HARDWARE ORIENTED RATE CONTROL ALGORITHM AND IMPLEMENTATION FOR REALTIME VIDEO CODING

Hung-Chi Fang, Tu-Chih Wang, Yu-Wei Chang and Liang-Gee Chen

DSP/IC Design Lab., Graduate Institute of Electronics Engineering and
Department of Electrical Engineering, National Taiwan University
{honchi, eric, wayne, lgchen}@video.ee.ntu.edu.tw

ABSTRACT

In this paper, a novel rate control algorithm suitable for realtime video encoding is proposed. The proposed algorithm uses mean absolute error (MAE) results of motion estimation (ME) to achieve bitrate control. Neither pre-analysis nor multi-pass encoding is required in our algorithm, which makes realtime hardware implementation possible. A new hardware oriented scene change detection method is also included in this rate control framework to achieve better video quality. Experiment shows our rate control algorithm behaves well in all situations. Hardware architecture for this algorithm is also described. Implementation shows our proposed algorithm can be efficiently integrated into low cost, high efficiency video encoder.

1. INTRODUCTION

Bitrate control is very critical to video quality in a video encoder. For realtime video streaming, the bitrate of the coded bitstream must be well controlled to meet the bandwidth of the channel and the size of bitstream buffer on the decoder. If the encoded bitrate exceeds the channel bandwidth, frame skipping will occur. On the other hand, if the bitrate is too low, the remnant bandwidth will be wasted and the video quality will decrease. Both situations make the quality of service (QoS) unacceptable.

Video sequence itself is inherently variable bitrate (VBR) since its content varies in time domain. However, bandwidth of the channel is usually constant bandwidth (CBR). Reducing the mismatch between source and channel becomes the mission of the rate control algorithm. The goal of rate control may be interpreted as following. Maximize the quality of the coded bitstream for a given target bitrate and buffer constraint. The buffer can be viewed as a rate regulator. It softens the encoded bitrate to fit the channel bandwidth. Its fullness is proportional to the buffer delay, which is very important in the video communication applications. In order to judge the effectiveness of the algorithm, subjective and objective methods should be both used. Peak-signal-to-noise (PSNR) is the most popular way to measure the quality objectively. On the other hand, the subjective is hard to measure and compare by statistical data. Minimize the quality fluctuation of subsequent frames and the number of frame skipping are the two ways to optimize the visual quality in the rate control algorithms.

Many rate control methods in the literatures utilize source modelling to estimate the complexity of the source sequence. The source model may be either theoretically or experimentally established. In [1], [2], the source is modelled as power and quadratic functions. However, the algorithms is much complex since power or quadratic function is required and they are not suitable for hardware implementation. In [3], the source is modelled as uncorrelated and Laplacian distributed and it is adapted by H.263[4] as TMN8 rate control. The TMN8 performs very well in bitrate control and it requires small buffer which is suitable for low delay, handheld communication devices. In [5], a ρ-domain source modelling is proposed and the bitrate estimation of this method is good. The two algorithms perform well in bitrate control, but require frame-based motion estimation (ME), i.e. all macroblocks (MB) perform ME first before coding first MB. And a lot of memory space is required in these two algorithms. Therefore, they are not applicable in hardware implementation.

It is valuable to implement rate control by hardware instead of the system CPU. The rate control algorithm must be performed every MB. If it is controlled by the system CPU, the interrupt occurs too often, which is inefficient. Therefore, a dedicated controller in the encoder is the best solution.

This paper organizes as following. The rate-distortion model and the optimized quantization is shown in Sec. 2. The proposed rate control algorithm is explained in Sec. 3. Simulation results are shown in Sec. 4. Hardware architecture of the proposed algorithm is explained in Sec. 5. Sec. 6 presents the hardware implementation of the proposed algorithm.We conclude this work in Sec. 7.

2. RATE DISTORTION MODEL AND OPTIMAL QUANTIZATION

We use the rate model and distortion model in [3] for optimizing quantization step. They are given as

$$R(Q_i) = \mu \cdot \frac{\sigma^2}{Q_i^2} + \theta \tag{1}$$

$$D = \frac{1}{N}\sum_{i=1}^{N}\frac{Q_i^2}{12} \tag{2}$$

where μ and θ are the model parameters and N is the number of MBs in a frame. The optimized quantization step is found by solving the Lagrange equation and are given as

$$Q_i^* = \sqrt{\frac{\mu\sigma_i}{T-\theta N}\sum_{i=1}^{N}\sigma_i} \tag{3}$$

where T is the target bits for the frame.

Thanks to AVerMedia Technologies, Inc. for funding.

0-7803-7663-3/03/$17.00 ©2003 IEEE

This paper was originally published in the Proceedings of the 2003 IEEE International Conference on Acoustics, Speech, & Signal Processing, April 6-10, 2003, Hong Kong (cancelled). Reprinted with permission.

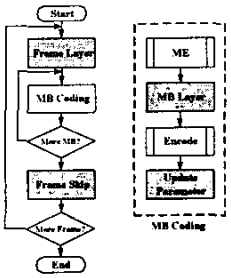

Fig. 1. Flow chart of the proposed rate control algorithm.

3. HARDWARE ORIENTED RATE CONTROL ALGORITHM

Flow char of the proposed rate control algorithm is shown in Fig. 1. There are four subroutines in the proposed algorithm. The frame layer rate control contains the calculation of target number of bits of a frame and the I frame rate control. MB layer rate control determines the MBs mode and quantization parameters (QP) of P frames. Model parameters and scene change detection are done in update parameter routine. If the buffer overflows, some frames must be skipped to lower the buffer fullness. They are elaborated in the following sections. Although only the rate control algorithm about I and P frames are explained in the following sections, the rate control of B frames are exactly the same with that of P frames except that the model parameter of B frames are separated from that P frames.

3.1. Frame Layer Bit Allocation

The target number of bits of the current frame is firstly set to a initial bitrate as

$$T^i = BR \times \alpha_v \tag{4}$$

(5)

where BR denotes the target bitrate and

$$\alpha_v = \rho_v \times L/(\rho_I + (L-1)\cdot\rho_P). \tag{6}$$

The suffix, v, indicates frame type, which may be I or P. L is the length of group of pictures (GOP), i.e. the distance between two I-frames. If there is no GOP structure present, α_v becomes ρ_v. ρ_v represents the relative bitrate of the v-type frame to others. They are chosen that the quality of the I frames are about 2 dB higher than P frames. Then the target bits of the frame ($\hat{T}$) is

$$\hat{T} = T^i - (B - B^t)/4 \tag{7}$$

$$B^t = B^l + BR \times \sum(\alpha_v - 1)\cdot m_v \tag{8}$$

where B represents the number of bits in buffer. B^t is the target buffer fullness and B^l is a constant defined as one-eighth buffer size. The parameter "m_v" is the number of v-type frames previously coded in the same GOP. If there is no GOP structure present, B^t is the same as B^l.

3.2. Rate Control for I Frame

Before we introduce our I frame rate control scheme, some important observations about coding of I frames are proposed:

1. The coding efficiency of subsequent P frames are affected by the quality of the I frame.
2. Pre-analysis is too complex and is not allowed in realtime video encoding.
3. Results of the previously coded I frames are available and useful.
4. Small variation of QP is preferred.

I frames play a key role in video coding in many ways. It is the origin of all subsequent P, all of them are reconstructed by referencing it. But the bitrate paid for the quality of I frame is much higher than that of P. If we make the quality of I frame higher, P frames can get better prediction and produces smaller bitrates. But higher quality means higher bitrate. Thus, the bit budgets for the subsequent P frames are smaller and the qualities are lower. On the other hand, the quality of I frame should not exceed the average quality to much since the viewer will feel uncomfortable. Therefore, the target quality of I frames of the proposed algorithm are about 2dB higher than the average quality. Since pre-analysis will cause latency, it is unwanted in realtime encoding. However, the results of previously coded I frames can be utilized to control the bitrate. Experimental results in Sec. 4 show that the more frequently we change the QP of a frame, the lower the quality we get. So the best strategy of I frame rate control is to decide a fair QP for the whole frame that makes its quality about 2 dB higher than average bitrate. The QP is determined by

$$QP^i = QP_I + F(\hat{T}/T_I) \tag{9}$$

where

$$F(\kappa) = \begin{cases} -4 & \kappa >= 4 \\ -3 & 4 > \kappa >= 2 \\ -2 & 2 > \kappa >= 1.5 \\ -1 & 1.5 > \kappa >= 1.25 \\ 0 & 1.25 > \kappa >= 0.875 \\ 1 & 0.875 > \kappa >= 0.75 \\ 2 & 0.75 > \kappa >= 0.625 \\ 4 & \kappa < 0.625 \end{cases} \tag{10}$$

and $\hat{T}$ is the target bitrate for the I frame. T_I is the number of bits used for previously coded I frame and QP_I is the QP of it. The equations are quite simple and easy to implement. Experimental results in Sec. 4 show that they are quite robust and effective.

3.3. MB Layer Rate Control

Only P frames are allowed to change QP at MB layer in the proposed algorithm. The QP is calculated by Eq. 3. But it is adjusted to fit the hardware implementation. First, σ_i is replaced by MAE from ME to reduce the hardware complexity. Second, the MAE of the MB which does not coded yet is replaced by the average MAE of previous frame. The hardware complexity is greatly reduced since the ME need not to be frame based.

3.4. Scene Change Detection and Handling

Before we introduce our scene change detection method, a definition of scene change is given first. A scene change happens when

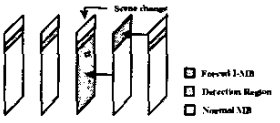

Fig. 2. Diagram of scene change detection.

the correlation between two subsequent frames is small or the motion of them is larger than the search range of ME. The correlation between two frames is found by motion estimation in video coding. If the scene has been changed, the motion estimation will fail. However, if the motion between two frames is too large, these two frames are considered that they are in different scenes. Both situations lead to large MAE of ME. If there is no scene change detection, many MBs in the frame will be coded as intra type (IMB). By the above observations, we use a simple but very effective scene change detection. If the ratio of IMB of a frame exceeds a threshold value, τ, a scene change is said to happen.

However, the scene change must be found as soon as possible. We choose the k-th row to detect the scene change that

$$k = \lfloor \frac{SR}{16} \rfloor + 1 \qquad (11)$$

where SR is the search range of the motion estimation. By choosing the k-th row, the mis-detection owing to the downward global motion can be avoided. Consequent scene changes are forbidden in order to prevent the buffer from overflow, i.e. no scene change detection is performed at the frame immediately following a scene change frame.

When a scene change happens, the coding type of the rest MBs in this frame and the first k rows of next frame as intra, as shown in Fig. 2. By doing so, every MB in the subsequent frames can have a reference MB in the same scene. This makes the prediction error small and stops the propagation of prediction error due to scene change. Therefore, the bitrates of the following frames become small. Thus the extra bits used for coding the scene change frame can be compensated and the average bitrate and quality can be keep stable. The quantizer calculated by Eq. 3 is not reliable when scene change occurs since the current and previous frames are not in the same scene. The QPs of these IMBs are set to QP_s defined as $QP_l + 2$. This is a trade-off between the burst bitrate and average quality. Since the scene has been changed, the QP_l is updated by the coding result of this frame using Eq. 10

The proposed scene change detection and handling scheme are suitable for hardware implementation in two aspects. First, only a counter of intra MB and small detection logics are needed. Second, the coding flow is the same whether the scene changes or not.

4. ALGORITHM SIMULATION RESULT

In order to evaluate and highlight the performance of the proposed I frame rate control scheme. We encode all the frames as I frames. The test sequence is foreman in CIF format and 300 frames are encoded each time. The rate-distortion curves and frame skip numbers are shown in Fig. 3. The rate-distortion curve labelled "VBR" is that all the frames of the sequence are coded using a single QP. That is, there is no rate control at all and no buffer constraints. The other three rate distortion curves use Eq. 10 to adapt QP in frame,

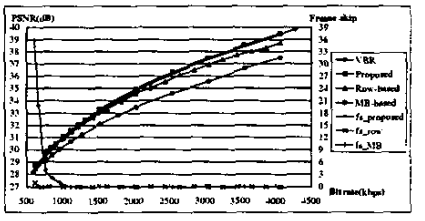

Fig. 3. Rate-distortion curve and frame skip numbers for foreman all I frame coded.

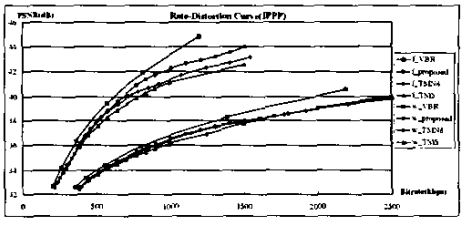

Fig. 4. Rate-distortion curves of foreman and weather coded in IPPP format.

MB-row and MB based. The number of frames skipped is also shown in Fig. 3 referencing the Y-axis at right side. The results show that the proposed frame-based I frame rate control scheme is quite effective. Experimental results of other sequences are almost the same as the result of foreman.

The rate-distortion curves of various rate control algorithm are shown in Fig. 4. The sources are CIF format at 30 fps and the search range of the ME is set to 16. The performance of the proposed algorithm is almost the same with TMN8 [3] and TM5 [6] at medium to low bitrate and outperforms them at high bitrate. Fig. 5 shows PSNR variations with respect to frame number. To explore the performance of the proposed scene change algorithm, we concatenate news, foreman, weather and mobile, each has 50 frames, in a sequence of 200 frames. It shows that the proposed algorithm has a 0.5 dB gain over TM5. When scene change happens, the bit budget of the frame increases to keep the quality smooth. Note that the smoothness of quality comparing to next frame is much more important than comparing to previous frame. Because the quality fluctuations between frames in the same scene are easier to observe that that in different scene. It is interesting that the quality of I frames in small motion sequence is lower than it is respected but the ones in large motion sequence are not. However, this can be overcame by adapting the values of ρ to the property of sequence.

5. ARCHITECTURE OF THE PROPOSED ALGORITHM

Hardware implementation of the algorithm is exactly the same as in software implementation. Fig. 6 is the direct hardware mapping of the proposed algorithm. However, hardware utilization of this architecture is too low and the area is too large. Fig 7 shows the optimal architecture of the proposed rate control architecture. The processing element (PE) consists of three adders, one multiplier and one divider. The R bank is the register bank that contains either the model parameters (M Reg.) or the statistical data (S Reg.). The

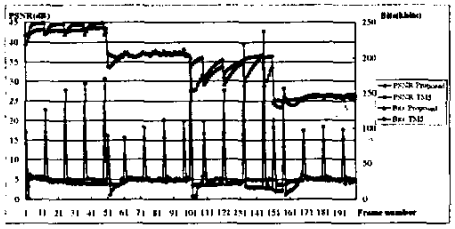

Fig. 5. Comparisons of PSNR and bit number versus frames.

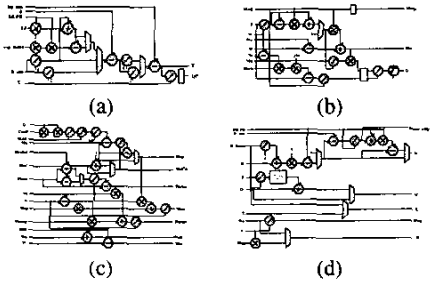

Fig. 6. Direct hardware mapping of the proposed rate control algorithm. (a) Frame Layer. (b) MB Layer. (c) Update parameters. (d) Frame skipping.

state machine is the controller of the PE. It feeds the proper input data to the PE from R bank and configures the PE. The outputs of the PE are stored back to the R bank. Thus, the utilization of the PE is nearly 100% and the area is minimized. It takes six, nine, eleven and three clock cycles to complete the function of the four stage of the rate control algorithm. Only another set of M Reg. is needed to support the B frame coding and the architecture is exactly the same as Fig. 7.

6. IMPLEMENTATION RESULT

The proposed architecture is implemented by TSMC 0.35 μm CMOS technology. The gate count (in two-input NAND gate equivalents) is 12380 and the operating frequency is 40 MHz. The specification of the rate control chip is listed in table 1. The maximal resolution, highest frame rate, maximal bitrate and frame types are not the restrictions of the proposed algorithm. They are the considerations of the bitwidth of the parameters and processing elements. The proposed algorithm and architecture are general for various applications. If B frame rate control is needed, another set of M Reg. of size 2.5k gates is required.

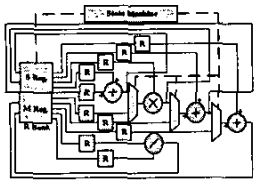

Fig. 7. Optimal architecture of the proposed rate control algorithm.

Table 1. Specification of the rate control chip.

Technology	TSMC 0.35μm 1P4M
Gate count	12380
Operating frequency	40 MHz
Maximal resolution	CIF (352×288)
Highest frame rate	30 fps
Maximal bitrate	1500 kbps
Frame types	I or P

The rate control chip works as following. At the beginning of a frame, the frame layer rate control must be performed. After motion estimation, MAE is passed to the rate control module to calculate the QP and decide a coding mode for the MB. After a MB is coded, the number of bits used is used to update the model parameters. When a frame is finished, frame skipping is determined according to the buffer status. To indicate a start of a stage, a start signal is set by the encoder. Therefore, the proposed rate control algorithm is suitable for hardware implementation in two aspects. First, the interactions between the encoder and the rate control module are quite simple and regular. Second, the spacial and temporal requirements of the proposed rate control algorithm are small. Only about 12k gate count and no on chip memory is required. The number of clock cycles needed are quite small that scheduling of the original encoder does not change too much.

7. CONCLUSION

We proposed a hardware oriented rate control algorithm in this work. Neither multi-pass coding nor pre-analysis is needed, which makes realtime hardware encoding possible. A new scene change detection method suitable for hardware implementation is also included in the proposed algorithm. The performance of the proposed algorithm is better than well-known TMN8 and TM5. The architecture and implementation of the proposed algorithm are also shown in this paper. Implementation results show that the proposed architecture can be integrated into a low cost, realtime video encoder.

8. REFERENCES

[1] W. Ding and B. Liu, "Rate Control of MPEG Video Coding and Recoding by Rate-Quantization Modeling," *IEEE Trans. Circuits Syst. Video Technol.*, pp. 12–20, 2 1996.

[2] T. Chiang and Y. Q. Zhang, "A New Rate Control Scheme using Quadratic Rate Distortion Model," *IEEE Trans. Circuits Syst. Video Technol.*, pp. 246–250, 2 1997.

[3] J. Ribas-Corbera and S. Lei, "Rate Control in DCT Video Coding for Low-Delay Communications," *IEEE Trans. Circuits Syst. Video Technol.*, pp. 172–185, 2 1999.

[4] ITU-T, *Draft ITU-T Recommendation H.263*, 1997.

[5] Z. He, Y. K. Kim, and S. K. Mitra, "Low-Delay Rate Control for DCT Video Coding via -Domain Source Modeling," *IEEE Trans. Circuits Syst. Video Technol.*, pp. 928–940, 8 2001.

[6] MPEG-4, "MoMuSys-FDIS 1.0," Tech. Rep., 8 1999.