for video communication over wireless data networks and ISDN channels. In the upcoming third generation (3G) wireless systems, the outdoor data rates range from 144-384 kbps [1]. The ISDN channel can provide data rate as the multiples of basic channels of 64 kbps each, with the multiplying factor p ranging from 1-24, corresponding to a bandwidth of 64-1536 kbps [1]. In the low bit rate applications, for example, at 128 kbps ($p = 2$), only very low-quality video (QCIF at 10fps) can be achieved, and at 384 kbps ($p = 6$), better video quality (CIF at 15-30fps) can be achieved. Efficient video coding at low bit rate is a very challenging and is an important task for these applications.

Generally, a network-based multimedia communication system can be viewed as a four-layer system: application layer, compression layer, transport layer and transmission layer, as shown in Fig. 1.1 [1].

The most challenging part to design a multimedia communication system is how to make good use of the limited resources (bandwidth, buffer etc.) to achieve an optimal

video quality. There are two mechanisms that have been studied on this issue: rate shaping and rate control [1]. Rate shaping is a transport layer technique, it adapts the pre-compressed video bit stream to a target rate constraint. The typical rate shapers are: codec filters, frame-dropping filters, layer-dropping filters, frequency filters and re-quantization filters. Rate control is a compression layer technique. It determines the sending rate of video traffic based on the estimated available bandwidth in the network [1]. Different rate control schemes have been proposed for different video coding standards such as H.26x and MPEGx [9] [10] for various applications [3] [4]. Since these standards only specify the bit-stream syntax and the decoding method, rate control schemes on the encoder side are left open to designers.

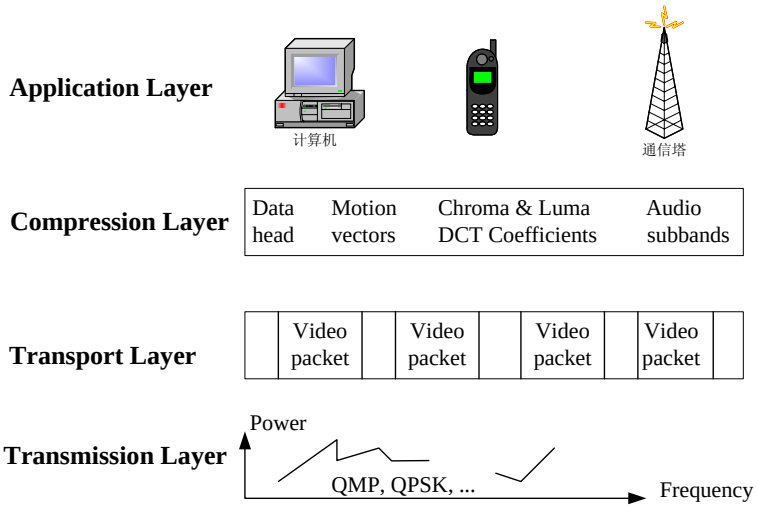


Figure 1.1. Layer structure of a network-based multimedia system [1]

1.2 Outline of Dissertation

In this dissertation, a region of interest (ROI) based rate control scheme is proposed for block-based video coding standards [9] [10]. The proposed algorithm is simulated on H.263 TMN8 platform [6] [13]. A new ROI detection and tracking method is proposed as the preprocessing stage of the ROI based coding system. Rate control model is studied at frame layer, video object plane (VOP) layer and macroblock layer. Based on the analysis of these rate models, a novel rate control scheme is proposed. The proposed rate control scheme is tested for both constant bit rate (CBR) video and variable bit rate (VBR) video. The simulation results are discussed in detail.

The dissertation is organized as follows. Chapter 2 provides a brief review of the standard based coding such as H.26x and MPEGx [9] [10], with emphasis on their rate control algorithms. Then the two commonly used optimization tools in rate control problems are introduced briefly. Chapter 3 presents the motivation for region of interest (ROI) based rate control for block based coding standards [9] [10], and the existing solutions are reviewed. Chapter 4 proposes a novel human face detection and tracking method for real time video conferencing applications. In chapter 5, rate control models are developed for frame layer, VOP layer, and macroblock layer. The advantages and disadvantages of these three rate control models are discussed. Chapter 6 presents a novel joint VOP layer and MB layer rate control algorithm for CBR video. The rate control algorithm proposed in chapter 6 is employed in VBR application in chapter 7. Chapter 8 summarizes the proposed research, and future directions are discussed.

CHAPTER 2

BACKGROUND OVERVIEW

2.1 Digital Video Coding Standards Overview

Before getting into the rate control part, the most popular digital video coding standards such as H.26x and MPEGx [9] [10] are first introduced. Most standards-based coding performs hybrid DCT / interframe coding (DCT: discrete cosine transform). Different features are designed aimed at various applications in different standards, which will be emphasized in this section.

2.1.1 H.261/H.263

In 1990, H.261 was published as the first widely-used international standard for video coding at low bit rate of $p \times 64$ kbps, where p is an integer between1 and 30. Its typical applications are videophone and video conference over ISDN circuit-switched networks [10][11]. H.261 is a block-based hybrid coder with motion compensation [12]. It applies 8x8 DCT for each block to reduce spatial redundancy, a differential pluse code modulation (DPCM) loop to exploit temporal redundancy, and unidirectional integer pixel forward motion compensation for macroblocks to improve the performance of the DPCM loop (Fig. 2.1). An optional loop filter is used to low-pass filter the motion-compensated prediction data so that blocky artifacts of the predicted picture can be reduced. H.261 uses two quantizers for DCT coefficients. For DC

coefficient in intra mode, a uniform quantizer with stepsize 8 is used, and for AC coefficients in intra mode and in inter-mode a nearly uniform midtread quantizer with the stepsize (2-62) is used. Except for the dead zone, the quantizer is uniform. The quantized DCT coefficients are scanned using a zigzag scan and converted into symbols. Each symbol is coded by a variable length code (VLC) which is derived from statistics of test sequences.

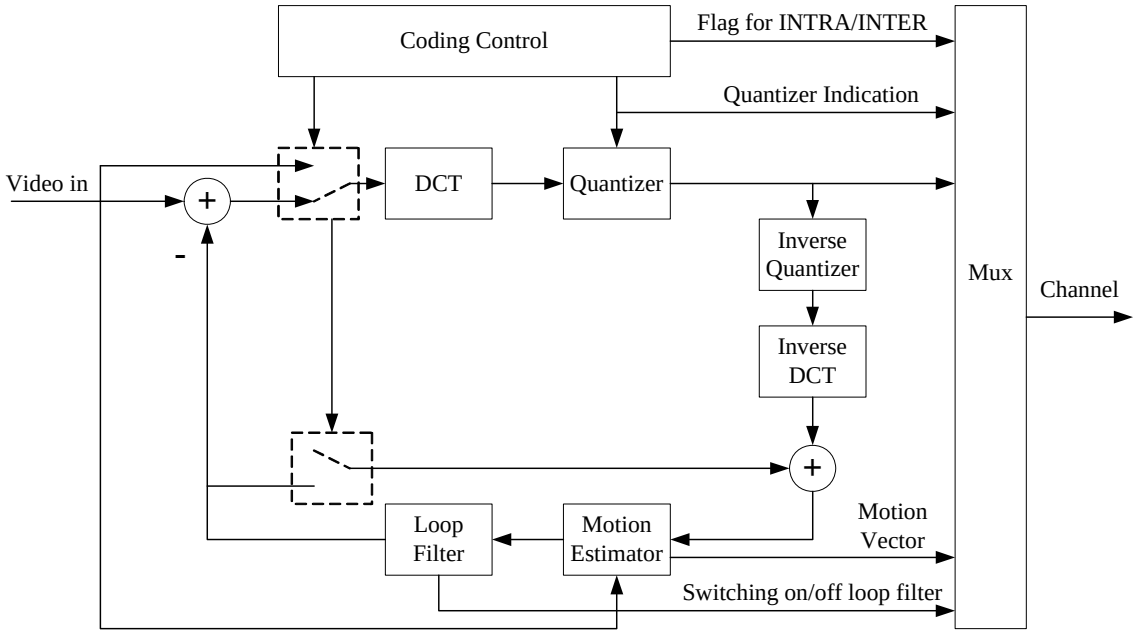


Figure 2.1. Block diagram of H.261 encoder [60]

H.263 [10] [11] [12] was first developed in 1995 in order to improve the compression performance of H.261. It supports basic video quality at bit rate of below 30kbps, and it is designed to operate over a wide range of circuit- and packet- switched networks [10]. In 1997, an extension of H.263was incorporated into the standard, which is called H.263+. In 2000, more optional modes have been approved for the third phase of H.263 development, called H.263++ [1].

The H.263 standard is based on the framework of H.261 [10] [14] and it distinguishes H.261 standards from the following major features. First, the motion compensation is based on half-pixel precision motion vectors, it improves the prediction capability of motion compensation when fine spatial resolution of motion modeling is needed. Second, it improves the 2D VLC in H.261 into 3D VLC for higher coding efficiency. Third, the overhead at the group of block level is reduced. Fourth, H.263 supports five standardized picture formats: 16CIF (common intermediate format), 4CIF, CIF, QCIF (quarter CIF), sub-QCIF, more than what H.261 supports. All these five picture formats are in the popular 4:2:0 sampling format, each of the chrominance components Cb and Cr has half of the horizontal and vertical resolution of the luminance component Y, as illustrated in Fig. 2.2.

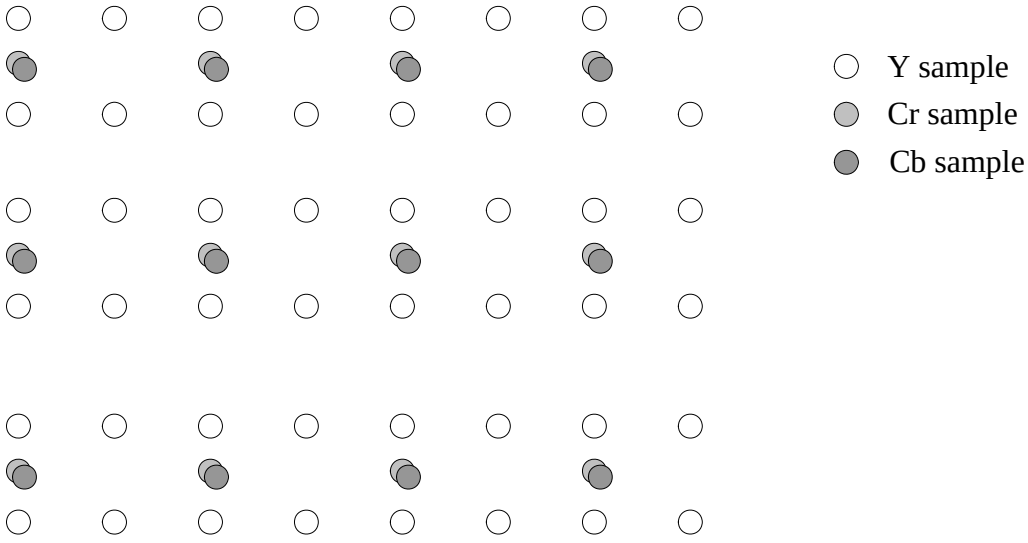


Figure 2.2 4:2:0 sampling pattern

In addition to the core algorithm improvements, H.263 includes four negotiable advanced coding modes: unrestricted motion vectors, advanced prediction, PB frames,

and syntax-based arithmetic coding. Moreover, H.263++ added a list of new optional features to further improve the coding efficiency. The details of these modes can be found in the standard [6][13].

2.1.2 MPEG-1/MPEG-2

MPEG-1 is the first MPEG standard, it was designed for progressively scanned video storage and playback on compact disks, and the target was to produce near video home system (VHS) quality video at a bit rate about 1.2 Mbps. MPEG-1 was finalized in 1993 [1]. There are many similarities between MPEG-1 and H.261, as shown in Figs. 2.1 and Fig 2.3. Comparing with the H.261 standard published in 1990, the major differences are listed below [2][10].

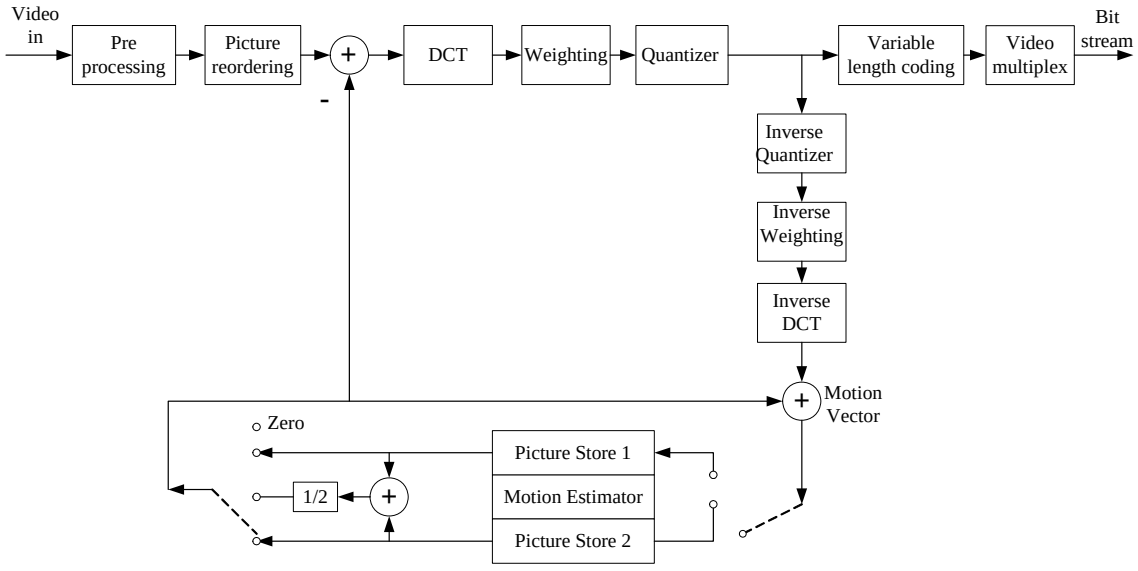


Figure 2.3 Block diagram of MPEG-1 encoder

First, MPEG-1 motion compensation is based on half-pixel accuracy motion vectors. The motion vector range is extended from ± 16 pixels to ± 64 pixels. There is

no loop filter in MPEG-1. Second, MPEG-1 uses I, P and B frames (Fig.2.4). The introduction of B frame coding requires a more complex motion compensation unit. Each macroblock of a B frame has two motion vectors, one is estimated with respect to the preceding I or P frame, the other one is estimated from the next I or P frame. Third, a weight matrix is used to adapt the quatization of DCT coefficients for I to the human visual system. Fourth, the DC coefficient of an I block may be predicted from the DCT coefficient of its neighbor to the left. Fifth, to accommodate random access, a video sequence is partitioned into groups of pictures (GOP). A GOP has to start with an I frame in order to facilitate the periodic synchronization within a sequence to reduce the damages from the transmission errors.

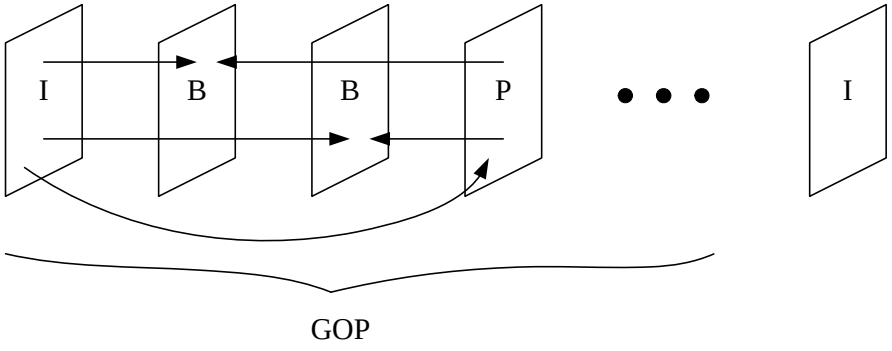
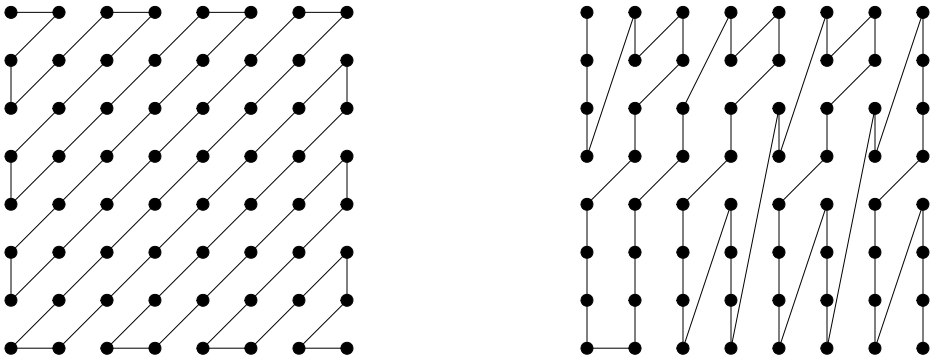


Figure 2.4 A group of pictures (GOP) structure [10]

MPEG-1 video does not provide sufficiently better quality than VHS type video, so the market of MPEG-1 video was not very successful. Following on from MPEG-1 standard, the MPEG-2 standard was developted to support digital broadcasting of compressed television. The target of MPEG-2 was to provide TV quality video at data rates of 4-8 Mbps and high quality video at 10-12 Mbps for interlaced broadcasting video, while enabling all MPEG-1 functions [12]. There are many similarities between

MPEG-1 and MPEG-2. The major differences between these two standards are as follows [10][11].

First, comparing with H.261, H.263 and MPEG-1, chrominance samples in 4:2:0 format of MPEG-2 are horizontally shifted by half pixel. Second, MPEG-2 supports interlaced video coding. So, additional scan patterns for DCT coefficients (Fig. 2.5.) and motion compensation with block size of 16x8 are supported, in order to code the interlaced blocks that have more correlation in horizontal than in vertical direction. Third, to provide high quality video, DC coefficient is 10 bits quantized instead of 8 bits quantized in MPEG-1. Nonuniform quantization and improved VLC tables have been designed. Fourth, the major feature of MPEG-2 is that it supports various modes of scalability, like spatial scalability, temporal scalability and signal to noise ratio (SNR) scalability. The fifth difference is that MPEG-2 operates at much higher bit rates. MPEG-2 has defined many profiles and levels to flexibly select different tools for various applications.



(a) (b)

Figure 2.5 Zigzag scan pattern (a) Scan pattern in H.261, H.263 and MPEG-1 [10] (b) Alternate scan pattern in MPEG-2 [10]

2.1.3 MPEG-4

MPEG-4 was developed by the MPEG committee in 1998, and an amendment version was released in late 2001. It supports a wide range of multimedia applications ranging from digital television, streaming video to mobile multimedia and games [9] [65] [66].

MPEG-4 Visual consists of a ‘core’ video codec together with many additional features. The key features that distinguish MPEG-4 from the previous coding standards are: First, it supports video object based coding. Second, it supports effective transmission over practical networks. Third, it supports still texture coding. Fourth, it supports animated visual objects coding, like 2D or 3D polygonal meshes and animated people. Fifth, it supports coding of specialist applications such as studio quality video. Because not all the coding tools are appropriate for a particular type of application, the standard groups recommended coding tools together forming a series of profiles for different type of applications [1][15].

To code rectangular video frames, there are three profiles. The Simple profile (SP) includes a minimum set of tools for low complexity applications, the Advanced Simple profile (ASP) provides better compression efficiency at the price of increased complexity, and the Advanced Real-Time Simple profile (ARTSP) includes the tools for error resilient transmission with low delay [15]. These three profiles are the most popular profiles of MPEG-4 at the present time. The coding tools in these profiles are based on the H.263 standard, and coding efficiency has been designed to out-perform MPEG-1 and MPEG-2 [9].

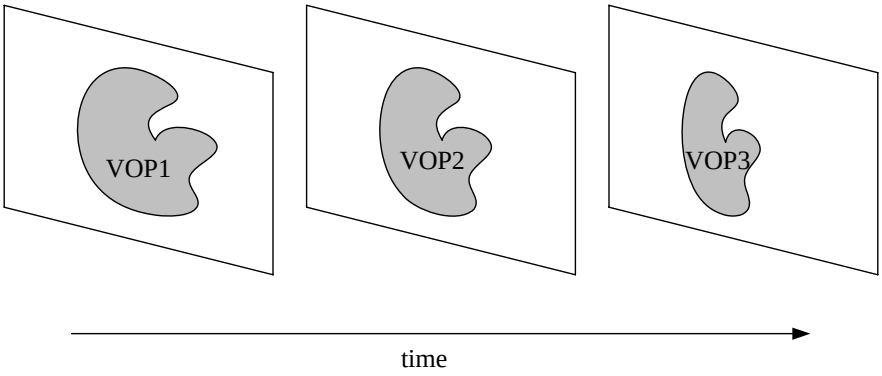


Figure 2.6. VOPs with arbitrary shape [65]

The major contribution of MPEG-4 Visual is it supports viewing a video sequence as many video objects other than just rectangular video frames. To code the arbitrary shaped objects MPEG-4 Visual provides four profiles: the Core profile, the Main profile, and Advanced Coding Efficiency profile and the N-bit profile. These profiles share the following new concepts for object-based coding. Video object is the central concept, it has three dimensions (2-D in spatial and 1-D in time). The time sample of a video object (VO) is called a video object plane (VOP), as illustrated in Fig.2.6. VOPs can be encoded independently of each other (I-VOP) or dependent on each other (P-VOP or B-VOP). A conventional video frame can be represented by a VOP with a rectangular shape. Each VOP is coded separately, containing shape information, motion parameters and texture data. The shape of VOP is defined by Alpha Blocks, which are based on macroblock of size (16 x 16). Shape information is defined using the concept of transparency. A transparent pixel means it does not belong to the current VOP, while an opaque pixel does belong to the VOP and a semi-transparent

pixel is partly inside of the VOP. The shape information is defined as binary in the Core profile and in grey scale in the Advanced profile [15].

The Core profile in MPEG-4 is based on the SP tools adding object-based coding features like binary shape coding, motion compensated coding of arbitrary shaped VOPs and texture coding in boundary MB [9].

Binary shape coding is used to encode a binary alpha mask (BAB) to indicate which pixels are inside VOP and which are not inside. At the MB by MB basis, a code *bab-type* is transmitted indicating whether a MB is transparent or opaque or a boundary MB. For a boundary MB, the BAB is coded using context based binary arithmetic encoding (BAE). Each BAB pixel value x is coded by three steps. First, a context which defines a region of n neighboring pixels that have previously been coded is calculated for the current pixel. For intra coded BABs, spatial neighbors are used, while for inter coded BABs, spatial and temporal neighbors are used. The n values of each BAB pixel in the context form a n-bit word. There are 2^n possible contexts. The probability that x is 0 given a particular context is stored in the encoder and decoder, by looking up in the probability table, the relevant entry can be achieved and encoded with an arithmetic encoder.

In order to code VOP using block-based motion compensation, there is a special case that needs to be dealt with. I.e., some of the opaque pixels in the current MB are motion compensated from transparent pixels in the reference VOP. Since naturally there are no values defined for the transparent pixels, it is necessary to fill transparent pixel positions in each boundary MB. By extrapolating horizontally and vertically from

opaque pixels, transparent pixels in boundary MB can be defined. Moreover, fully transparent MBs need to be filled with padded pixel values from neighboring boundary MB because they may also be inside a motion-compensated reference region. So, there are two steps of padding in MPEG-4 Visual, first all boundary MBs are fully padded, and then transparent MBs are padded [15].

Each 8x8 texture block within a boundary MB can be coded by shape adaptive DCT (SA-DCT) [9], quantization, run-level coding and entropy coding.

In other more advanced object based coding profiles, there are a number of additional tools to increase the coding efficiency. The grey shape coding and static sprite coding in the Main profile, and quarter pixel motion compensation and global motion coding in the Advanced Coding Efficiency profile are few examples. The details of these coding tools can be found in [1][9][15].

2.2 Standard Rate Control Algorithms

Neither MPEGx nor H.26x standards [9] [10] does not specify how to perform rate control. However, rate control takes an important role in video coding to optimize the video quality according to the network conditions. To provide testing and to implement simulations using a common set of encoder routines, both MPEGx and H.26x created a series of test models. Also rate control algorithms are specified in the test models. In the following, the most well-known rate control algorithms adopted by these international standard test models will be reviewed.

2.2.1 TM5 Rate Control Algorithm

The rate control algorithm [17] outlined in MPEG-2 Test Model 5 (TM5) is performed in two major steps. In the first step, the target bit allocation for each frame inside a GOP is achieved by a frame level bit allocation scheme. In the second step, the quantization parameter for each macroblock is determined by a virtual buffer status and the spatial activity of the macroblocks.

2.2.1.1 Target Bit Allocation

The goal for the target bit allocation is to achieve the average target bit rate at a GOP level. To allocate bits for each frame depends on the frame type: I frame, P frame or B frame. For each frame type, there is a complexity model to estimate the number of bits needed to encode a frame of a given type using a specific quantization parameter. This model is

$$X_I = Sb_I \times Q_I, X_P = Sb_P \times Q_P, X_B = Sb_B \times Q_B \quad (2.1)$$

where X_I, X_P and X_B represent the complexities of I frame, P frame and B frame respectively. Similarly Sb_I, Sb_P, Sb_B are the number of bits used for each frame type. Q_I, Q_P, Q_B are the quatization parameters used for each frame type. The complexity model is updated after encoding each frame, based on the average quantization parameter and the number of bits used for that frame.

A target number of bits is then assigned to the next frame with the same type according to (2.2), (2.3) and (2.4)

$$T_I = \max\left\{\frac{R}{1+\frac{N_P X_P}{X_I K_P}+\frac{N_B X_B}{X_I K_B}},\quad \frac{bit\_rate}{8\times picture\_rate}\right\} \tag{2.2}$$

$$T_P = \max\left\{\frac{R}{N_P+\frac{N_P K_P X_B}{K_B X_P}},\quad \frac{bit\_rate}{8\times picture\_rate}\right\} \tag{2.3}$$

$$T_B = \max\left\{\frac{R}{N_B+\frac{N_P K_B X_P}{K_P X_B}},\quad \frac{bit\_rate}{8\times picture\_rate}\right\} \tag{2.4}$$

where K_P and K_B are constant parameters with default values 1.0 and 1.4 respetively. R is the remaining number of bits assigned to the current GOP, T_I, T_P and T_B are target number of bits for the corresponding frame type, N_P and N_B are the number of P frames and B frames remaining in the current GOP respectively [17].

2.2.1.2 Quantization Step Size Determination

The reference quantization step size is calculated at a macroblock layer based on the status of buffer fullness. The buffer fullness is predicted by

$$dd_j^m = dd_0^m + B_{j-1} - \frac{T_{I,P,B}\times(j-1)}{MB_{number}} \tag{2.5}$$

where $m = I, P, B$, and dd_0^m is the initial buffer fullness for the corresponding frame type. B_{j-1} is the total number of bits up to macroblock $j-1$. MB_{number} is the total

number of macroblocks. Then the quantization step size Q_j for macroblock j is adjusted by the buffer fullness dd_j from the 31 QP observation points as

$$Q_j = \frac{dd_j \times 31 \times picture\_rate}{2 \times bit\_rate} \tag{2.6}$$

The overall TM5 rate control algorithm is based on the assumption that the distortion increases linearly with the quantization parameter and the bit rate is inversely proportional to the distortion. Because the second assumption is based on an extremely simplified RD model, the performance of TM5 rate control algorithm is not accurate and robust [17].

2.2.2 TMN8 Rate Control Algorithm

The rate control scheme in H.263 Test Model Near-term Version 8 (TMN8) includes a frame skipping scheme, a frame layer rate control scheme and a macroblock layer rate control scheme [4][6].

2.2.2.1 Frame Skipping

The frame skipping scheme in TMN8 includes the following steps [6]:

1) Calculate the number of bits in the encoder buffer, while using the actual bit-count used for $(k-1)$ th frame to predict the k th frame.

$$Buf\_f = \max(Buf\_f_{prev} + R_{k-1} - \frac{r}{f}, 0) \tag{2.7}$$

where Buf_f_{prev}: Previous number of bits in the buffer.

R_{k-1}: Actual number of bits used for encoding the previous frame.

r : Channel rate.

f : Frame rate.

r / f : Number of bits taken by the channel per frame interval.

Buf_f : Buffer fullness.

2) Determine how many frames need to be skipped.

If $Buf\_f \geq M$, skip frames until the buffer fullness is below M , where $M = \frac{r}{f}$ is threshold of buffer fullness. For each skipped frame, buffer fullness is reduced by $\frac{r}{f}$ bits.

If L frames are skipped, the updated buffer fullness is:

$$Buf\_f = \max(Buf\_f_{prev} + R_{k-1} - (L+1)\frac{r}{f}, 0) \tag{2.8}$$

2.2.2.2 Frame Layer Rate Control

Frame layer rate control selects a target number of bits based on the buffer fullness to encode the current frame. The frame target is calculated by (2.9) [6]

$$T = \frac{r}{f} - \Delta \tag{2.9}$$

where Δ is defined as

$$\Delta = \begin{cases} \dfrac{Buf\_f}{f} & Buf\_f > ZM \\ Buf\_f - ZM & otherwise \end{cases} \tag{2.10}$$

The default value for Z is 0.1. Δ is the feedback from the buffer fullness Buf_f. Only small variation in target number of bits T is achieved because low delay applications are finally aimed at.

2.2.2.3 Macroblock Layer Rate Control

TMN8 macroblock layer rate control algorithm is based on the following logarithmic $R(q)$ model [4].

$$R(q) = \begin{cases} \frac{1}{2}\log_2(2e^2\frac{\sigma^2}{q^2}) & if \quad \frac{\sigma^2}{q^2} > \frac{1}{2e} \\ \frac{e\sigma^2}{\ln 2 \times q^2} & if \quad \frac{\sigma^2}{q^2} \le \frac{1}{2e} \end{cases} \tag{2.11}$$

The rate control scheme consists of the following 5 steps, as depicted in Fig.2.7.

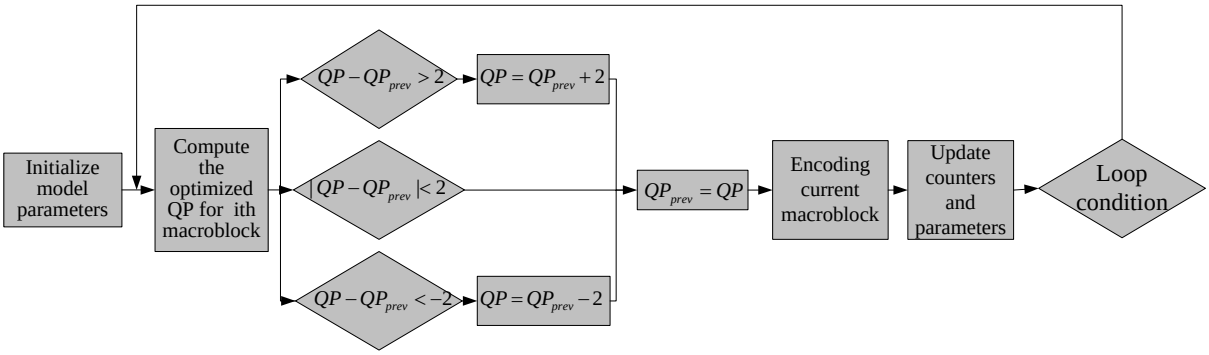


Figure 2.7 Flowchart of the TMN8 macroblock layer rate control [4]

1), Initialization. The motion estimation for the whole frame has to be done before using the macroblock layer rate control. The sum of weighted variances of the macroblock prediction errors S' is computed as

$$S' = \sum_{k=1}^{N} \alpha_k \sigma_k \tag{2.12}$$

$$\text{where } \alpha_k = \begin{cases} 2\dfrac{T}{16^2 N_{mb}}(1-\sigma_k)+\sigma_k & \dfrac{T}{16^2 N_{mb}} < 0.5 \\ 1 & otherwise \end{cases} \tag{2.13}$$

σ_k^2 is the variance of the luminance and chrominance values in the i th macroblock. If the i th macroblock has a type intra, then $\sigma_k^2 = \sigma_k^2 / 3$. T is the target number of bits for the frame and N_{mb} is the number of macroblocks in a frame.

2), Calculate optimized Q for the i th macroblock by

$$Q_i^* = \begin{cases} \sqrt{\dfrac{16^2 K \sigma_i S_i'}{L \alpha_i}} & if \quad L = \beta_i - 16^2 N_i C > 0 \\ 2(QP_{prev} + 2) & otherwise \end{cases} \tag{2.15}$$

where β_i is the number of bits left for encoding the frame, and N_i is the number of macroblocks left to be encoded in the current frame. K and C are model parameters.

3), Adjust QP and encode macroblock. QP is set to ($Q_i^* / 2$) rounded to the nearest integer in {1,2,… 31}. $DQUANT = QP - QP_{prev}$

If $DQUANT > 2$, set $DQUANT = 2$. If $DQUANT < -2$, set $DQUANT = -2$. Then encode the macroblock with $QP = DQUANT + QP_{prev}$.

4), Update counters and model parameters

5), Loop condition. If $i = N_{mb}$, stop the loop. Otherwise, let $i = i + 1$ and go to step 2.

TMN8 rate control algorithm can meet the target bit rate pretty accurately because the QP is adjusted at a macroblock layer. However, the RD model is too simple to characterize statistics of a frame by estimating only the variance of the frame [4] [5].

2.2.3 VM8 Rate Control Algorithm

MPEG-4 provides a scalable rate control (SRC) scheme as a guideline for implementation. It has been adopted as part of the standard, and is used in Verification Model Version 8 (VM8) and in succeeding versions [2]. The SRC algorithm is based on the assumptions that adjacent video frames of the same type have very similar RD characteristics, and the RD function can be modeled by the following quadratic formula derived in [49].

$$R(q) = a \times q^{-1} + b \times q^{-2} \tag{2.16}$$

There are four stages for VM8 rate control algorithm: initialization stage, pre-encoding stage, encoding stage and post-encoding stage [3] [50].

2.2.3.1 Initialization Stage

At the initialization stage, the encoder needs to initialize the buffer size based on latency requirement, initialize the buffer fullness in the middle level, and initialize bits count and model parameters.

2.2.3.2 Pre-encoding Stage

In the pre-encoding stage, the rate control scheme contains the following major parts: target bits allocation, target bits adjustment based on the buffer status and quantization parameter calculation.

The target bit count for a P frame at time $(t+1)$ is estimated by

$$R_{t+1} = \frac{R_t}{N_t} \times (1 - wf) + A_t \times wf \tag{2.17}$$

where R_t is the remaining bit counts at time t, N_t is the remaining number of P frames at time t, and A_t is actual bits used for the previous P frame. wf is a weighting factor with default value 0.05 determining impact from the previous frame in the target bit allocation for the current frame.

Based on the buffer fullness, the target bit counts can be further adjusted as follows:

$$R_t = \frac{(B_t + 2 \times (B_{size} - B_t))}{(2 \times B_t + (B_{size} - B_t))} \times R_t \tag{2.18}$$

where B_t is the current buffer fullness at time t and B_{size} is the buffer size.

$$R_t = \begin{cases} R_t + B_t - (1 - Y_{margin}) B_{size} & if \quad (R_t + B_t) > (1 - Y_{margin}) B_{size} \\ C_0 - R_t - B_t + Y_{margin} B_{size} & if \quad (R_t + B_t - C_0) < Y_{margin} B_{size} \end{cases} \tag{2.19}$$

where $C_0 = \frac{r}{f}$ is the channel output rate, Y_{margin} is the safety margin of buffer to avoid buffer overflow and underflow.

2.2.3.3 Encoding Stage

Quantization parameters can be calculated from the pre-encoding stage given the target bit count and the rate model. In this stage, the current video frame is encoded with the quantization parameter and the actual bit rate needs to be recorded.

The macroblock layer rate control can be chosen to be active in this stage. Normally, for low delay applications, the macroblock layer rate control is required to

meet strict buffer regulations. However, macroblock layer rate control is costly for low bit rate applications since quantization parameter change information needs to be transmitted as additional overhead. There is about 10% coding efficiency loss at low bit rate applications by using the macroblock layer rate control [2].

2.2.3.4 Post-encoding Stage

In the post-encoding stage, the rate control has two major tasks: updating the RD model parameters and performing the frame-skipping scheme to prevent the potential buffer overflow and underflow.

The SRC rate control introduced in this section is widely used in MPEG-4 based applications because it provides scalability to different video objects. However, the SRC rate control algorithm often suffers from relatively large control error due to the limited accuracy and robustness of its rate model (2.19).

2.3 Optimization Tools

The rate control problem can be formulated as an optimization problem. The optimal solution to minimize distortion for a given rate constraint needs to be searched among a finite but probably very large set of operating points. To avoid employing an exhaustive search, Lagrange multiplier and dynamic programming are the two commonly used techniques on optimization problems [16].

2.3.1 Lagrange Multiplier Method

To solve a constrained optimization problem as:

$$\min_{B \in S_B} D(h) \qquad \text{subject to: } R(h) \le R_{\max} \tag{2.20}$$

the Lagrange multiplier method is used to convert it into an unconstrained problem:

$$\min_{B \in S_B}(D(h) + \lambda R(h)) = \min_{B \in S_B}(J(h)) \tag{2.21}$$

where $J(h)$ is the Lagrange cost, and h is the coding parameter. λ is the Lagrange multiplier to tradeoff the cost between rate and distortion. A small value in λ favors minimizing distortion over rate, and a large value in λ favors minimizing rate over distortion.

The Lagrange optimization can be explained by Fig. 2.8. For a given slope $-\lambda$, the minimization of Lagrange cost is to find the operating point on the operational rate distortion (ORD) curve that is first intersected by a line with slope $-\lambda$. Moreover, in order to find the good operating point which satisfies the rate or distortion constraints, the appropriate λ needs to be chosen. The review of RD optimization in video compression can be found in [18] [19].

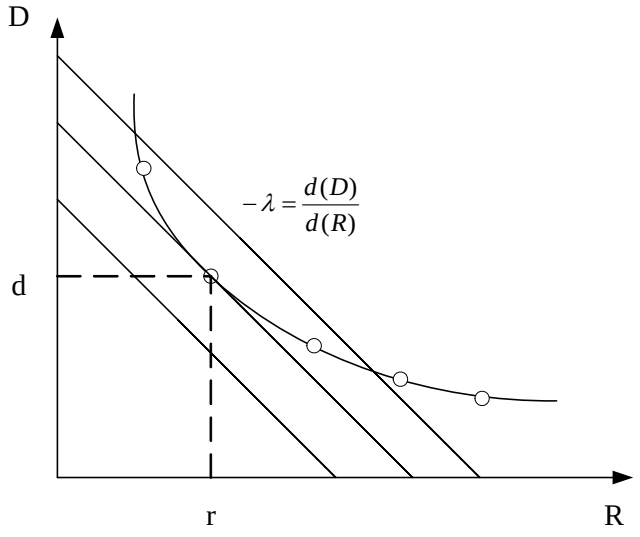


Figure 2.8 Lagrange optimization [16]

2.3.2 Dynamic Programming

Dynamic programming (DP) is another approach to find the minimum cost path through a tree or trellis where each branch has a cost attached and the cost is additive over the path [20][51]. DP is based on the principle that the optimal solution of an entire problem consists of optimal solutions to its sub-problems [16] [61]. When the exhaustive search solves the overall problem by solving the same sub-problems again and again, DP solves each sub-problem just once and stores the solutions in memory. Therefore, DP algorithm significantly reduces computational complexity than exhaustive search. However, for problems with very large dimensions, straightforward DP algorithm is still impractical due to the large delay. A greedy suboptimal matching pursuit approach may be needed [21].

2.4 Summary

This chapter briefly reviews the most popular video coding standards emphasizing their corresponding rate control algorithms. The most well-known rate control algorithms adopted by these international standard test models (TM5, TMN8, VM8) are discussed in detail. Two commonly used optimization tools in rate control are introduced.

CHAPTER 3

ROI BASED RATE CONTROL

3.1 Motivation

Due to the extensive applications of videoconferencing, research on low bit rate video coding is very active over the past ten years [2][10]. It is a very challenging topic since the available bandwidth is very limited. Rate control therefore becomes essential in low bit rate video coding, because it is the mechanism responsible to optimize the video quality at a given target bit rate.

From the review in chapter 2, it can be seen that the existing low bit rate coding solutions can be grouped into two schools of thought: block-based and object-based. Typical low bit rate video telecommunication standards H.26x are all block-based motion compensated DCT coding techniques [10][11][12]. The second school of thought (MPEG4 Visual) supports video content representation [9] It can benefit from visual perception properties because it has the capability of dealing different contents in a video differently, which is more like the way human eyes process a video. Moreover, human eyes are the final judgers to evaluate processed video sequences. So object-based coding has the potential to overcome the conventional block-based coding. However, due to the high computational complexity involved in solving the challenging

computer vision problems, the object-based approach cannot replace the block-based approach for many real-time low bit rate video coding applications [9].

It is natural to combine the two schools of thought together, using a simplified video content segmentation method to design weights for different regions within a video sequence based on human visual properties. With the appropriate use of weighting information, the efficiency of the conventional block-based coding method can achieve a higher peak than ever before. A rate control scheme motivated by this idea will conceptually provide better human visual quality for the block-based coding standards.

3.2 Existing Solutions

The idea of using the weights for different video contents to control the bit rate is not new. In 1991, Puri and Aravind [22] proposed an activity-based adaptive quantization for MPEG video. They presented a frame layer bit model to estimate the needed bit-count for video frames with different complexities, and a set of macroblock layer bit models to estimate the needed bit-count for each macroblock type. The models are trained using a number of sequences off line. At frame layer, they calculated the variances for a group of pictures (GOP) which contain 15 frames, and based on the variances they classified each frame into 8 levels, then estimated the needed bit-count for each frame according to the bit model. At macroblock layer, they divided all macorblocks by their activity levels, according to homogeneity, flatness, various degrees of texture and presence and strength of edges. Then they searched the quantization parameter for each class and estimated the bit-count by the models. This

process may be repeated for many times until the estimated bit-count is smaller than the target bit-count for the current frame. This scheme of macroblock layer rate control is illustrated in Fig. 3.1. It is the prototype of a single-path rate control.

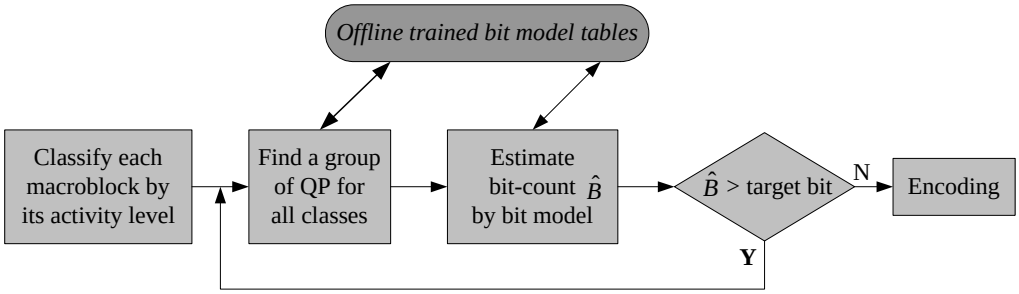


Figure 3.1. Flow chart for macroblock layer rate control [22]

The drawback is all bit models trained off line are targeted at 1Mb/s bit rate MPEG video. So it is hard to extend it to other coding platforms and other bit rate applications.

Eleftheriadis and Jacquin [23] first abstracted face model from teleconferencing video. They applied an elliptical model to detect human face. With the face location information, they applied a coarser quantizer outside the face region and a finer quantizer inside the face region. This rate control is designed for a 3D subband-based video coder. The drawback is there was no buffer regulation, and the authors did not introduce how to pick up the right quantizer parameters. So it is too simple to be a rate control scheme in real-time teleconferencing applications.

In 2001, Sethuraman and Krishnamurthy proposed a macroblock layer model based rate control scheme for MPEG4 video [24]. Skin color is the only characteristic they used to locate a face. They divided all macroblocks into three groups: background, foreground, and transition. The differences between each adjacent group are fixed at 2.

The required bit-count for a frame is calculated by a linear model: $R = \frac{x}{QP} f(s)$, where x is a constant, QP is the quantization level, and $f(s)$ is a measure of the motion compensated distortion. This rate control scheme is for macroblock layer only (Fig. 3.2). The linear rate model is good for slow motion video. However, the model accuracy is a major problem as long as the quantizers need to change drastically from frame to frame. There is no buffer regulation involved in this work.

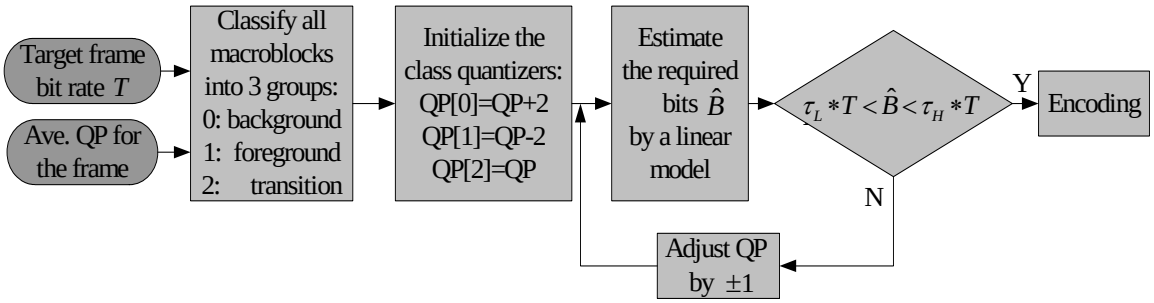
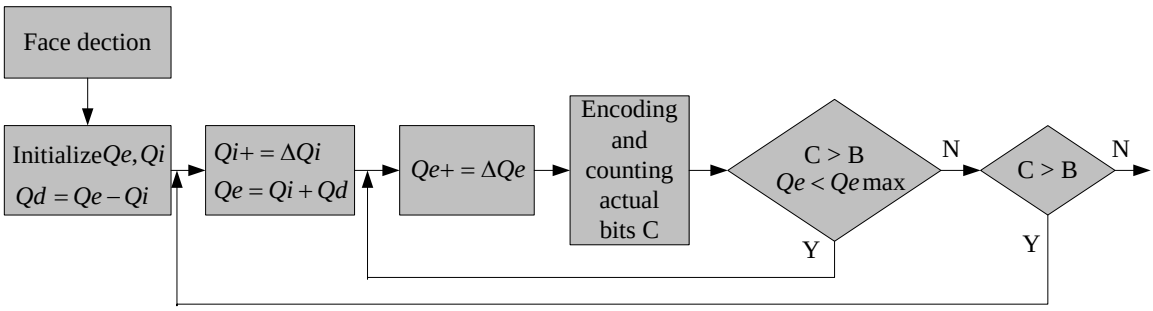


Figure 3.2. Flow chart of macroblock layer rate control scheme [23]

Daly, Matthews and Ribas-Corbera [25] described a face-based visually optimized coding scheme. They also used a simple geometric model for face detection, and local visual sensitivity is applied for further ROI determination. The rate estimation model adopted is the same as TMN8 [6], which is a statistical model. The model is not scalable to different video contents. Also no buffer control is considered.

Hartune *et al* proposed an object-oriented rate control scheme in 1998 [26]. Their scheme is developed for H.263 compatible video. They use geometric shape, skin color and motion information to detect face first, and then apply different level quantizers to encode face region and background region. They need to repeat encoding

many times to achieve the target bit rate because there is no model to estimate the needed bit count (Fig. 3.3.). The biggest problem of this rate control scheme for teleconferencing video application is the complexity, since it needs to continually compute quantized DCT coefficients, Huffman codes and the resulting bit rates.



Qe : Quantization step in non-face (exterior) region.
Qi : Quantization step in facial (interior) region.
B: Bit budget for current predicted frame.
Figure 3.3. Macroblock layer rate control flowchart [26]

The recent work published by Song and Kuo in [5] proposed a region-based rate control scheme for VBR video without buffer constraint. They added two components to the H.263+ [10] baseline coder: a moving region segmentation algorithm to improve the quality of moving regions of the underlying video and an encoding frame selection algorithm to enhance the quality of interpolated frames Fig. 3.4. They define the ROI by an image processing tool: histogram of difference. They designed rate control scheme for frame layer using the same rate model in VM8 [3], which can provide scalability for different video contents. This scheme is claimed for VBR without buffer constraint applications only, because it has a drawback at buffer regulation.

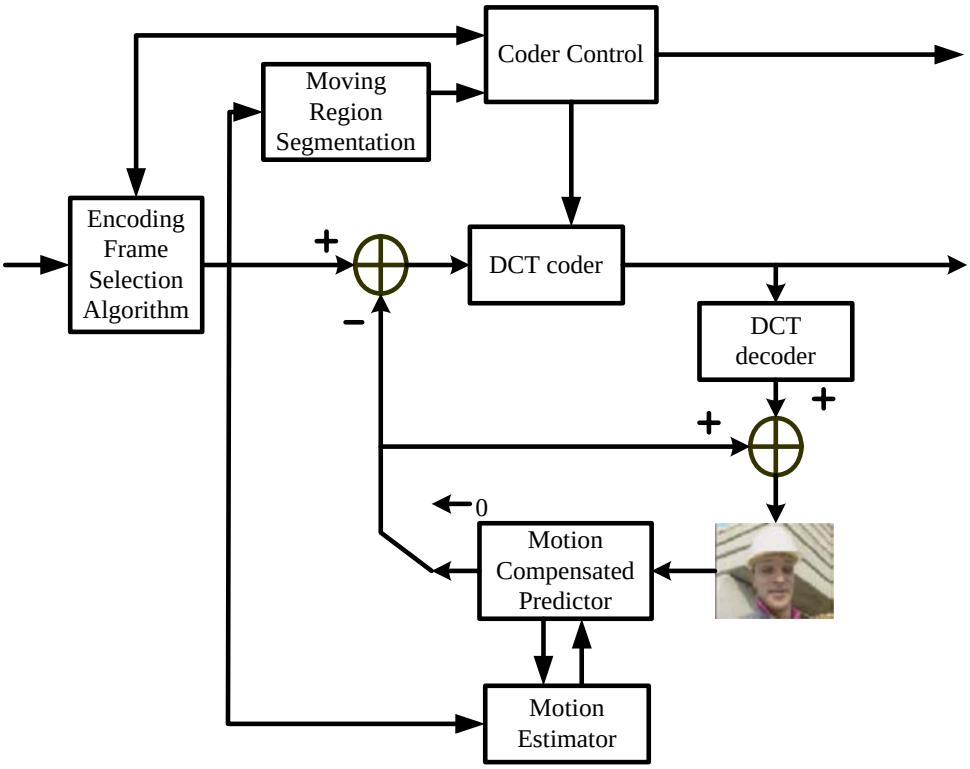
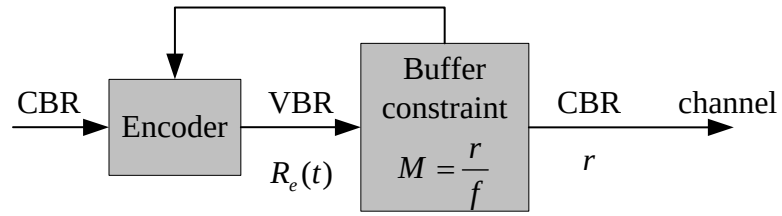


Figure 3.4. Frame layer rate control block diagram [5]

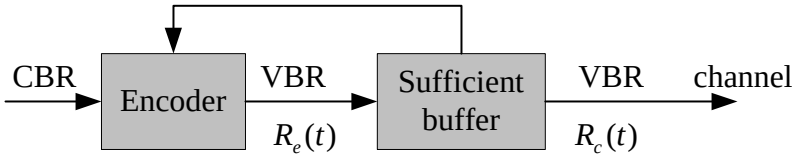
From the literature survey, it can be noted that there does not exist a mature ROI based rate control scheme for the H.263 compatible video coding platform for real-time conferencing applications. The purpose of the proposed research is to design such a rate control scheme to achieve perceptual improvements for real time video conferencing applications, while confirming to either CBR channel with low time delay and strict buffer constraints or VBR channel without buffer constraints.

The two rate control schemes are depicted in Fig. 3.6. In both cases, buffer is needed at the interface from encoder to channel to smooth out the variations in bit rate. For a CBR channel, the buffer's function is to absorb all the variations of the output bit rate generated by the encoder and provide CBR bit stream to the channel. The

maximum buffer size is limited by time delay. In a real time multimedia communication system, maximum time delay is 100ms, when the target frame rate $f = 10\,fps$, channel rate is denoted by r, then the maximum buffer size is $M = \frac{r}{f}$, which is a very critical requirement in design. For the VBR channel rate control, the U-VBR case is taken into account, which means sufficient buffer is available to transfer $R_e(t)$ to $R_c(t)$, where $R_e(t)$ indicates the variable bit rate of output of the encoder at time t, and $R_c(t)$ is the constant bit rate of the channel at time t. The details of these two systems are discussed in this proposal.



(a)



(b)

Figure 3.5. Proposed rate control (a) Rate control for CBR channel (b) Rate control for VBR channel

3.3 Summary

In this chapter, the motivation to integrate block-based video coding and object-based video coding is discussed. All the exiting approaches of object based rate control

schemes are reviewed. Their advantages and disadvantages are presented. It concludes with the proposed ROI based rate control scheme at the end.

CHAPTER 4

ROI DETECTION AND TRACKING

4.1 Introduction

The proposed ROI based rate control scheme assigns bits based on the weighting factors of different video contents. Hence accurately separating ROI from its background is very critical for the system performance. Considering the video conferencing applications under all environments, the background can be complicated and moving other than uniform color and still scenery. In video conferencing applications, most sequences that need to be transmitted are head and shoulder sequences. Therefore, the ROI segmentation can be formulated as human face detection and tracking in complicated and moving background. To meet the requirements of the real-time application, computation complexity of the detection and tracking algorithm must be very low.

In the most recent work of ROI based rate control proposed by Song and Kuo [5], a moving region segmentation algorithm is used for ROI detection. It may be observed that this algorithm is good for face detection in a still or near still background. If the background is also moving, the detected ROI will contain not only the real ROI but also the moving background. This will degrade the performance of rate control. So,

for a complicated and moving background, segmentation based on motion is not sufficient. A face detection algorithm to preprocess the video is needed.

There are many techniques proposed for face detection since it plays an important role in applications such as video surveillance, intelligent human computer interaction and face recognition. Reference [27] classified all the face detection techniques into four categories: (1), knowledge-based, (2) feature invariant, (3) template matching and (4) appearance-based.

The knowledge-based methods encode human knowledge of what constitutes a typical face. These rules normally capture the relationships among facial features. For instance, a frontal face often appears with two eyes that are symmetric to each other, a nose and a mouth. The relationships among features can be represented by their relative positions and distances. Normally, the features in an input image need to be extracted first, and face candidates are identified based on the coded rules. A representative work of knowledge-based method is hierarchical rule-based method [28]. The difficulty in this approach is how to translate human knowledge into well-defined rules. If the rules are too general, there may exist false positives. If the rules are too strict, the system may fail to detect real faces that do not meet all the rules. Moreover, it is challenging to enumerate all possible cases for different poses. On the other hand, this approach can work very well for video conferencing applications where frontal faces are the major objects being detected.

Feature invariant method utilizes the structural features that are not sensitive to pose, viewpoint and lighting conditions for detecting human faces. The implied

assumption is human can effortlessly detect faces under all lighting conditions and in different poses. Therefore there must exist features which are invariant over all these changes. Commonly facial features such as eyes, eyebrows, nose, mouth are extracted using edge detectors [29]. A severe problem with these feature-based algorithms is that the image features can be corrupted due to illumination, noise and occlusion. Feature boundaries can be weakened for faces, while shadows can cause numerous strong edges as well. So, there are other features being used like texture [30] and skin color [31]. However, none of these features alone can provide adequate detection results.

Template matching method uses several standard patterns to describe an entire face and facial features. For a given input image, the correlation values with the standard patterns are computed for the contours of face, eyes, nose and mouth independently. The existence of the face is determined based on the correlation values. This approach has the advantage of being simple to implement. However, it has been proven to be insufficient for face detection since it cannot effectively deal with variation in scale, pose and shape [32].

In contrast to template matching methods [32][45] where templates are determined by experts, the models of appearance-based method [33][46] are learned from a training image set. In general, appearance-based methods rely on techniques from statistical analysis and machine learning to find the relevant characteristics of face and nonface images. The learned characteristics are in the form of distribution models or discriminant functions that are used for face detection. Moreover, for the sake of computational complexity and detection accuracy, dimensionality reduction is usually

carried out. The representative work of appearance-based method is the eigen-face based face detection technique [33].

A novel face detection system that combines a feature invariant method and a knowledge-based method is proposed. In the first stage of detection, a feature invariant method is applied, i.e., using skin color detection to make sure the system is insensitive to variabilities like face orientation and lighting conditions; The outputs of the first stage are face candidate regions which are the inputs to the second stage. In the second stage, a mosaic rule-based method is applied to offset the weakness of first stage detection and achieve real-time execution capability. At the end, the system outputs the locations of detected faces.

Although the proposed face detection algorithm is based on full consideration of computational complexity, it still introduces too much computation burden if it has to be done for each frame. In [34], the authors skip some of the detection steps for inter frames. This reduces some of the computational complexity but at the price of false detection. However, accuracy of detection is very critical for rate control scheme in all frames.

The idea to reduce the computational complexity is to utilize the face location in the intra frame and other information to track the face in inter frames. In the field of computer vision, there are a number of algorithms for region-based tracking, like kernel-based object tracking [35], histogram-based object tracking [36] and spatiogram-based tracking [37]. Although these algorithms can provide good tracking results, there are too many extra calculations involved.

In H.263 standard series [10][11], motion estimation and prediction algorithms are applied to utilize the temporal correlation to efficiently compress a video. The same concept of temporal correlation can be extended to efficiently track faces in motion-compensated coding frames. To cope with the H.263 standard series [10][6], more time can be spent on intra frame face detection, while the encoder deals with intra frame compression. However for inter frames, alleviating the computational burden and reducing delay time need to be considered. A face tracking method with only motion information is proposed for inter frame face tracking.

Motion information is achieved during compression, so using motion information to track face location is also called compressed domain face location. There are several approaches on compressed domain face location. Reference [38] uses the DC component of chrominance blocks to detect skin color region. In [39], segmentation of background and foreground is performed using DCT coefficients only. The classification is based on thresholding the average temporal change of each region. Motion vectors have also been used in clustering background and foreground [40][41][42]. In [41], translational motion vectors are accumulated over a number of frames and the magnitude of the displacement is calculated for each macroblock; macroblocks are subsequently assigned to regions by uniformly quantizing the magnitude of the displacement, and then, connected region with similar motion is clustered. Reference [42] proposes a method for manually identifying moving objects in the compressed stream based on macroblock motion vectors. Reference [40] modifies

the work in [42] to derive motion information for the I-frames, so that the tracking based on motion vectors can exceed the GOP boundary in MPEG-2 video [10].

In the proposed face tracking algorithm, no human intervention is used as a backup method as the objective is real-time application. To meet the high accuracy requirement for automatic face tracking, not only the accumulated motion vectors over a number of frames but also the motion vectors of individual frames are considered to achieve a higher tracking accuracy than the existing solutions.

The face detection and tracking algorithm is needed for preprocessing of rate control scheme for H.263 [6][10] compatible codec. The quantization parameter changes are based on a macroblock basis (Fig.4.6). It indicates that the ROI and non_ROI segmentation should be implemented at the macroblock level. Hence the face detection algorithm needs to provide face location information with macroblock level precision.

4.2 Macroblock Level Face Detection in Intra Frame

The face detection method designed for intra frame is a combination of feather based (skin color) method [6] and knowledge-based (mosaic rule-based) method [3]. There are two stages in this face detection system: skin color detection and mosaic rule-based detection. The block diagram of the proposed face detection method for intra frame is shown in Fig. 4.1.

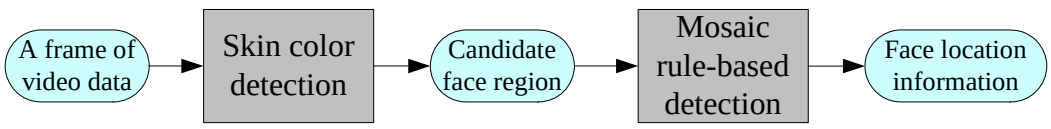


Figure 4.1. Block diagram of the proposed intra frame face detection

4.2.1 Stage One: Skin Color Detection

The color components of common intermediate format (CIF) and quarter CIF (QCIF) video sequences are: Y,Cb,Cr [43]. Y indicates the luminance value, it does not contain color information. The information related to color like hue and saturation is retained in the chrominance components Cb and Cr.

The appearance of human face color is not only determined by Cb and Cr, but also is related to the lighting conditions. Skin color definition in Cb-Cr domain [34] cannot provide enough accuracy for complicated lighting conditions.

A lighting compensation method [31] is used at the very beginning of the system to normalize the color appearance. This is based on the assumption that there are always real white pixels in the eye regions in video conferencing sequences. Therefore, about top 5% luminance value should be linearly scaled to 255. All the luminance values Y can be linearly scaled to new luminance values Y'. Figure 4.2. shows the images before and after lighting compensation.

(a) (b) (c) (d)

Figure 4.2. Lighting compensation effect. (a) & (c) before lighting compensation; (b) & (d) after lighting compensation.

The selection of different color spaces is very critical in skin color modeling. Skin color can be modeled mainly by chrominance information, however, it is also nonlinearly dependent on luminance, especially on the extreme luminance. For example, dark brown color contains the same chrominance components as yellow skin color, but it is too dark to be a skin color. To solve the difficult problem in detecting low luminance and high luminance skin color, Hsu and Jain [31] proposed a nonlinear color transform which can provide a transformed color space most appropriate for skin color detection. Their algorithm is adopted in this research. The elliptical skin color model is described by (4.1) and (4.2) in the transformed $C_b^{'}C_r^{'}$ space.

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} C_b^{'} - c_x \\ C_r^{'} - c_y \end{bmatrix} \tag{4.1}$$

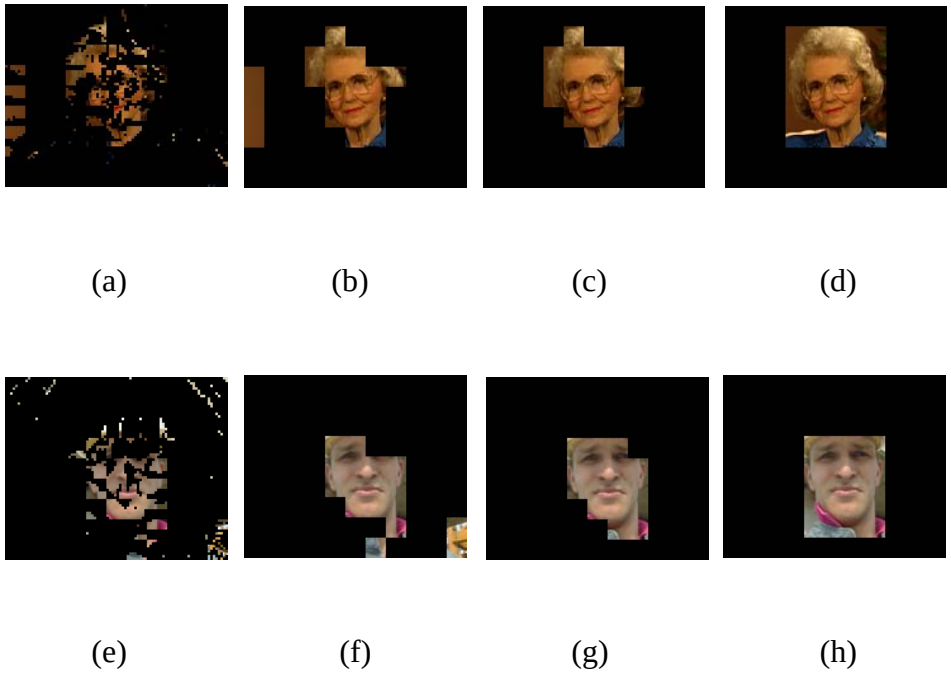
Equation (4.1) defines a mapping from a pair of $C_b^{'}$ $C_r^{'}$ value to x y value, so that the mapped skin color x y can be modeled as an ellipse defined by (4.2).

$$\frac{(x - e_{c_x})^2}{aa^2} + \frac{(y - e_{c_y})^2}{bb^2} = 1 \tag{4.2}$$

where $c_x = 109.38$, $c_y = 152.02$, $\theta = 2.53$ (in radian). $e_{c_x} = 1.60$, $e_{c_y} = 2.41$, $aa = 25.39$, $bb = 14.03$.

Transformed chrominance value of each pixel is processed by the skin color filter which is defined in $C_b^{'}C_r^{'}$ domain. However, the face detection scheme is designed to work on a macroblock layer. Hence, pixel-based skin color classification information needs to be integrated into the macroblock layer. By observing the face region in a

macroblock divided frame, it can be summarized that if more than a certain number of pixels in a macroblock are skin color pixels, then the macroblock is probably a face region. The threshold is determined by taking into account the tradeoff between face edge loss and skin color noise disturbance.



(a) (b) (c) (d)

(e) (f) (g) (h)

Figure 4.3. Skin color detection step-by-step results. (a) & (e) skin color pixels; (b) & (f) integrated skin color macroblocks; (c) & (g) face candidate macroblocks after median filter; (d) & (h) face candidate after projection.

The classification results are written into a binary mask image, where ‘1’ indicates a macroblock is face candidate, and ‘0’ means a background macroblock. The binary mask image of face candidate may have isolated values. A 3x3 median filter is applied to smooth out the binary mask image and to remove the noise. The effects of median filter are shown in Fig. 4.3., where (b) & (f) are face candidate regions before

the median filter, and (c) & (g) are the corresponding results after the median filter. It can be seen that the face candidate regions are smoothed and the isolated macroblocks which do not contain faces are eliminated by applying the median filter.

Considering that the candidate face region with a rectangular shape is preferred by the second stage face detection, a simple strategy [34] for rectangular candidates segmentation for the sake of low complexity is adopted. The filtered binary mask image is then projected horizontally and vertically. Based on the zero-runs and nonzero-runs in these two directions, the rectangular face candidate region can be obtained. Figure 4.3 (d) & (h) shows the final results of first stage face detection.

As a summary of this section, the block diagram of first stage face detection is depicted in Fig. 4.4.

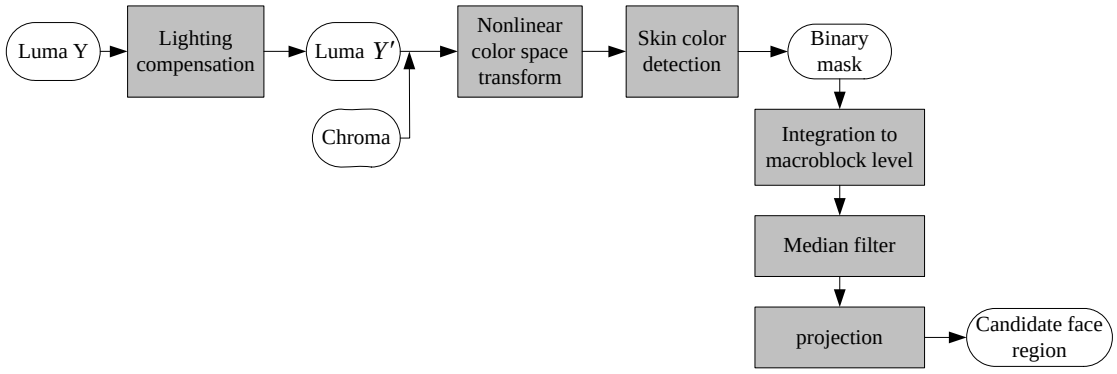


Figure 4.4. Block diagram of face detection stage one: skin color detection

4.2.2 Stage Two: Mosaic Rule Based Detection

To design a well-performed face detection system, different characteristics of faces need to be utilized. The first stage uses only the color information and a little bit shape information. There are more common features of human faces that need to be

utilized for further detection, like the position of eyes, nose, mouth etc. To accurately describe these features while keeping the description effective for all kinds of faces is a difficult task.

Researchers in the field of pattern recognition found that to describe the inter-relationship among different objects inside a pattern is much more efficient [28][44]. This approach is based on the theory of the syntax of language. By introducing the concept of formal grammar, syntax classifiers can be represented as a string of symbols. Instead of carrying on an analysis based on quantitative characteristics of a pattern, the inter-relationship among the primitives that makes up the patterns can be highlighted.

Primitives are the basic units of the syntactic approach [27][28][44]. It is important to choose the appropriate primitives for the analysis. If the chosen primitives are too low, the rules, which are used to describe patterns, will be too complicated; otherwise, some important structure information will be defined with the primitives, which makes it too complicated to define the primitives, meanwhile, the rules will be too easy. The tradeoff between complexity of primitives and rules needs to be balanced.

Designing primitives and rules for human faces is described in [28]. They define the average luminance value of each mosaic cell as primitives, and a set of rules are summarized based on mosaic images.

A mosaic image is an image constructed at different resolutions. The original image is divided into square cells of equal sizes. In each cell, there are N x N pixels, where N is the length of a cell. All the pixels in a cell are represented by the average luminance value of that cell. Figure 4.5 shows mosaic images constructed at different

resolutions for the first frame of QCIF Foreman. It can be seen that the mosaic image with too small cell size still contains too much detail information. On the other hand, a mosaic image with too large cell size may lose the texture information at all. So the pick up an appropriate cell size is important for the detection algorithm design.

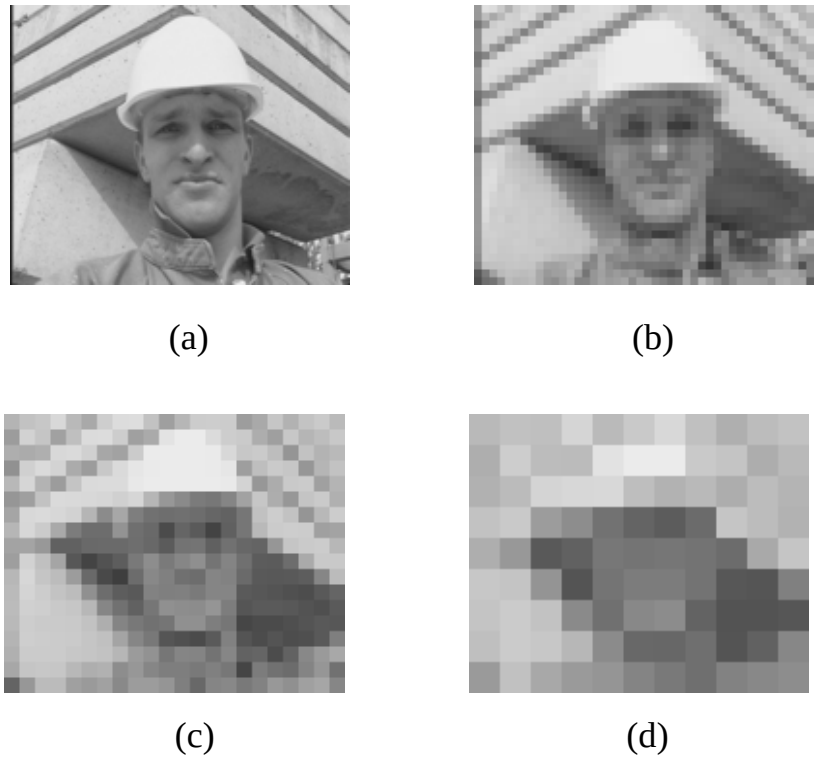


Figure 4.5. Mosaic images at different resolutions for the first frame of QCIF Foreman sequence (a) original image (b) N=4 mosaic image (c) N=8 mosaic image (d) N=16 mosaic image

Mosaic-based human face detection was an active topic around 1993 [28]. It did not become a main trend algorithm in real time face detection because the rule matching process is exhausted when searching the face in an entire image. To reduce the

computational complexity in searching, the proposed system applies skin color detection first. Hence the face searching only needs to be done in that candidate face region. Another improvement of the proposed system is reducing the complexity in calculating the mosaic image. In the previous work [28], multiple level mosaic images need to be calculated to match faces with different sizes. Considering in a wireless video conferencing system as the face size does not vary too much, only one level mosaic image is enough. Moreover, to introduce mosaic-based face detection algorithm in H.263 video can further reduce the complexity in calculating mosaic image. The codec based on H.263 [6][10] is macroblock-based, and each macroblock is composed of six blocks: four luminance blocks and two chrominance blocks, as shown in Fig. 4.6. Each luminance block can be viewed as a mosaic cell directly. The value of each mosaic cell is the DC component of the block. So the mosaic image can be constructed without any extra calculation. It indicates the good compatibility of mosaic-based face detection method with the H.263 [6][10] series codec.

The most attractive feature of mosaic-based face detection for the real-time H.263 [10][6] compatible video coding system is: mosaic image contains far less information than the original one. The low complexity property makes it suitable for real-time applications. Second, the algorithm is stable for various sizes face detection. It is insensitive to minor parallel movement and minor rotational movement. Moreover, it utilizes the luminance information, which is a good complement of the detection using chrominance information in the first stage. Therefore, the mosaic-based method is chosen in the second stage face detection.

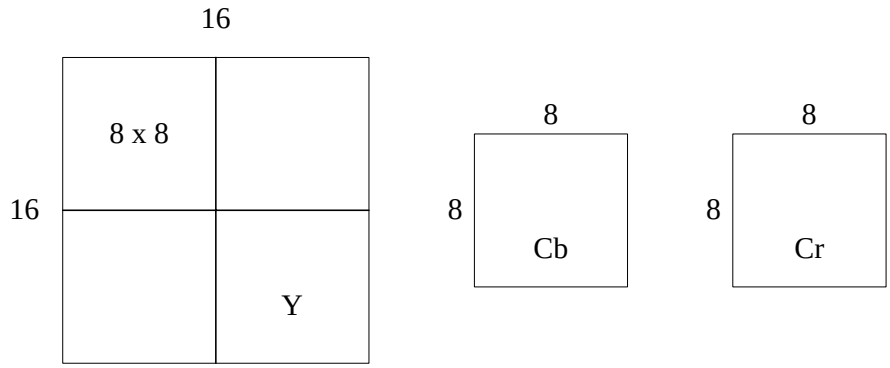


Figure 4.6. Each macroblock (4:2:0 format) consists of six 8x8 blocks, four of them are luminance blocks and two are chrominance blocks.

The flow chart of the second stage face detection is shown in Fig. 4.7. For each segment that contains face candidate, the corresponding binary mask image is checked and the number of "1"s in this segment is counted. Here "1" means the macroblock is face candidate. If the number of face candidate macroblocks is greater than a threshold, the system will look into all the possible size mosaic images inside this segment, otherwise, the system will check another face candidate segment. Before processing a segment, the luminance of macroblocks with mark "0" will be replaced by an average skin luminance value to eliminate the distortion of dark background color. For each possible size mosaic image, human faces are searched by matching several face rules. If all rules are satisfied, a face is detected. Next step is to remove the overlapping macroblocks from the segment to avoid repeating different size mosaic images searching for the same face. When the searching for all the face candidate segments is completed, the face locations are returned. The searching is starting from the largest possible face size and ending with the smallest possible face size. So far, for a QCIF

sequence, only sizes of 3 x 4 and 3 x 3 are implemented. Larger sizes are needed for higher resolution video sequences.

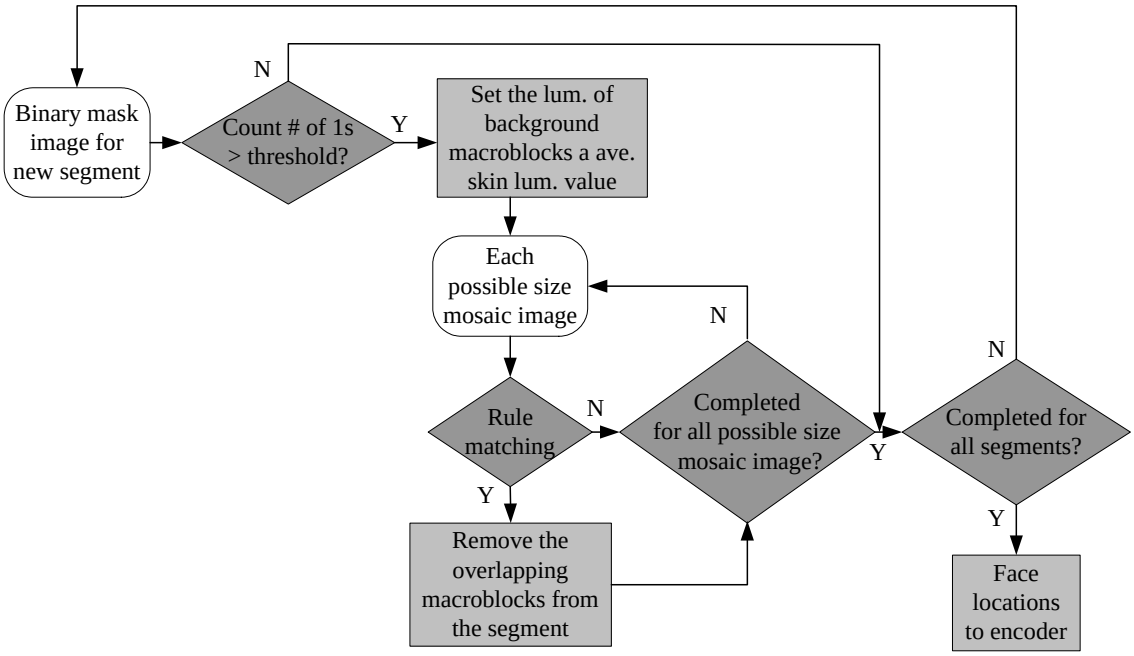


Figure 4.7. Flow chart of face detection stage two: mosaic rule-based detection

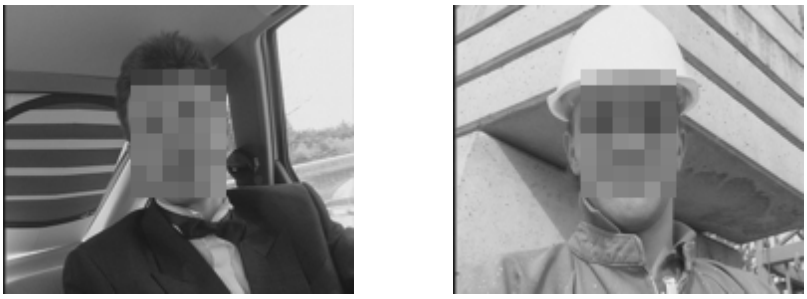
The rules to define a human face in block based mosaic images are listed below:

1), There are two local minimum cells on the same row or on adjacent rows.

2), There is a local minimum cell below the two local minimum cells.

Here the local minimum cell is defined as a cell with the largest luminance value compared to the eight surrounding cells.

Two simulation results are shown in Fig. 4.8. The mosaic part indicates that a face matching is achieved. In order to make sure the face is totally covered by the detected region, the face region is extended by one macroblock along each of four directions: top, bottom, left and right. The system then returns the extended face location.



(a) (b)

Figure 4.8. Mosaic images of human face

4.3 Macroblock Level Face Tracing in Inter Frame

In H.263 [10][6], macroblock based motion estimation and motion compensation is the strategy to compress video data utilizing the temporal correlation. By comparing the current macroblock with each macroblock in the search area of a reference frame, the best matched macroblock can be found. The displacement between the current macroblock and the position of the best matched macrolock is called motion vector. Motion vector for each macroblock in H.263 [10][6] series is composed of two components: displacement in horizontal direction and displacement in vertical direction. Motion vectors describe the motion level of each macroblock. The idea of utilizing motion vector to track ROI is relatively straightforward and computationally tractable in a H.263 [10][6] compatible video codec.

The proposed scheme for face tracking using MV is based on the work of [40]. One branch to detect abrupt motion is added to achieve higher level tracking accuracy. The entire procedure is illustrated in Fig. 4.9. First, the motion vectors (MV) of ROI

macroblocks are sent to the tracking system after motion estimation of an inter frame. Then the average MV of the current ROI (MV_{ROI}) is calculated as follows:

$$MV_{ROI} = \frac{\sum_{i=1}^{k} MV_i}{k}, \text{ where k is the number of macroblocks in a ROI} \qquad (4.1)$$

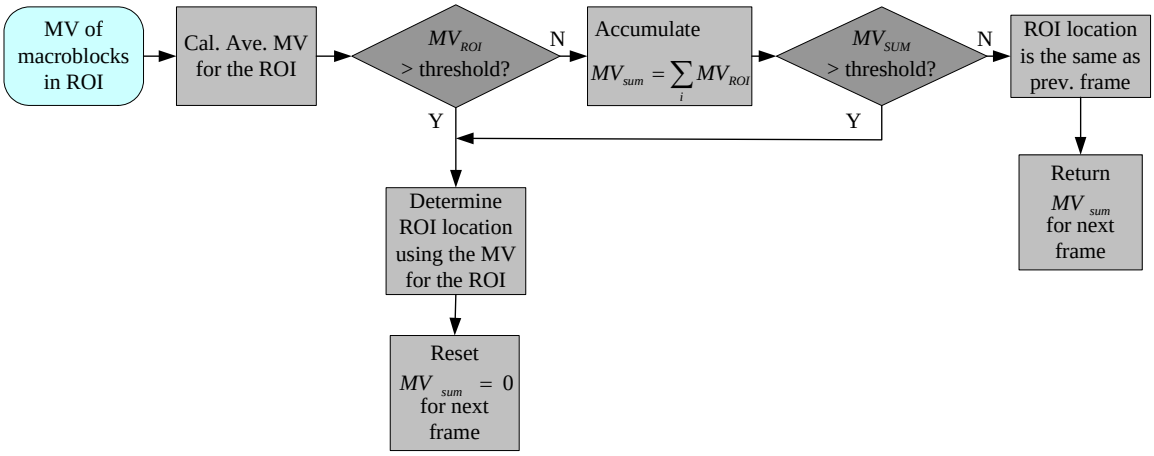


Figure 4.9. Block diagram of face tracking in inter frame

Determining a ROI location adaptation is based on two steps. One is based on the work of [40], which is good for slow motion. The average motion of ROI is accumulated for the continuing frames until the summation is greater than 16, which is one macroblock length. It means the ROI location has steadily moved by one macroblock. The system seems to be too insensitive in case the motion between adjacent frames is relatively large while the motion is near zero in the following frames, and the first branch has failed to track this kind of motion. To compensate the tracking ability of the first branch, another branch is proposed. If the MV_{ROI} is greater than 8,

then the ROI location will move by one macroblock in the MV_{ROI} direction. The ROI location size is designed to be elastic so that more face region can be located within the ROI location, i.e., when the ROI location moves half macroblock length, the ROI location will extend one macroblock in the moving direction, and the ROI location size will shrink back to original when more motion along the same direction is detected. The test results show that the proposed two branch tracking system has a relatively high accuracy in inter frame face tracking.

A series of simulation results are shown below. Figure 4.10 describes the tracking result for Claire.qcif, which shows the tracking algorithm is robust to slight face rotation. Figure 4.11 is the tracking result for Carphone.qcif. It is an example of tracking of parallel motion.

Figure 4.12 is the result for the most challenging sequence: Foreman.qcif, since it contains motions of large rotation, large parallel shift and hand movements. The performance of the tracking algorithm is good for all these cases, even for the fading out procedure of a face, which is shown in the last row of Fig. 4.13.

Figure 4.10. ROI tracking in Claire.qcif

Figure 4.11. ROI tracking in Carphone.qcif

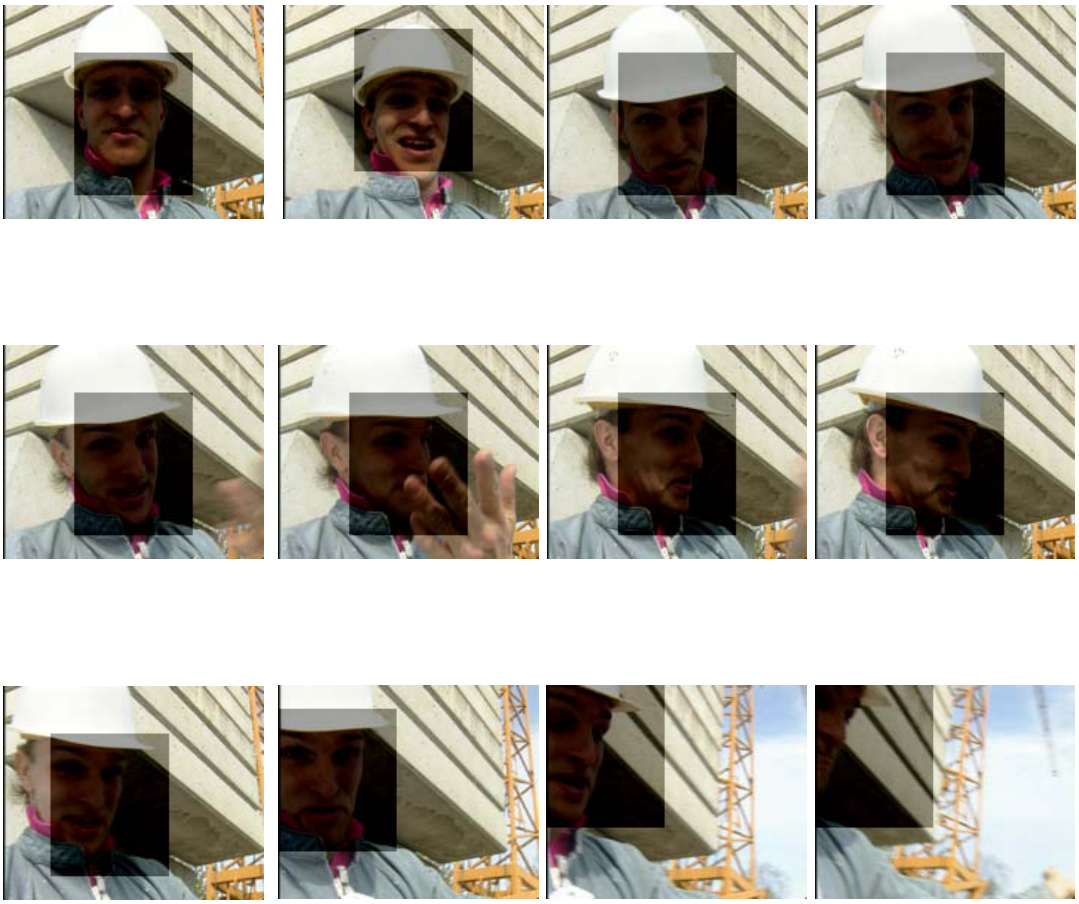


Figure 4.12. ROI tracking in Foreman.qcif

4.4 Summary

A novel face detection and tracking system with very low complexity is proposed to segment human face region from video conferencing sequences in real-time. For intra frames, the proposed detection method is a hybrid skin color and mosaic-rule based face detection and for inter frames the face tracking method uses motion vectors only. To balance sensitivity and stability of the tracking system, motion vectors of each individual frame are also checked in the proposed algorithm. Either the accumulated motion vector greater than 16 pixels (a macroblock length) or the motion vector of individual frame greater than 8 pixels (half macroblock length) is detected and the ROI location is modified along the motion direction. With this modified face tracking method, relatively good tracking efficiency can be obtained.

CHAPTER 5

RATE DISTORTION MODEL

To design a rate control scheme for real-time multimedia communication system, the most critical things are computational complexity and buffer regulation [1][2]. Computational complexity determines the encoding time. To save encoding time, low complexity algorithm is preferred. The feed-forward rate control scheme that requires repeating encoding operations is too complex for real-time applications. Hence a rate model is usually needed to estimate the bit count without actually coding [3] [4] [57]. Besides the famous rate model introduced in chapter 2 [3] [4], there are other rate models assuming various distributions and characteristics of signal source models with associated quantizers, such as: the exponential model in [55], the normalized rate distortion model [56], and the spline approximation model [57]. Buffer regulation determines the efficiency of bandwidth use [4]. Buffer overflow causes frame skipping, i.e., temporal video quality degradation. Buffer underflow causes insufficient use of channel bandwidth, which is a waste of the very limited bandwidth resource. A neat buffer regulation requires an accurate rate model to estimate bit count for the frame being encoded. Therefore, finding an accurate rate model is an important issue before developing a new rate control scheme.

In this chapter, the analysis of rate-distortion models in frame layer, ROI layer and macroblock layer are presented respectively.

5.1 ROI Based Measurements for Rate-Distortion Model

Considering R-D modeling for a ROI based rate control scheme, the rate model has to be scalable with different video contents and the distortion measure should be able to reflect the different weights for distortion of ROI and non-ROI. In the proposed research, a weighted mean square error (MSE) [5][53] is used to measure human visual perceptual quality and a weighted mean absolute difference (MAD) [5][53] to measure video contents. They are defined as:

$$\text{Weighted Distortion: } D_w = \frac{1}{W}\sum_{i=0}^{MM-1}\sum_{j=0}^{NN-1} w_{i,j}(p_{i,j} - \hat{p}_{i,j})^2 \tag{5.1}$$

where, $\hat{p}_{i,j}$ is the intensity of reconstructed pixel at row i and column j, $p_{i,j}$ is the intensity of original pixel at row i and column j. MM is number of pixels per row, NN is number of pixels per column, and $w_{i,j}$ is the weighting factor. Considering the human visual perceptual effect, w_{ROI} should be larger than w_{Non_ROI}.

$$w_{i,j} = \begin{cases} w_{ROI} \\ w_{Non\_ROI} \end{cases} \tag{5.2}$$

$$W = \sum_{i=0}^{MM-1}\sum_{j=0}^{NN-1} w_{i,j} \tag{5.3}$$

$$\text{Weighted MAD: } M_w = \frac{1}{W}\sum_{i=0}^{MM-1}\sum_{j=0}^{NN-1} w_{i,j}\left|p_{i,j}^{cur} - \hat{p}_{i,j}^{ref}\right| \tag{5.4}$$

where, $\hat{p}_{i,j}^{ref}$ is intensity of pixel in the reconstructed reference frame and $p_{i,j}^{cur}$ is the intensity of pixel in the current frame, both at row i and column j.

5.2 Rate-Distortion Modeling

5.2.1 Frame Layer Rate-Distortion Modeling

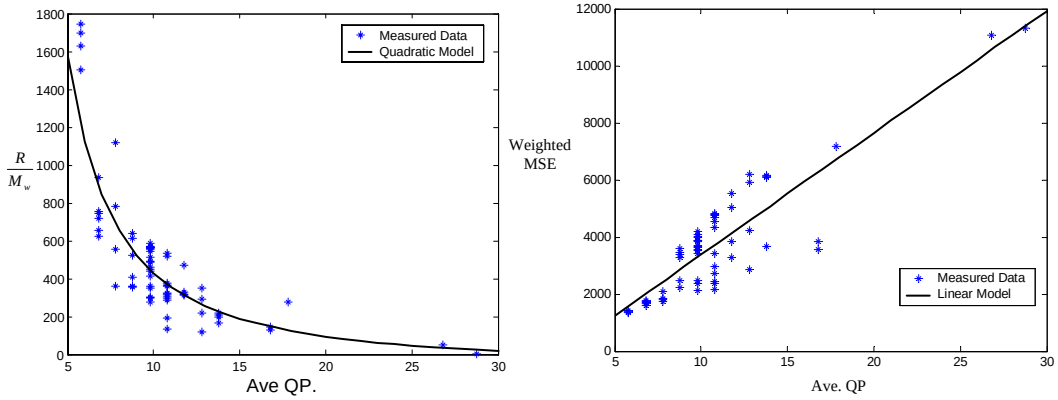
5.2.1.1 Rate-Distortion Modeling

In order to achieve good R-D models for the ROI based rate control scheme, the rate, distortion and quantizer parameter have been examined first, and then the R-D models are derived.

This is implemented on the H.263+ Test Model Near-term Version 8 (TMN8) platform, with the rate control schemes in TMN8 turned off [6]. First, all macroblocks are divided into two groups: ROI and Non_ROI based on a face detection method. Different weighting factors are assigned for each macroblock: $w_{ROI} = 5$ and $w_{Non\_ROI} = 0.5$. The weighted MAD and the weighted MSE for the entire frame can be calculated. A finer quantizer for ROI macroblocks and a coarser quantizer for Non-ROI macoblocks are applied. The quantizer parameters are changed from frame to frame. The rate over weighted MAD with respect to the average QP of each frame is examined, and a quadratic model is employed to fit the measured data. This is done by curve fitting tool in MATLAB [7], which is a fitting method in the least squares sense, i.e., the parameters of the quadratic model are estimated by using the least squares method to minimize the summed square of residuals. The residuals are defined as the errors

between the measured data and the fitted data. Figure 5.1(a) shows the curve fitting for the QCIF carphone sequence.

The distortion versus the average quantization parameter of each frame is also examined, and a linear model is employed to fit the measured data. The data for carphone sequence is shown in Fig. 5.1(b). Carphone sequence contains large motion for both ROI and Non_ROI. It can be viewed as a typical sequence for wireless video conferencing application.



(a) (b)

Figure 5.1. Frame layer R-D modeling (a) Rate modeling (b) Distortion modeling

With the above simulation and analysis, the quadratic rate model and the linear distortion model can be derived as:

$$\frac{\hat{R}}{M_w} = a\bar{q}^{-1} + b\bar{q}^{-2} \tag{5.5}$$

$$\hat{D}_w = a'\bar{q} + b' \tag{5.6}$$

where $\hat{R}$ is the estimated number of bits and $\hat{D}_w$ is the estimated weighted distortion, $\overline{q}$ is the average quantization level for a frame, and a, b, a' and b' are the model parameters.

5.2.1.2 Model Parameter Determination

The model parameters a, b, a' and b' in (5.5) and (5.6) can be determined using a linear regression method. After encoding each frame, the encoder can collect the following information for that frame: the actual number of bits used, the average quantization parameter, the weighted MAD and the weighted MSE. Then, a, b, a' and b' can be calculated. The R-D model with the updated parameters will be used in the next frame rate control. A statistical technique (least squares estimation) is adopted to estimate a, b, a' and b' on line.

For the distortion model (5.6), the square error to be minimized is:

$$\overline{S}(a', b') = \sum_{i=1}^{k} (D_{wi} - a'\overline{q}_i - b')^2 \tag{5.7}$$

where k is the number of selected past frames, $\overline{q}_i$ is the average quantization level for the i th frame, and D_{wi} is the weighted distortion for the i th frame.

The least squares estimators of a' and b' must satisfy

$$\frac{\partial \overline{S}}{\partial b'} = 0, \qquad \frac{\partial \overline{S}}{\partial a'} = 0 \tag{5.8}$$

Then, the formula to find the distortion model parameters can be easily derived:

$$b' = D_{w,k} - a'\overline{q}_k \tag{5.9}$$

$$a' = \frac{\sum_{i=1}^{k} \bar{q}_i D_{w,i} - (\sum_{i=1}^{k} \bar{q}_i)(\sum_{i=1}^{k} D_{w,i})/k}{\sum_{i=1}^{k} \bar{q}_i^2 - (\sum_{i=1}^{k} \bar{q}_i)^2/k} \tag{5.10}$$

The rate model (5.5) is a quadratic model. Its parameters also can be estimated using a linear regression method [8] since the equation is linear in parameters a and b by rearranging (5.5) as

$$\frac{\hat{R}}{M_w}\bar{q} = a + b\bar{q}^{-1} \tag{5.11}$$

The square error to minimize is:

$$\bar{S}(a,b) = \sum_{i=1}^{k} (\frac{R_i}{M_{wi}}\bar{q}_i - b\bar{q}_i^{-1} - a)^2 \tag{5.12}$$

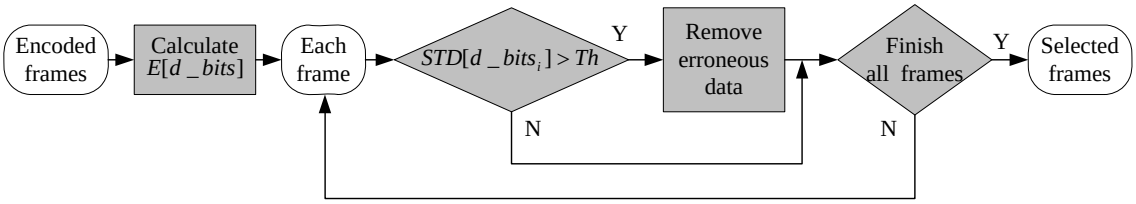
Taking the partial derivatives of $\bar{S}$ with respect to a and b leads to

$$a = \frac{\sum_{i=1}^{k} \bar{q}_i \left(\frac{R_i}{M_{w,i}} - b\bar{q}_i^{-1}\right)}{k} \tag{5.13}$$

$$b = \frac{k\sum_{i=1}^{k} \frac{R_i}{M_{w,i}} - (\sum_{i=1}^{k} \bar{q}_i^{-1})(\sum_{i=1}^{k} \bar{q}_i \frac{R_i}{M_{w,i}})}{k\sum_{i=1}^{k} \bar{q}_i^{-2} - (\sum_{i=1}^{k} \bar{q}_i^{-1})^2} \tag{5.14}$$

Equations (5.9),(5.10),(5.13) and (5.14) are used for updating parameters a, b, a' and b' after encoding each frame.

5.2.1.3 Statistical Removal of Data Outliers



d_bits_i: prediction error between the actual bit rate and the target bit rate of the i th frame. $d\_bits_i = t\arg etbit_i - actualbit_i$

$E[d\_bits]$: Average error of the selected k frames. $E[d\_bits] = \frac{1}{k}\sum_{i=1}^{k} d\_bits_i$

$STD[d\_bits_i]$: standard deviation of prediction error.

$STD[d\_bits_i] = d\_bits_i - E[d\_bits]$

Figure 5.2. Scheme to remove data outliers

In order to use the linear regression method efficiently, it is better not to include erroneous data for estimation. The same outlier removal procedure is adopted to improve the model accuracy as what has been done in MPEG-4 video verification model version 10 [3]. The erroneous data are defined in a statistical sense, i.e., if the data whose prediction errors between the actual bit rate and the target bit rate are larger than one standard deviation. By removing the erroneous data, more representative data can be selected to update the model parameters. The scheme is described in Fig. 5.2.

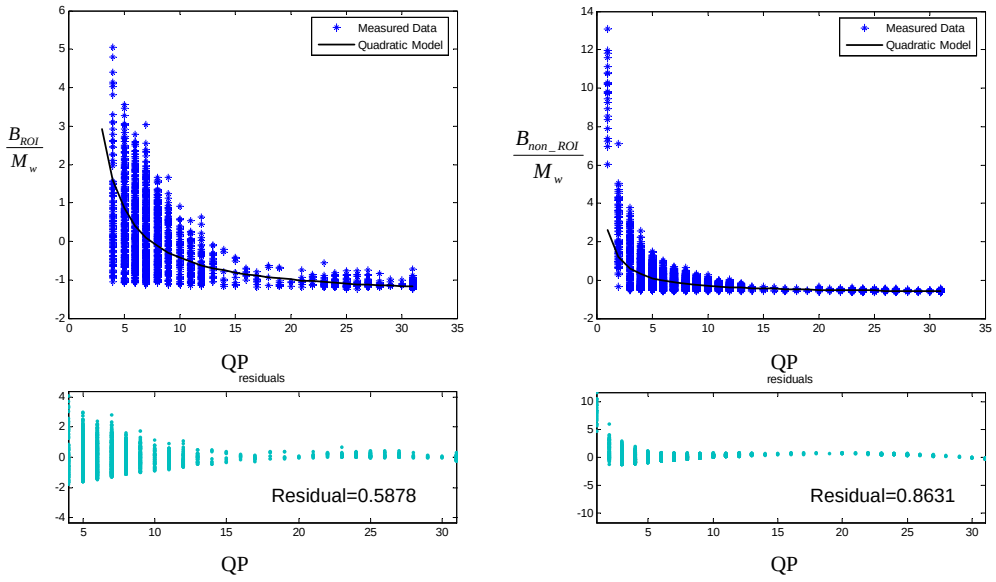
5.*2.2 Macroblock Layer Rate Modeling*

The study of rate modeling is extended to macroblock layer also. The bits-count over weighted residual for each macroblock as a function of QP is examined. Figure 5.3 shows the curve fitting for the QCIF Carphone sequence for both ROI macroblocks and NonROI macroblocks. The rate model can be written as:

$$B_{class} = (a_{class} QP_{class}^{-1} + b_{class} QP_{class}^{-2}) M_{w,class} \tag{5.15}$$

where a_{class} , b_{class} , B_{class} and $M_{w,class}$ are model parameters, number of bits and weighted residual respectively for each of the two classes: ROI and non_ROI.

It can be seen from Fig. 5.3 that the weighted bits for a MB vary widely with different macroblocks at same QP. It indicates that the macroblock layer rate model cannot properly model the required number of bits for any macroblocks. This is a consequence that statistics for each macroblock may vary widely, which disobeys the assumption of derivation of the quatratic rate model.



(a) (b)

Figure 5.3. Macroblock layer rate modeling (a) ROI macroblock rate model (b) Non_ROI macroblock rate model

The quadratic rate model is derived based on the assumption that source statistics is Laplacian distributed,

$$P(x) = \frac{\alpha}{2} e^{-\alpha|x|} \text{ , where } -\infty < x < \infty \tag{5.16}$$

The close form solution for the RD function can be derived as [49][51]

$R(D) = \ln(\frac{1}{\alpha D})$, where $D(x, \tilde{x}) = |x - \tilde{x}|$, and $D_{\min} = 0$, $D_{\max} = \frac{1}{\alpha}$

Then the RD function can be expanded into a Taylor series

$$R(D) = -\frac{3}{2} + \frac{2}{\alpha} D^{-1} - \frac{1}{2\alpha^2} D^{-2} + R_3(D) \tag{5.17}$$

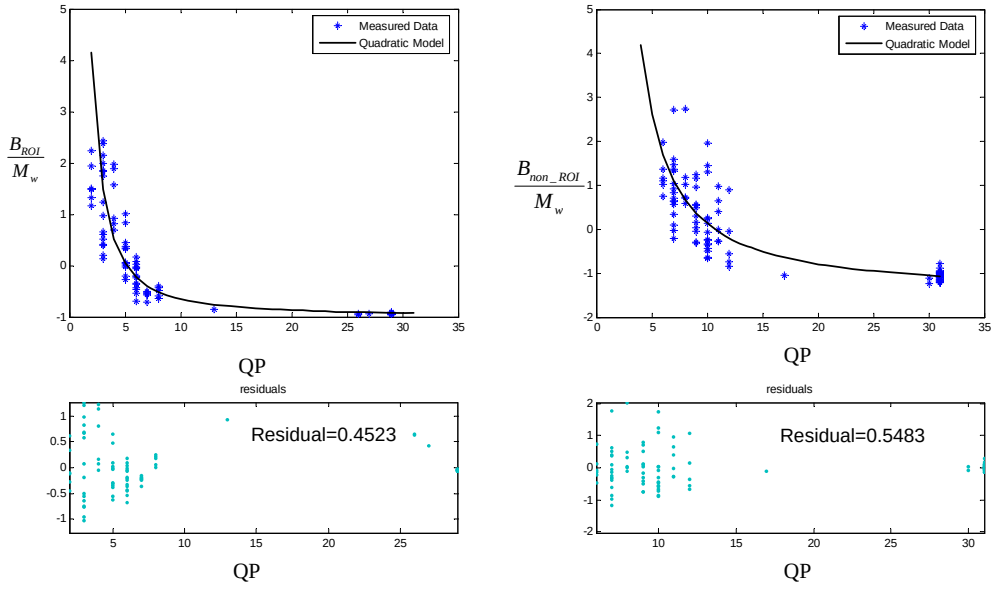
The quadratic model in (5.5) is derived by taking the linear term and the quadratic term in (5.17). The theoretical foundation of this rate model is based on the assumption that source is Laplacian distributed. And to accurately estimate the model parameters is based on the assumption that adjacent sources have similar statistics. However, these two assumptions are not correct for the macroblock layer data. The rate control algorithms based on this macroblock layer rate model suffers a lot from the model inaccuracy, such as macroblock layer rate control algorithm in VM8 [2][3].

5.2.3 VOP Layer Rate Modeling

Video object plane (VOP) is defined in MPEG-4 Visual as a video object at a particular point in time [9][12]. The introduction of VOP into ROI based coding in H.263 [10][14] allows more flexible analysis and manipulation of different video contents.

From Fig. 5.4, it can be seen that for the same QP, the weighted bits for a MB vary widely with different macroblocks. This is because the statistical property of each macroblock data can vary widely. Grouping macroblocks with similar statistical

property is a reasonable idea to compensate this shortcoming. In ROI based coding, each ROI and NonROI can be viewed as such a group of macroblocks, i.e., the VOP in MPEG4. Rate modeling in such a group of macroblock layers can be also called a VOP layer rate modeling.



(a) (b)

Figure 5.4. VOP layer rate modeling (a) ROI rate model (b) Non_ROI rate model

The rates over weighted MAD with respect to the average quantization parameter of ROI and Non_ROI are examined separately. Following the similar process discussed in the previous two sections, the rate models for ROI and Non_ROI have been achieved. The modeling is shown in Fig. 5.3. It can be seen that quadratic rate models in VOP layer provide much better fitting of the measured data than the macroblock layer rate models do. It indicates, at an average point of view, more

accurate bit count estimation can be achieved from VOP layer than from the macroblock layer.

5.3 Summary

Building accurate rate models are the key issues for developing good rate control schemes. In this chapter, rate models in frame layer, VOP layer and macroblock layer have been analyzed. For frame layer rate control, quadratic rate model and linear distortion model are achieved by fitting the measured data in the least squares sense. For macroblock layer rate control, rate models in both VOP layer and macroblock layer have been studied. It shows that VOP layer rate model gives better fitting of the measured data than the macroblock layer rate model, because VOP layer rate model is based on the average statistics of VOP data. However, rate control based on macroblock layer rate model can provide more flexible bit rate adjustment for individual macroblock data. The design of rate control scheme based on these three rate models will be discussed in the next two chapters.

CHAPTER 6

CBR VIDEO RATE CONTROL

The rate control for real time constant bit rate (CBR) video is very challenging due to the strict requirements on low latency, small buffer, and CBR output [1][2]. For real-time communication applications, a maximum delay of 0.1s is specified [4]. Also the maximum buffer size is limited by time delay. So, the maximum buffer size is $M = r \times d = \frac{r}{10}$, where r is the target bit rate of CBR channel and d indicates the delay time. If the target frame rate $f = 10\,fps$ is chosen, the buffer size is then equal to $\frac{r}{f}$, which is the constant number of bits for each frame. The function of such a buffer is to absorb the variation of the bit rate from the encoder and output the bit stream at a constant bit rate to the channel. A buffer overflow indicates frame-skipping, and a buffer underflow indicates bits are wasted. In this case, the CBR means constant bit-count for each frame, instead of average bit-count over a number of frames in the applications that allow a higher delay. Hence, it is very challenging to design rate control scheme for real-time CBR video, as there is little room for variable bit count from frame to frame.

There is no frame layer rate control needed in rate control scheme for CBR video [4]. Moreover, the variation of bits-count for each frame should be very small.

Hence the estimation error caused by model accuracy is not allowed. Only operational R_D approach can be considered.

In this chapter, a joint VOP layer and macroblock layer rate control scheme for real-time CBR video is proposed.

6.1 VOP Layer Rate Control

6.1.1 Advantages of Rate Control at VOP Layer

In the traditional video coding standards, like H.261, H.263 and H.264, a video sequence is viewed as a collection of rectangular frames of video [6][13][11][12]. One of the key contributions of MPEG-4 Visual is to move away from this tradition [9][10]. Instead, it views a video sequence as a collection of video objects, which allows more flexible operations on different video contents [4]. By introducing VOP into ROI based H.263 rate control, more flexible bit allocation can be achieved between foreground and background.

The other advantage of doing rate control in VOP layer is: the macroblocks belonging to ROI and NonROI are grouped separately. Then macroblocks in a group can be viewed as a random variable with more smooth and uniform statistical properties than the individual macroblock data. This property can be used to compensate the inaccuracy of statistical analysis of macroblock layer data. In other words, the rate model built based on VOP statistical property can be used as a guidance to regulate the rate model built based on individual macroblock statistics. The VOP layer rate control scheme will be presented in the following section.

6.1.2 Rate Control at VOP Layer

6.1.2.1 VOP Layer Bit Allocation

The target number of bits for a frame can be allocated among different VOPs based on their coding complexity and perceptual importance. The distribution of bit budget is proportional to the square of the human visual weighted residual of a VOP. Let $T_{k,i}$ be the target number of bits allocated for the i th VOP in the k th frame, and R_k is the target number of bits for the k th frame, $M_{w,i}$ is the weighted MAD for the i th VOP, assuming there are N VOPs in the k th frame.

$$T_{k,i} = R_k \times \frac{M_{w,i} \cdot M_{w,i}}{\sum_{j=i}^{N} M_{w,j} \times M_{w,j}} \tag{6.1}$$

Figure 6.1 shows the results for VOP layer bit allocation for different video sequences. For each sequence, the upper subplot shows the variation of weighted residual M_w with frame number, and the lower subplot shows the frame target bits (indicated by real line) and ROI target bits (indicated by dotted line) assigned for each frame. The frame target bits are determined by the TMN8 frame layer rate control, which is employed to provide near constant target number of bits for each frame. The small variation in target bits from frame to frame is caused by the feedback of buffer fullness [4]. The bit budget for ROI is determined by the VOP layer bit allocation. It can be easily observed from Fig.6.1. that the variation of ROI target bit allocation corresponds to the variation of weighted residual M_w from frame to frame. It indicates

the VOP layer bit allocation successfully assigns more bits to the human visual weighted 'important' region with high coding complexity.

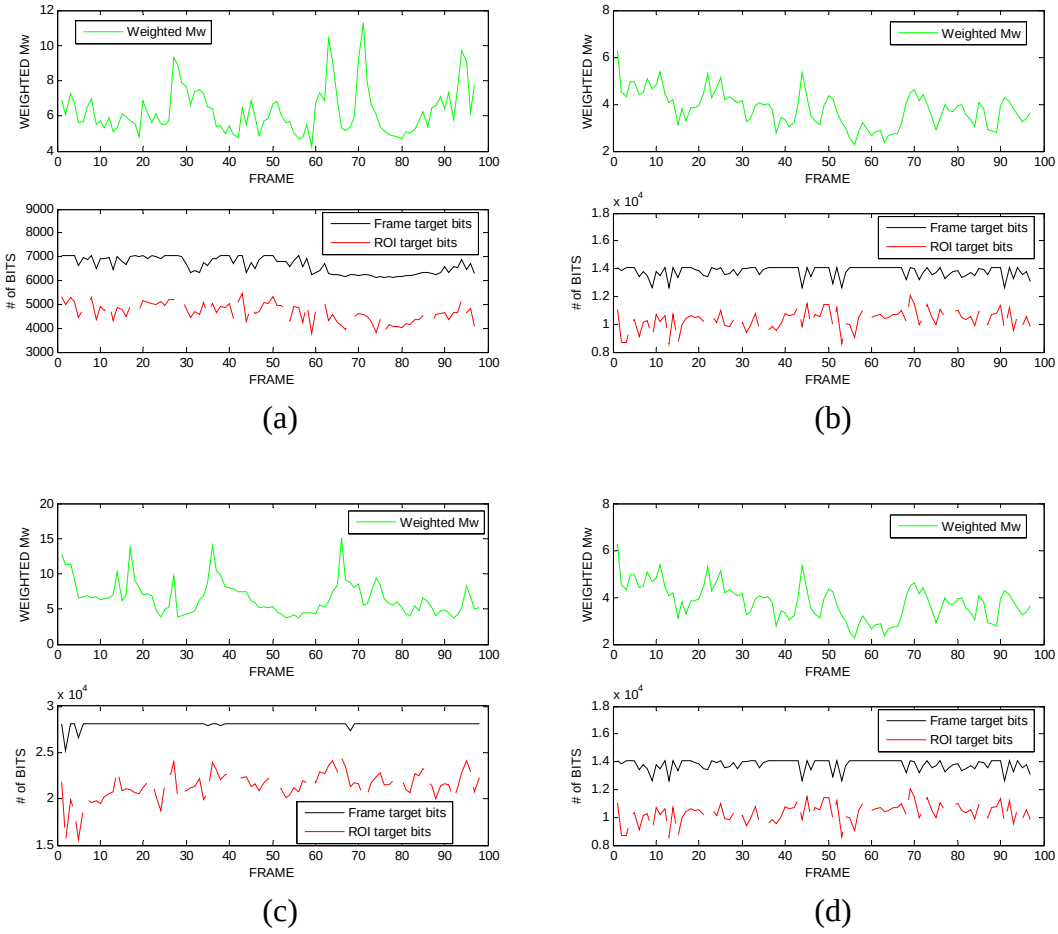


Figure 6.1. VOP layer bit allocation, weighted M_w is defined in (3.4). (a) QCIF Claire sequence at 64kbps (b) QCIF Akiyo sequence at 128 kbps (c) QCIF Salesman sequence at 256kbps (d) QCIF Carphone sequence at 128kbps.

6.1.2.2 VOP Layer QP Determination

For each ROI and NonROI macroblock group, the following procedure is designed to determine the QP.

1) Achieve target bits $T_{k,i}$ for the i th VOP in the k th frame by (6.1).

2) Compute the required number of bits for the i th VOP.

$$R_i = (a_i QP_i^{-1} + b_i QP_i^{-2}) M_{w,i}$$

3) Adapt QP_i

If $\tau_L \times T_{k,i} < R_i < \tau_H \times T_{k,i}$, go to step 4;

Otherwise,

if $R_i < \tau_L \times T_{k,i}$, decrement QP_i by one, and go to step 2.

if $R_i > \tau_H \times T_{k,i}$ increment QP_i by one, and go to step 2.

where τ_L and τ_H are the lower and higher percentage bounds that restrict the current VOP bits-count.

4) Encoding the current frame.

5) Update model parameters by linear regression method.

6.1.2.3 VOP Layer Rate Model Parameter Determination

To accurately determine rate model parameters is an important task in model based rate control. Although linear regression is employed to achieve better model parameters for any specific video sequences, the choice of initial value remains a key problem. Using the same initial value for different video sequences will cause inaccuracy in model parameters for a large number of frames for some video sequences, which causes the rate control result appears errant.

In order to avoid the malfunction caused by inappropriate choice of initial value, the first two frames of a group of inter frames are encoded using fixed QPs, and then the initial model parameters for each VOP can be calculated. Choosing initial model

parameters by this way reduces the number of frames needed to adapt model parameters into appropriate values for any video sequences.

6.2 Macroblock Layer Rate Control

6.2.1 Advantages of Rate Control at Macroblock Layer

Although the VOP layer rate control gives a solution of QP value for each VOP macroblock group, encoding the entire VOP with this QP cannot accurately achieve the target bit-counts. This is because the rate model is based on the average statistics of a group of VOP macroblocks. Therefore, QP determination at a further precise level is needed.

6.2.2 Highlights of The Proposed Macroblock Layer Rate Control

To balance the inaccurate use of average statistics and the inaccuracy of using individual statistics which may be not uniform for the whole group, several new features are designed for the proposed macroblock layer rate control.

1), The quadratic model used in the proposed rate control is a modified version of the quadratic model in VM8. The individual MB is replaced with the group of unencoded MBs in the current VOP.

2), The QP achieved for each VOP is used as a guidance to regulate the QP value from macroblock layer in the current VOP group. When the macroblock layer rate model is applied, the first 5-10% macroblocks are used to adapt the rate parameters to appropriate values. The guidance of QP from VOP layer is very important for these macroblocks.

3), For the rest macroblocks, to regulate the QP in a small range, the average QP of all the encoded macroblocks in the current VOP is used, so that to achieve higher precise level rate control and to make use of the average model statistics are considered at the same time.

6.2.3 Determining QP for Each Macroblock

In the macroblock layer rate control scheme, the major task is to determine QP for each macroblock so that the rate constraint $\tau_L \times T_i < R_i < \tau_H \times T_i$ for each VOP can be satisfied. The proposed macroblock layer rate control consists of the following steps:

1) Initialize QP as the QP value from the previous macroblock.

2) Update the weighted residual based on the uncoded macroblocks.

3) Calculate the estimated bits-count for the uncoded macroblocks using the rate model, and calculate the residual bits-count by subtracting the estimated bits-count and the used bits-count from the target VOP bits-count T_i.

4) If the QP value satisfies the rate constraint, then go to step 7.

5) Otherwise, if $R < (\tau_L \times T_i)$, decrement QP by 1, and repeat step 3.

6) Otherwise, if $R > (\tau_H \times T_i)$, increment QP by 1, and repeat step 3.

7) Modify QP based on the VOP layer QP or the average QP of the encoded MBs in the current VOP group. The reference QP is denoted by $\overline{QP}$.

If QP < $\overline{QP}$, then $QP = \max(\overline{QP} - 2, QP)$

Else $QP = \min(\overline{QP} + 2, QP)$

8) Encode the current macroblock.

9) Update model parameters by linear regression method [8].

$$b_{class} = \frac{k\sum_{i=1}^{k}\frac{R_i}{M_{w,i,class}} - (\sum_{i=1}^{k}QP_{class,i}^{-1})(\sum_{i=1}^{k}QP_{class,i}\frac{R_i}{M_{w,i,class}})}{k\sum_{i=1}^{k}QP_i^{-2} - (\sum_{i=1}^{k}QP^{-1})^2} \tag{6.2}$$

$$a_{class} = \frac{\sum_{i=1}^{k}QP_{class,i}\frac{R_i}{M_{w,i,class}} - b_{class}QP_{class,i}^{-1}}{k} \tag{6.3}$$

The modified quantization mode in Annex T of H.263 standard [6] is adopted to achieve more flexible change in QP at the macroblock level. For small changes in QP among adjacent MBs, Annex T still uses two bits in DQUANT, which does not increase the overhead bit count than the normal mode. For a large change in QP, this mode will signal any new value for QUANT using six bits. At the expense of extra bits used in this case, a flexible change in QP can be achieved.

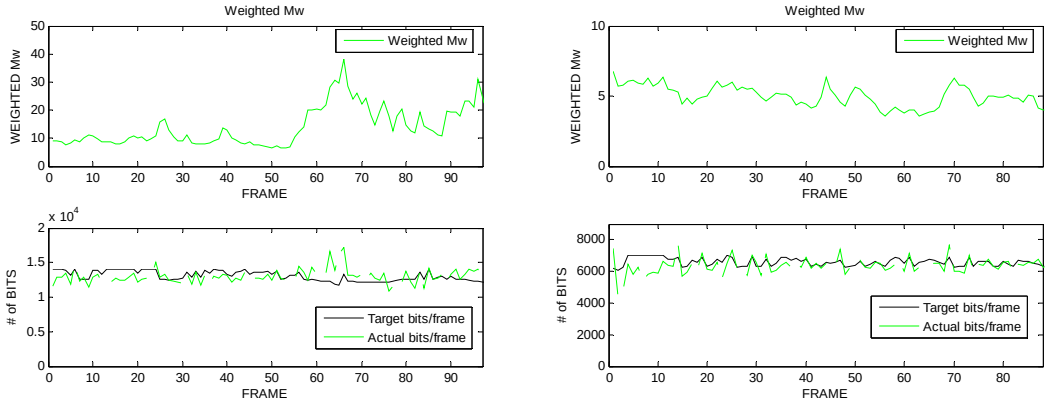
6.2.4 Simulation Results

Several groups of simulations have been conducted to test the efficiency of the proposed rate control algorithm and the test sequences are available at website [63] [64].

Figure 6.2. shows the comparison of the target bit rate with the actual bit rate for two video sequences, as well as the variation of weighted residual M_w for each sequence. Two observations can be made from this group of simulation results: the actual bits can be very close to the target bits. Fig. 6.2. (a) shows the results for QCIF Carphone sequence, the target bit rate is 128kbps while the average actual bit rate is

128.13kbps, and (b) shows the result for QCIF Akiyo sequence, the target bit rate is 64kbps and the average actual bit rate is 63.62kbps.

The other observation is that the variation of the actual bits corresponds to the variation of the weighted MAD. The relatively big differences between target bit rates and actual bit rates are due to the appearance of unfamiliar data statistics. Like when a sharp peak in weighted MAD appears, the rate model is less accurate and it makes the actual bit rate differ from the target bit rate.



(a) (b)

Figure 6.2. Comparison of target number of bits/frame and actual number of bits/ frame (a) QCIF Carphone sequence (b) QCIF Akiyo sequence

Figure 6.3. depicts the target bit rate achievement at both frame layer and VOP layer. (a) shows the results for QCIF Carphone sequence at target bit rate 256kbps, (b) for QCIF Akiyo sequence at target bit rate 128kbps (c) for QCIF Claire sequence at target bit rate 64kbps and (d) for QCIF Salesman sequence at target bit rate 128kbps. It can be seen that the proposed joint VOP layer and MB layer rate control algorithm can achieve a smooth bit rate close to the target for both ROI and NonROI and the entire frame.

The qualities of the output sequences are compared with conventional TMN8 rate control as a baseline. The PSNR comparisons are shown in Table 6.1. It can be observed that the PSNR for ROI is higher at the cost of lower PSNR for non_ROI regions. The perceptual quality improvement is shown by Fig. 6.4. When the same bit rate is applied for both the baseline and the proposed algorithm, side by side perceptual quality improvement over the baseline can be observed.

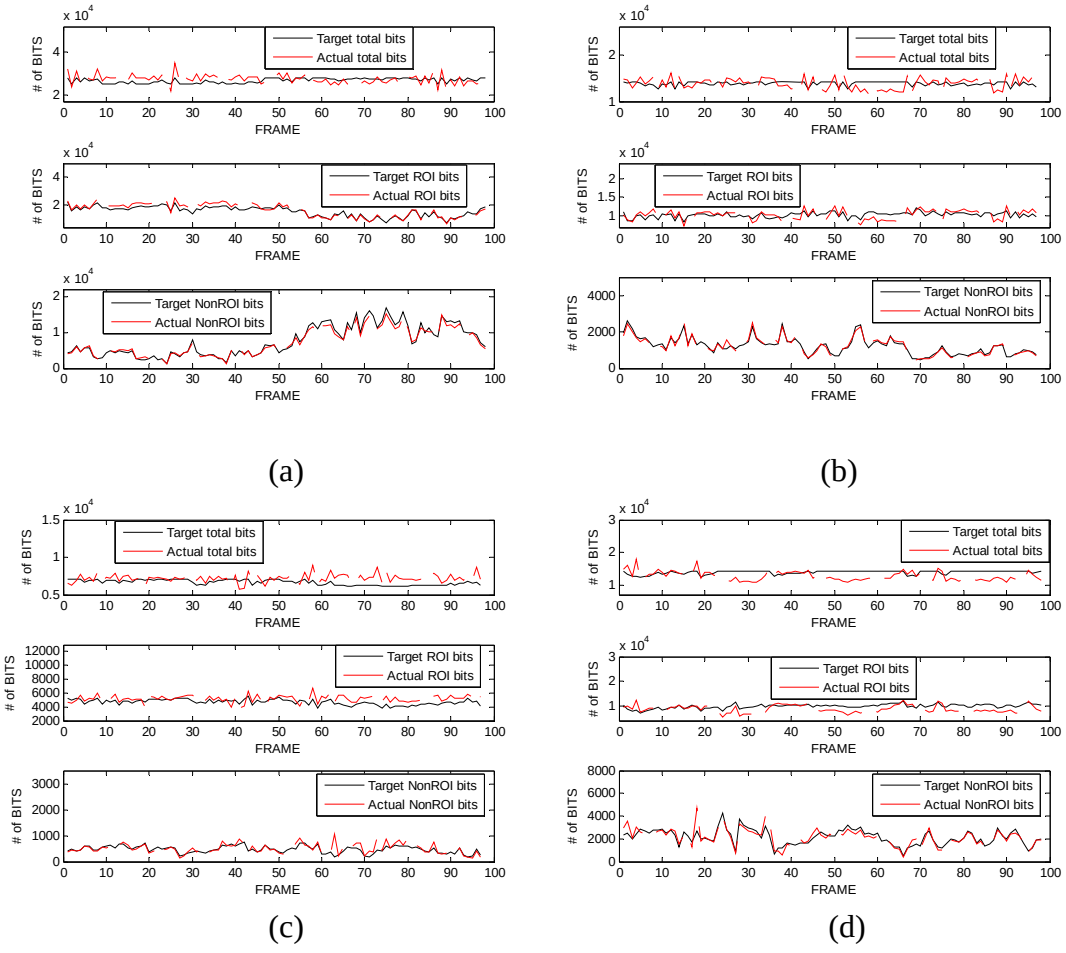
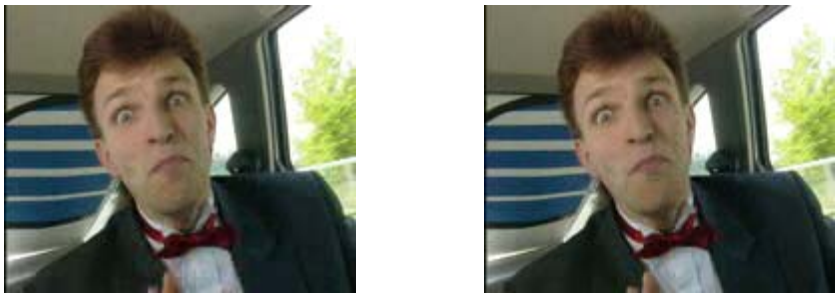


Figure 6.3. Comparison of target number of bits and actual number of bits for ROI and NonROI (a) QCIF Carphone sequence (b) QCIF Akiyo sequence (c) QCIF Claire sequence (d) QCIF Salesman sequence



(a) (b)

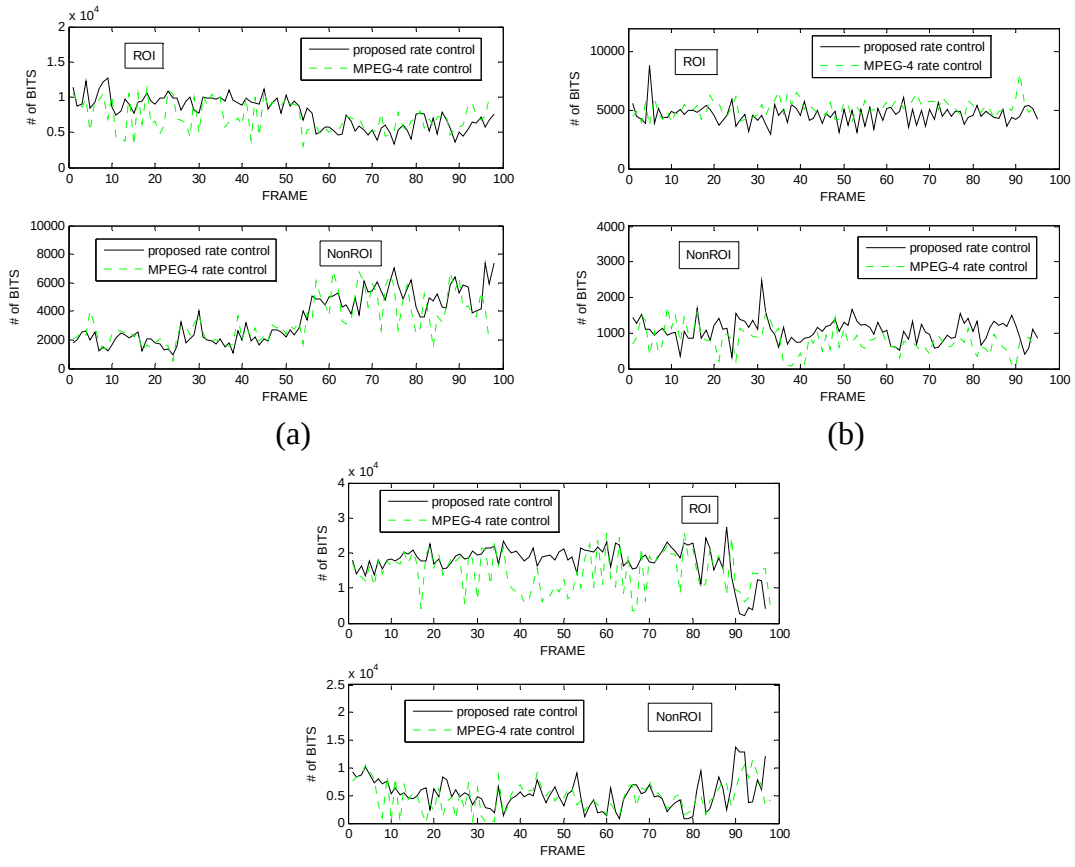
Figure. 6.4. Perceptual improvement for Carphone sequence (a) proposed rate control at 256kbps (b) TMN8 rate control at 256kbps

Table 6.1. PSNR(dB) comparison between TMN8 baseline rate control and proposed CBR video rate control for Carphone, Akiyo, Claire and Salesman sequences

	Carphone 256kbps		Akiyo 128kbps	
	TMN8	CBR_RC	TMN8	CBR_RC
ROI	39.71	45.20	40.83	42.48
NonROI	40.13	37.36	45.54	42.44
	Claire 64kbps		Salesman 128kbps	
	TMN8	CBR_RC	TMN8	CBR_RC
ROI	37.67	40.36	39.20	45.52
NonROI	43.03	38.40	40.54	36.43

To evaluate the proposed ROI based rate control scheme, the comparison with object based rate control scheme in MPEG-4 VM 8 has been made. Figure 6.4 shows the comparison of the actual bits used for ROI and NonROI between the proposed rate

control scheme and the VM8 rate control. It can be observed that smoother bit rate can be achieved using the proposed rate control. Table 6.2 compares the average bit deviation in detail. And table 6.3 provides the comparison of PSNR of the proposed rate control and the MPEG-4 VM8 rate control. It can be seen that less bit deviation and higher PSNR are achieved by the proposed rate control when comparing with the object based VM8 rate control.



(a) (b)

(c)

Figure 6.5 Comparison of encoded bits for ROI and NonROI with VM8 rate control. (a) QCIF Carphone sequence at 128 kbps (b) QCIF Salesman sequence at 64 kbps (c) QCIF Forman sequence at 256 kbps

Table 6.2. Average bit deviation comparison between MPEG4 VM8 rate control and proposed CBR video rate control.

Video sequence	Target bit rate (kbps)	VO	Bit deviation (bits)		
			MPEG4	CBR	Improvement
Carphone	128	Frame	1901	843	55.65%
		ROI	1486	527	64.54%
		NonROI	461	456	1.11%
Salesman	64	Frame	1231	677	45.00%
		ROI	1275	522	59.06%
		NonROI	247	148	40.08%
Foreman	256	Frame	6068	3443	43.26%
		ROI	3968	3016	23.99%
		NonROI	1855	538	71.00%

Table 6.3. PSNR(dB) comparison between MPEG4 VM8 rate control and proposed CBR video rate control.

Video sequence	Target bit rate (kbps)	VO	PSNR (dB)		
			MPEG4	CBR	Improvement
Carphone	128	ROI	38.01	39.55	1.54
		NonROI	34.47	34.85	0.38
Salesman	64	ROI	35.17	37.18	2.01
		NonROI	32.43	33.72	1.29
Foreman	256	ROI	39.09	39.86	0.77
		NonROI	34.06	34.31	0.25

6.3 Summary

This chapter has proposed a joint VOP layer and macroblock layer rate control algorithm for real time CBR video. The concept of VOP is borrowed from MPEG-4 Visual [9][15] to achieve more flexible rate control among VOPs. VOP layer rate control consists of bit allocation among VOPs and average QP determination for each VOP. Macroblock layer rate control is needed since it can provide model accuracy at a more precise level. The benefits of using average statistics from VOP layer and using individual statistics from macroblock layer are balanced in the proposed joint rate control algorithm.

The average QP for each VOP is used as a guidance to regulate the QP determined by the macroblock layer rate control. For the first 5-10% macroblocks, the guidance of QP from VOP layer is very important, since these macroblocks are needed to adapt the MB layer rate parameters to appropriate values. For the rest of the macroblocks, to regulate the QP in a small range, the average QP of all the encoded macroblocks in the current VOP is used, so as to achieve higher precise level rate control and to make use of the average model statistics at the same time. The proposed joint VOP layer and MB layer rate control can closely achieve target bit rates for various video sequences. The PSNR improvements on ROI have also been obtained.

CHAPTER 7

VBR VIDEO RATE CONTROL

7.1 System Structure

Many VBR rate control schemes have been developed under an unconstrained variable bit rate (UVBR) assumption [10] [11] i.e., assuming there are sufficient buffers at both the encoder side and the decoder side. Then the rate control problem can be formulated as an optimization problem constrained by the bit budget only. Although it does not aim at optimal rate control solution under multiple channel constraints, it does provides a good R-D tradeoff operating point for a given target bit rate. The proposed rate control scheme designed for VBR video is also under this UVBR scenario.

The block diagram of the proposed rate control scheme is illustrated in Fig. 7.1. It is developed for a H.263 [14][23] compatible video codec. Four components are added for the proposed rate control scheme. ROI detection for intra frame and ROI tracking for inter frame are applied to segment ROI from background, which are discussed in chapter 4. Based on this segmentation information, a frame layer rate control to decide the target bit budget for the current frame and a macroblock layer rate control to determine the quantization parameters (QP) for ROI macroblock and Non-ROI macroblock respectively are proposed. With these QPs, the DCT coder encodes the current frame, and sends the relevant information to the rate control scheme for

parameter adaptation. At the end, the efficient frame-skipping algorithm in TMN8 and VM5 is adopted as a component in our overall scheme to control the frame rate (see section 2.2.2).

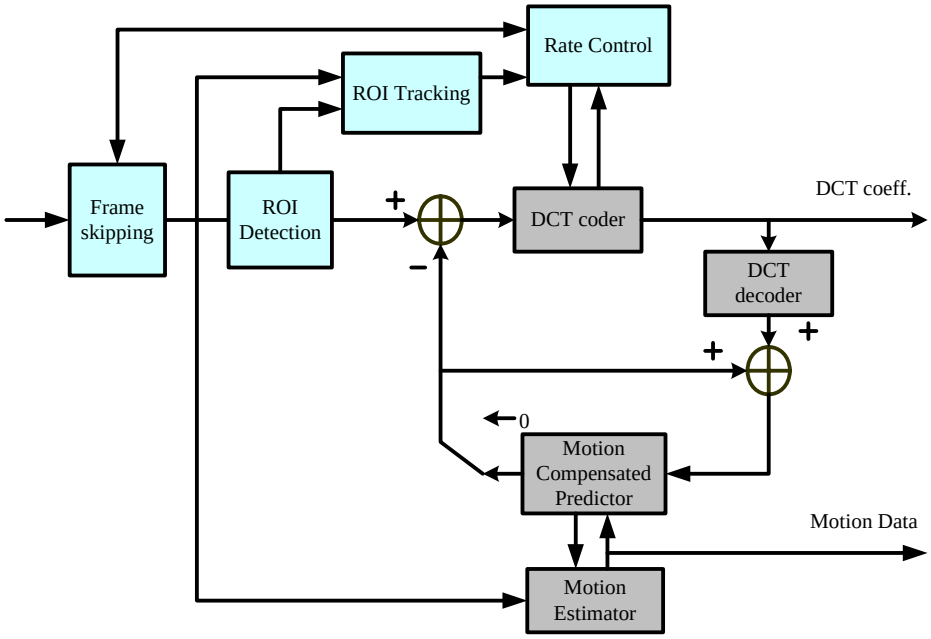


Figure 7.1. Block diagram of rate control for H.263 compatible codec

7.2 Frame Layer Rate Control

7.2.1 Problem Formulation

Different video frames have different complexities regarding the degree of motion. If a constant target number of bits is assigned for each frame, the video quality degradation of the fast changing frame is unavoidable. If the channel allows variable bit rate, dynamically adjusting the target number of bits for each frame based on the complexity is always preferred. This is the objective of frame layer rate control, which is required to efficiently encode VBR video.

Under a UVBR scenario, the frame layer rate control can be formulated as: find the right $\overline{q}_k$ to minimize the weighted distortion (5.1) for the current frame, subject to

, where and are the number of encoded frames and the target bit rate in a GOP, and is the number of bits for the th frame.

The Lagrange multiplier method is widely used as a tool to solve such a constrained optimization problem [38] [58] [59]. The idea is to embed the constraint into the objective function to be minimized, and then the problem can be stated as an unconstrained problem:

(7.1)

where is the residual number of bits of the th frame, and is the weighted distortion for the th frame.

Assuming that a given bit rate is assigned to every frame equally, and

(7.3)

is the Lagrange multiplier, a weighting factor of the trade-off between the rate and the distortion. In the optimization problem, the optimal solution is a function of λ. A critical step is to appropriately select λ. Choosing λ can be viewed as determining the appropriate trade-off between the rate and the distortion. If the rate budget is tight, then a higher weighting of the rate constraint needs to be considered for the next frame optimization. If the rate budget is plentiful, then lower weighting factor

can be applied for the next frame optimization. In the proposed system, the weighting factor λ is adapted based on the residual bits. The optimization procedure is explained in the following section.

7.2.2 Optimization Procedure

1) Optimization with R-D model:

Penalty function for the k th frame:

$$P_k(\bar{q}_k) = D_{w,k} + \lambda_k \max(0, \hat{B}_k^{res}) \tag{7.4}$$

Applying the rate and distortion model defined in (5.5) and (5.6),

$$P_k(\bar{q}_k) = a'\bar{q}_k + b' + \lambda_k \cdot \max\left[0, \sum_{i=1}^{k-1} R_i + (a\bar{q}_k^{-1} + b\bar{q}_k^{-2})M_w - k\frac{B_{gop}}{N_{gop}}\right] \tag{7.5}$$

The optimal $\bar{q}_k$ to minimize the penalty function can be determined by

(7.6)

Under the assumption that is convex, the fast and simple interval cutting algorithm is used to solve (7.6). The interval cutting algorithm can be explained as follows.

Require: range_a, range_b and precision ε.

Set = range_a, = range_b

Repeat

If ,

Else,

End if

until

output:

2) Lagrange multiplier adaptation: [6]

The Lagrange multiplier is updated based on the residual bits after coding a frame, in order to achieve appropriate trade-off between the rate and the distortion. The adaptation algorithm is as follows:

(7.7)

where is the updated Lagrange multiplier for the th frame, and

(7.8)

(7.9)

(7.10)

3) Update model parameters using linear regression analysis

For the distortion model (5.6)

$$b' = D_{w,k} - a'\overline{q}_k \qquad a' = \frac{\sum_{i=1}^{k} \overline{q}_i D_{w,i} - (\sum_{i=1}^{k} \overline{q}_i)(\sum_{i=1}^{k} D_{w,i}) / k}{\sum_{i=1}^{k} \overline{q}_i^2 - (\sum_{i=1}^{k} \overline{q}_i)^2 / k}$$

For the rate model (5.5)

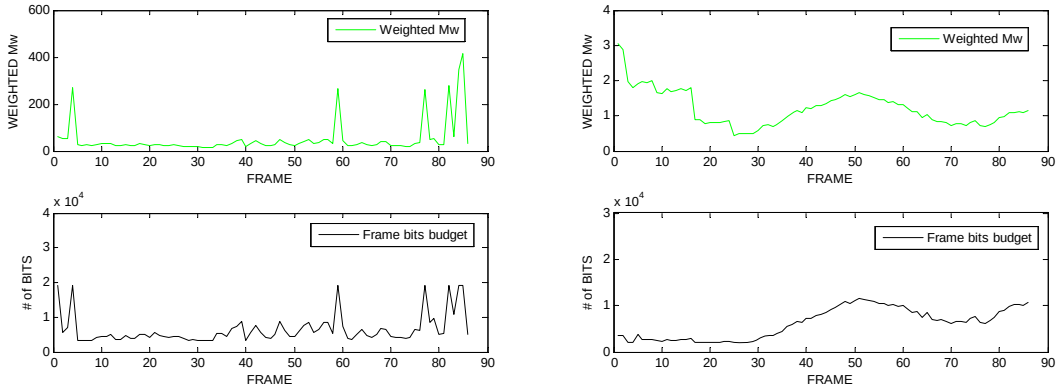
$$b = \frac{k\sum_{i=1}^{k}\frac{R_i}{M_{w,i}} - (\sum_{i=1}^{k}\overline{q}_i^{-1})(\sum_{i=1}^{k}\overline{q}_i \frac{R_i}{M_{w,i}})}{k\sum_{i=1}^{k}\overline{q}_i^{-2} - (\sum_{i=1}^{k}\overline{q}_i^{-1})^2} \qquad a = \frac{\sum_{i=1}^{k}\overline{q}_i \frac{R_i}{M_{w,i}} - b\overline{q}_i^{-1}}{k}$$

4) Calculate the target bit budget R_k for the th frame using rate model (5.5).

5), Lower and upper bound are applied to constrain the target bit budget R_k .

(7.11)

(7.12)



(a) (b)

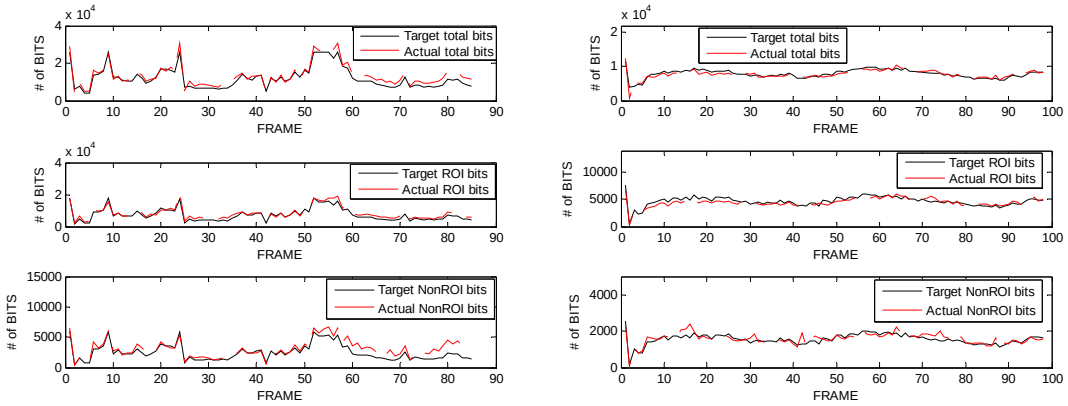
Figure 7.2 Frame layer bit allocation (a) QCIF Foreman sequence at 96kbps (b) QCIF Claire sequence at 64kbps.

The simulation results of frame layer rate control are shown in Fig. 7.1. For each testing sequence, the upper subplot shows the variation of weighted residual M_w with frame number, and the lower subplot shows the assigned frame bits budget. It can be easily seen from Fig.7.1. that the variation of the frame bits budget corresponds to the variation of weighted residual M_w. It indicates the frame layer rate control assigns

more bits to the frames with higher coding complexity and higher human visual importance.

7.3 VOP and Macroblock Layer Rate Control

The frame layer rate control algorithm determines the bit budget for each frame. Then the VOP layer and macroblock layer rate control algorithms proposed in chapter 6 have been applied. The VOP layer rate control distributes the bit budget among foreground and background based on their coding complexity and visual importance. The macroblock layer rate control is performed to adjust QP at a MB by MB basis.



(a) (b)

Figure 7.3 Comparison of target number of bits and actual number of bits for ROI and NonROI (a) QCIF Carphone sequence at 128kbps (b) QCIF Claire sequence at 96kbps

The simulation results for the joint frame layer and VOP layer and MB layer rate control scheme for VBR video are shown in Fig. 7.2. (a) shows the simulation results of QCIF Foreman sequence at 128kbps, and the average actual bit rate is 115kbps. (b) shows the simulation results of QCIF Claire sequence at 96kbps, and the average actual bit rate is 78kbps. The dotted line indicates the target number of bits per

frame, which is determined by the frame layer rate control, and the solid line represents the actual number of bits per frame. It can be seen that the target bit rate is well achieved at both the frame layer and each VOP layer. The PSNR comparison of ROI and non_ROI with TMN8 baseline rate control are compared in Table 7.1.

Table 7.1 PSNR (dB) comparison between TMN8 baseline rate control and proposed VBR video rate control.

	Carphone 128kbps		Foreman 96kbps	
	TMN8	VBR_RC	TMN8	VBR_RC
ROI	35.43	40.02	33.98	36.81
NonROI	36.82	35.01	33.52	31.00

7.4 Summary

A joint frame layer, VOP layer and macroblock layer rate control scheme for VBR video is presented in this chapter. Frame layer rate control is needed for efficiently encoding the VBR video. It has been formulated as a constrained optimization problem, and Lagrange multiplier method was used to solve it. The Lagrange multiplier was adapted based on the residual number of bits after coding each frame. The VOP layer and macroblock layer rate control scheme proposed in the previous chapter is employed after the frame layer rate control to determine bits budget for each VOP and QP for each macroblock. The performance of the joint rate control scheme is good at both the target bit rate achievement and the ROI PSNR improvement.

CHAPTER 8

CONCLUSIONS AND FUTURE WORK

8.1 Conclusions

ROI based rate control for low bit rate H.263 [6] [10] compatible video has been studied in this dissertation. Based on the different requirements of CBR video and UVBR video, two rate control schemes are proposed. For the CBR video, only VOP layer and macroblock layer rate control are needed due to the strict buffer constraint. An operational R-D approach is applied to achieve the suboptimal solution. For the VBR video, rate control scheme included a frame layer scheme, a VOP layer scheme and a macroblock layer scheme. An operational model-based approach is applied to solve the R-D optimization problem, and the Lagrange multiplier is adapted based on the residual number of bits after coding each frame. In both cases, a modified quadratic rate model is proposed at the macroblock layer: using a group of un-encoded MBs in the current VOP instead of using individual MB to achieve the rate model and update model parameters. Based on this proposed rate model, some new features are designed to use both average statistics of a VOP and individual statistics of MB and at the same time to achieve more efficient rate control performance.

The proposed rate control scheme is ROI based. Hence an accurate ROI segmentation scheme is very critical to the system performance. A new macroblock

layer face detection and tracking system for real time H.263 video is proposed to preprocess the video sequence. To reduce the computational complexity, a face detection algorithm is designed for intra frame and a face tracking algorithm is designed for inter frame using only the motion vectors.

In the intra frame detection scheme, skin color detection and mosaic-based detection are combined to achieve good detection accuracy. By applying these two schemes in H.263, the detection complexity is significantly reduced. In the inter frame face tracking scheme, not only the accumulated motion vector over a number of frames but also the motion vectors of each frame are considered to achieve a high tracking accuracy.

The performance of the proposed ROI based rate control is compared with conventional TMN8 rate control [4] [6] as a baseline, and object based VM8 rate control [3] as well. The comparison shows that the proposed algorithm can achieve more accurate bit rate and better PSNR for ROI by using several video sequences.

8.2 Future Work

Some ideas on future extension of this work are listed as follows:

To further improve the detection accuracy in intra frame, segmentation methods like level sets [47] and graph cuts [48] can be used between the skin color detection stage and the mosaic rule based detection stage. Spatiogram based tracking algorithms [37] can be considered in the face tracking stage for the applications that do not have strict requirements in computational complexity.

How to improve the performance of the ROI based rate control (like a neat buffer control for CBR, an accurate target bit achievement) is still a challenging task in the future research. To extend this research to other block-based video coding standards [62] [67] [68] with appropriate modification will be an interesting future work.

APPENDIX A

GLOSSARY OF ACRONYMS

1-D	one-dimensional
2-D	two-dimensional
ARTSP	advanced real-time simple profile
ASP	advanced simple profile
ASSP	acoustics, speech and signal processing
BAB	binary alpha block
BAE	binary arithmetic encoding
CBR	constant bit rate
DCT	discrete cosine transform
CIF	common intermediate format
CSVT	circuits and systems for video technology
CVPR	computer vision and pattern recognition
DP	Dynamic programming
DPCM	differential pulse code modulation
ICIP	international conference on image processing
IEEE	institute of electrical and electronics engineers
IP	image processing
ISCAS	international symposium on circuits and systems
ITU-T	international telecommunication union – telecommunication sector
QCIF	quarter CIF
GOP	group of pictures

JSAC	journal on selected areas in communications
MAD	mean absolute difference
MB	macroblock
MPEG	moving picture experts group
MSE	mean square error
MV	motion vectors
ORD	operational rate distortion
PAMI	pattern analysis and machine intelligence
QCIF	quarter common intermediate format
QP	quantization parameters
RC	rate control
RD	rate distortion
ROI	region of interest
SA-DCT	shape adaptive DCT
SP	simple profile
SPIE	society of photo-optical and instrumentation engineers
SRC	scalable rate control
TM5	Test Model 5
TMN8	Test Model Near-term Version 8
UVBR	unconstrained variable bit rate
VBR	variable bit rate
VCIP	visual communication and image processing

VHS	video home system
VLC	variable-length code
VM8	Verification Model Version 8
VO	video object
VOP	video object plane
YCbCr	color coordinate used in most digital video formats, consisting of luminance (Y) and two color difference signals (Cb and Cr)

APPENDIX B

ERROR CONTROL IN VIDEO COMMUNICATION

Error control is important in real time video communication due to the widely used compression methods and the stringent delay requirements. In the most popular video coding standards (H.26x and MPEGx) [1] [2] [9], temporal predictive coding and variable length coding (VLC) are extensively used by the encoder. A single error sample can lead to errors in the samples in the same and following frames because of the use of predictive coding. With VLC, the effect of a bit error may cause damage over a large portion of a video frame. Also, real time video communication is delay sensitive, so it is hard to make use of network protocols which use retransmission to ensure error-free delivery.

Therefore, error control is very challenging. In order to obtain successful video communication in the presence of errors, the error control mechanisms have been carefully designed at transport level, source encoder, source decoder or both encoder and decoder.

The transport-level error control is the most important part of error control. It provides a basic quality of service (QoS) level. Transport-level error control can be employed at the channel coder, packetizer and multiplexer and transport protocol levels.

Forward error correction (FEC) is a well known method for error detection and correction in channel coder. However, FEC is only effective for channels where single bit errors dominate. It is not very useful for transporting video over wireless networks or internet where burst error is usually longer than two bits.

Efficient packetization and multiplexing can reduce the effect of channel errors. For example, interleaved packetization can be used to prevent the loss of contiguous

blocks because of a single packet loss. Also because the picture header contains more important data, they should be heavily protected by FEC.

Although transport level error control can provide a basic QoS level, additional error control methods in other layers can further improve the robustness to transmission errors. There are several error resilient encoding methods that can produce a bit stream that is robust to transmission errors. For example, insert resynchronization markers to reduce the error damage by VLC, insert intra block or frames to stop temporal error propagation, and add data redundancy by generating and sending several bit streams of the same source signal over separate channels (multiple description coding) and so on.

To further improve the recovered video quality in presence of the transmission errors, error concealment is needed at the receiver side. Either spatial neighbor data or temporal neighbor data are used to interpolate the missing or damaged data. A number of algorithms are designed for error concealment [1].

It is obvious that if a backward channel from the decoder to the encoder is available, better performance can be achieved for error control. First, the coding parameter can be adapted based on channel conditions. For example, when a channel is very noisy, it is better to compress the source with lower quality while using more bits for error protection. Second, the reference picture can be selected based on feedback information, i.e., if a reference picture is heavily damaged during transmission, the encoder will make a decision to use other frames for the following temporal prediction. Also, there are a lot of techniques proposed for encoder-decoder interactive error control [1].

In the state of art coding standard H.264, several error control tools are designed as described above for transporting video in highly error prone environments. For example, intra block refreshing by RD control, SP/SI synchronization switching frame, error concealment by intra and inter picture interpolation and feed back channel error control. Details for these tools can be found in [9].

REFERENCES

[1] Y.Wang, J.Ostermann and Y.Q.Zhang, Video Processing and Communications, Upper Saddle River, NJ: Prentice Hall, 2001.

[2] K.R.Rao, Z.S. Bojkovic and D.A. Milovanovic, Multimedia Communication Systems: Techniques, Standards, and Networks, Upper Saddle River, NJ: Prentice Hall PTR, 2002.

[3] H.J.Lee, T.Chiang and Y.Q.Zhang, "Scalable rate control for MPEG-4 video", *IEEE Trans. CSVT*, vol.10, pp. 878-894, Sept. 2000.

[4] J.Ribas-Corbera and S.Lei, "Rate control in DCT video coding for low-delay communications", *IEEE Tran. CSVT*, vol.9, pp. 172-185, Feb. 1999.

[5] H. Song and C.-C.Jay Kuo, "A region-based H.263+ codec and its rate control for low VBR video", *IEEE Trans. Multimedia*, vol.6, pp. 489-500, June 2004.

[6] Video Codec Test Model, Near-Term, Version8 (TMN8), ITU-Telecommunication Standardization Sector, Portland, OR, June 1997.

[7] D. Hanselman and B. Littlefield, Mastering MATLAB6: A Comprehensive Tutorial and Reference, Upper Saddle River, NJ: Prentice Hall, 2001.

[8] J. Gross, Linear Regression, Berlin: Springer, 2003.

[9] I.E.G. Richardson, H.264 and MPEG-4 Video Compression: Video Coding for Next-generation Multimedia, Chichester, West Sussex, England, Wiley, 2003.

[10] K.R. Rao and J.J. Hwang, Techniques and Standards for Image, Video and Audio Coding, Upper Saddle River, NJ: Prentice Hall, 1996.

[11] I.E.G. Richardson, Video Codec Design: Developing Image and Video Compression Systems, Chichester, West Sussex, England, Wiley, 2002.

[12] M. Ghanbari, Standard Codecs: Image Compression to Advanced Video Coding, London, United Kingdom, The Institution of Electrical Engineers, 2003.

[13] ITU documents website, http://www.itu.int/ITU-T/publications/recs.html

[14] G. Cote et al, "H.263+: video coding at low bit rates", *IEEE Tran. CSVT*, vol.8, pp. 849-866, Nov. 1998.

[15] F. Pereira and T. Ebrahimi, The MPEG-4 Book, Upper Saddle River, NJ: Prentice Hall, 2002.

[16] L. Guan, S.Y. Kung and J. Larsen, Multimedia Image and Video Processing, Boca Raton, FL: CRC Press LLC, 2000.

[17] J.W.Lee and Y.S.Ho, "Target bit matching for MPEG2 Video Rate Control", *IEEE TENCON*, vol.1, pp.66-69, Dec, 1998.

[18] A. Ortega and K. Ramchandran, "Rate-distortion methods for image and video compression: an overview", *IEEE Signal Processing Magazine*, vol.15, pp.23-50, Nov, 1998.

[19] G.J. Sullivan and R.L. Baker, "Rate-distortion optimization for video compression", *IEEE Signal Processing Magazine*, vol. 15, pp.74-90, Nov, 1998.

[20] G.M. Schuster and A.K. Katsaggelos, "A video compression scheme with optimal bit allocation among segmentation, motion, and residual error", *IEEE Trans. IP*, vol.6, pp.1487-1501, Nov,1997.

[21] R. Neff and A. Zakhor, "Matching pursuit video coding at very low bit rates", *Data Compression Conference*, pp.411-420, March, 1995.

[22] A. Puri and R. Aravind, "Motion-compensated video coding with adaptive perceptual quantization", *IEEE Trans. CSVT*, vol.1, pp. 351-361, Dec. 1991.

[23] A. Eleftheriadis and A. Jacquin, "Model-assisted coding of video teleconferencing sequences at low bit rates", *IEEE ISCAS*, vol.3, pp. 177-180, June 1994.

[24] S.Sethuraman and R.Krishnamurthy, “Model based multi-pass macroblock-level rate control for visually improved video coding”, *Proceedings of Workshop and Exhibition on MPEG-4*, pp. 59-62, June 2001.

[25] S. Daly, K. Matthews and J. Ribas-Corbera, “Face-based visually-optimized image sequence coding”, *IEEE ICIP*, vol.3, pp.443-447, Oct. 1998.

[26] J. Hartung et al, “Object-oriented H.263 compatible video coding platform for conferencing application”, *IEEE JSAC*, vol. 16, pp. 42-55, Jan. 1998.

[27] D.Kiegman, M.H.Yang and N.Ahuja, “Detecting faces in images: a survey”, *IEEE Trans. PAMI*, vol.24, pp.34-58, Jan. 2002.

[28] G.Yang and T.S. Huang, “Human face detection in a scene”, *IEEE Trans. CVPR*, pp. 453-458, June 1993.

[29] H.P. Graf et al, “Locating faces and facial parts”, *Proc. First Int’l Wokshop on Automatic Face and Gesture Recognition*, pp.41-46, 1995.

[30] M.F. Augusteijn and T.L. Skujca, “Identification of human faces through texture-based feature recognition and neural network technology”, *Proc. IEEE Conf. Neural Networks*, pp.392-398, 1993.

[31] R.-L. Hsu, A.-M.Mohamed and A.K.Jain, “Face detection in color images”, *IEEE Trans. PAMI*, vol.24, pp.696-706, May 2002.

[32] I. Craw, D.Tock, and A.Bennett, “Finding face features”, *Proc. Second European Conf. Computer Vision*, pp.92-96, 1992.

[33] M. Kirby and L.Sirovich, “Application of Karhunen-Loeve procedure for the characterization of human faces”, *IEEE Trans. PAMI*, vol.12, pp.103-108, Jan, 1990.

[34] H.Wang and S.F. Chang, “A highly efficient system for automatic face region detection in MPEG video”, *IEEE Trans. CSVT*, vol. 7, pp. 615-628, Aug. 1997.

[35] D. Comaniciu, V. Ramesh, and P. Meer, “Kernel-based object tracking”, *IEEE Trans. PAMI*, vol. 25, pp.564-577, May 2003.

[36] S. Birchfield, “Elliptical head tracking using intensity gradients and color histograms”, *IEEE CVPR*, pp. 232-237, 1998.

[37] S.T. Birchfield and S. Rangarajan, “Spatiograms versus histograms for region-based tracking”, *IEEE CVPR*, vol. 2, pp.1158-1163, 2005.

[38] Y. Shoham and A. Gersho, “Efficient bit allocation for an arbitrary set of quantizers”, *IEEE Trans. ASSP*, vol. 36, pp. 1445-1453, Sept. 1988.

[39] O.Sukmarg and K.R.Rao, “Fast object detection and segmentation in MPEG compressed domain”, *IEEE TENCON*, vol.3, pp. 364-368, Sept. 2000.

[40] V.Mezaris et al, “Real-time compressed-domain spatiotemporal segmentation and ontologies for video indexing and retrieval”, *IEEE Trans. CSVT*, vol.14, pp. 606-621, May 2004.

[41] M.L.Jamrozik and M.H.Hayes, “A compressed domain video object segmentation system”, *IEEE ICIP*, vol.1, pp. 113-116, Sept. 2002.

[42] L. Favalli, A. Mecocci and F. Moschetti, “Object tracking for retrieval applications in MPEG-2”, *IEEE Trans. CSVT*, vol.10, pp. 427-432, April 2000.

[43] Encoding parameters of digital television for studios, Recommendation ITU-R BT.601-4, 1994.

[44] P. Trahanias and E.Skordalakis, “Syntactic pattern recognition of the ECG”, *IEEE Trans. PAMI*, vol.12, pp. 648-675, July 1990.

[45] I. Craw, H.Ellis, and J. Lishman, “Automatic extraction of face features”, *Pattern Recognition Letters*, vol. 5, pp.183-187, Feb, 1987.

[46] K.-K. Sung and T. Poggio, “Example-based learning for view based human face detection”, *IEEE Trans. PAMI*, vol. 20, pp.39-51, Jan, 1998.

[47] B. Raghunathan and S.T. Acton, “Image regimentation by level set analysis”, *IEEE Conference on Signals Systems and Computers*, Vol. 2, pp.916-920, 2000.

[48] N. Xu, R. Bansal and N. Ahuja, “Object segmentation using graph cuts based active contours”, *IEEE CVPR*, vol.2, pp.46-53, June, 2003.

[49] T. Chiang and Y.Q. Zhang “A new rate control scheme using quadratic rate distortion model”, *IEEE Trans. CSVT*, vol. 7, pp.246-250, Feb, 1997.

[50] A. Vetro, H. Sun and Y. Wang, “MPEG-4 rate control for multiple video objects”, *IEEE Trans. CSVT*, vol.9, pp.186-199, Feb, 1999.

[51] A. J. Viterbi and J.K. Omura, Principles of Digital Communication and Coding, New York: McGraw-Hill, 1979.

[52] A. K. Jain, Fundamentals of Digital Image Processing, Upper Saddle River, NJ: Prentice Hall, 1989.

[53] H. Song, Rate control algorithms for low variable bit rate video, Ph.D. Dissertation, University of Southern California, Los Angeles, CA, May 1999.

[54] Y. Yang, Rate control for video coding and transmission, Ph.D. Dissertation, Cornell University, Ithaca, NY, Jan. 2000.

[55] W.Ding and B.Liu, “Rate control of MPEG video coding and reordering by rate quantization modeling”, *IEEE.Trans. CSVT*, vol.6, pp. 12-20, Feb. 1996.

[56] K.H.Yang, A.Jacquin and N.S.Jayant, “Normalized rate-distortion model for H.263 compatible codecs and its application to quantizer selection”, *IEEE ICIP*, vol. 2, pp. 41-44, Oct. 1997.

[57] L.J.Lin, A.Ortega and C.-C.Jay Kuo, “Rate control using spline interpolated R-D characteristics”, *SPIE VCIP*, vol.2727, pp. 111-122, Mar. 1996.

[58] J.Choi and D. Park, “A stable feedback control of the buffer state using the controlled Lagrange multiplier method”, *IEEE Trans. IP*, vol. 3, pp. 546-558, Sept. 1994.

[59] T.Wiegand et al, “Rate-distortion optimized mode selection for very low bit rate video coding and the emerging H.263 standard”, *IEEE Trans. CSVT*, vol. 6, pp. 182-190, April 1996.

[60] ITU-T. Recommendation H.261: Video codec for audiovisual services at $p \times 64$ kbits, 1993.

[61] L. Lin and A. Ortega, “Bit-rate control using piecewise approximated rate-distortion characteristics”, *IEEE CSVT*, vol. 8, pp. 446-459, August 1998.

[62] Z. Lu, “Perceptual region-of-interest (ROI) based scalable video coding”, *JVT 15th meeting Busan*, JVT-O056, April, 2005.

[63] Video sequences website, http://www.stewe.org/vceg.org/sequences.htm

[64] Video sequences website, http://media.xiph.org/video/derf/

[65] MPEG website, http://www.chiariglione.org/mpeg/

[66] MPEG official website, http://mpeg.nist.gov/

[67] Z.G. Li et al, “Adaptive rate control for H.264”, *Journal of Visual Communication and Image Representation*, vol. 17, pp…, Jan, 2006.

[68] X. Yi and N. Ling, “Improved H.264 rate control by enhanced MAD-based frame complexity prediction”, *Journal of Visual Communication and Image Representation*, vol. 17, pp…, Jan, 2006.

[69] ITU-T. Recommendation H.263: Video coding for low bit rate communication, 1998.

BIOGRAPHICAL INFORMATION

Lin Tong was born in Beijing, China in 1975. She received the B.S. degree from University of Science and Technology of China, Hefei, China, in Automation Department, in 1999. From 1999 to 2000, she has been with the Legend Computer Company, Beijing, China, as a technical support engineer. She received the M.S. from Alfred University, NY, in Electrical Engineering in 2001. Since 2002, she has been working towards the Ph. D. at the University of Texas at Arlington. Her doctoral research is on rate control of video coding. She worked as an intern in Philips Research Center in 2003 for five months.