RATE-DISTORTION OPTIMIZED MULTI-STAGE RATE CONTROL ALGORITHM FOR H.264/AVC VIDEO CODING

Shuijiong Wu, Yiqing Huang, Qin Liu, and Takeshi Ikenaga

Graduate School of Information, Production, and System, Waseda University, Japan
email: wushuijiong@akane.waseda.jp

ABSTRACT

For real H.264/AVC codec, the bit allocation in rate control is crucial to coding efficiency. Meanwhile, the rate-distortion-optimization (RDO) based mode-decision technique also affects performance considerably. This paper presents a R-D optimized multi-stage rate control scheme where both bit allocation and the Lagrange multiplier (λ_{MODE}*) selection are considered. At the first stage, frame layer bit allocation is executed by analyzing frame complexity which is represented by mean-absolute-difference (MAD). Secondly, in order to a select suitable mode to generate reasonable bits,* λ_{MODE} *is further adjusted by using coded and remaining bits information. At the last stage, bits are allocated on macroblock (MB) layer and the computed quantization parameter (QP) is further adjusted. Simulation results verify the performance of the proposed algorithm. Compared with the recommended rate control in H.264/AVC reference software, a gain up to 0.89dB in PSNR is achieved, and the subjective coding quality is also improved.*

1. INTRODUCTION

H.264/AVC [1] is the latest state-of-the-art video coding standard. With many sophisticated techniques involved, it achieves a much higher coding efficiency compared to any other existing standard. Being faced with the constraints imposed by limited storage size and network bandwidth, rate control is employed in a practical H.264/AVC encoder to regulate output bit-streaming to meet channel bandwidth and buffer constraints while keeping the coding quality. Among several problems in recent rate control, the following two are essential ones that directly related to performance.

One problem is bit budget assignment for each coding unit, such as a group of pictures (GOP), frame, or single MB. In many famous rate control schemes, such as JVT-G012 [2], which is adopted by H.264/AVC reference software JM13.2 [3], bits are allocated according to target rate and virtual buffer status. Although the performance is improved by [2], there still exists some inaccurate allocation since the frame complexity and its movement do not receive enough attention when sequence is coded in frame by frame order. On the MB layer, the MAD ratio model utilized to represent frame remaining complexity is too simple to predict the bits required by upcoming MB.

The other aspect is to consider distortion as well as the bit-rate. Actually, in order to improve prediction accuracy, H.264/AVC provides variable-size block motion estimation (ME) and mode decision. Obviously, how to select a best mode for each MB from so many candidates, especially under rate constraint, will largely affect the coding quality and actual bit generation. Thus, the idea of RDO proposed in [4] should be fully considered when performing rate control. In H.264/AVC, mode is decided by (1) [5], where J_{MODE} is the Lagrangian cost, D denotes the sum of square differences between the original block and its reconstruction, and R is the actual number of bits associated with a chosen mode. λ_{MODE} is the Lagrange multiplier computed by (2) given in [6], and QP is the MB quantization parameter. However, there are no schemes which make the decision by considering the status of a whole sequence.

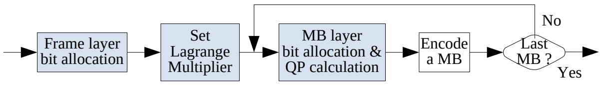


Figure 1: Block diagram of encoding one frame with rate control.

$$\begin{cases} \min\{J_{MODE}\} \\ J_{MODE} = D + \lambda_{MODE} \bullet R \end{cases} \tag{1}$$

$$\lambda_{MODE} = 0.85 \times 2^{(QP-12)/3} \tag{2}$$

Theoretically, the overall coding efficiency can be improved if either of the forementioned problems can be solved well. Based on this understanding, a R-D optimized multi-stage rate control scheme is presented in this paper. This paper mainly targets the real-time video coding for the applications such as videophone or video conference, especially for low-motion or indoor video content. For high-motion cases, improvements are also observed although they are not so significant. Figure 1 shows the block diagram of encoding one frame with rate control. Steps shown in grey colour will be executed with our proposed algorithm. At first, the bit budget will be allocated on frame layer with improved complexity analysis. To solve the forementioned problem, frame complexity proportion and its movement will be fully considered. Second, in order to select suitable mode to generate reasonable bits for upcoming frame, Lagrange multiplier is further adjusted with coded and remaining bits information. Note that pre-coded frames may select best modes, generally use more bits and obtain high quality, but more remaining frames may be affected because of the lack of bit budget. Our method is to adjust current frame λ_{MODE} to be better for whole sequence. The λ_{MODE} calculation is applied on frame layer which is cooperated with bit allocation modification. At the third stage, MAD information of both previous and current frame are fully utilized to predict target bits accurately for each MB. And the computed QP will also be adjusted before real coding.

The rest of this paper is organized as follows. Section 2 gives out the proposed multi-stage rate control algorithm based on the detail description of three sub schemes. In section 3, the overall RDO based rate control algorithm is described. Experiments and comparisons are given in section 4. The paper concludes with section 5.

2. MULTI-STAGE RATE CONTROL SCHEMES

2.1 Frame Layer Bit Allocation

By using a weighted combination of $T_r(j)$ and $T_{buf,j}$ given by [7], the target bits T_j for current j^{th} frame (j: $1 \sim N_P$, is P frame suffix, and N_p is the number of total P frames) is predicted by (3) in conventional work, where β is a constant with a typical value of 0.50, $T_r(j)$ is the total remaining bits after coding $(j\text{-}1)^{\text{th}}$ P frame, N_r is the total number of remaining frames.

$$\begin{cases} T_j = \beta \bullet T_{rf,j} + (1-\beta) \bullet T_{buf,j} \\ T_{rf,j} = \dfrac{T_r(j)}{N_r} \\ T_{buf,j} = (u/F_r) - \gamma \bullet (B_c(j) - B_t(j)) \end{cases} \tag{3}$$

$T_{rf,j}$ and $T_{buf,j}$ denote current j^{th} frame target bits estimated from the remaining bits and current buffer status, respectively. $T_{buf,j}$ is defined by (3), where u and F_r are the target bandwidth and frame rate predetermined for whole sequence, respectively. $B_c(j)$ denotes buffer occupancy after coding $(j\text{-}1)$ frames, and $B_t(j)$ represents target buffer level of the j^{th} frame. γ is a constant with typical value of 0.50.

In equation (3), the status of remaining bits $T_{rf,j}$ is only expressed in the form of $T_r(j)$ divided by N_r. The difference of frame complexity is not fully considered. In this paper, we improve the complexity analysis, and the distribution of bit count is scaled by a function of $MAD_{Ratio,j}$ named SF_j based on our experiments. They are shown in (4) to (6). The $MADP_j$ is the predicted MAD of current j^{th} frame. It is calculated by the original frame layer complexity prediction method in [2]. $MADA_j$ is the actual MAD value. SF_j is a scale factor based on $MAD_{Ratio,j}$.

$$T_{rf,j} = SF_j \bullet \frac{T_r(j)}{N_r} \tag{4}$$

$$SF_j = \begin{cases} 0.4 & , \quad MAD_{Ratio,j} < 0.5 \\ 0.8 \times MAD_{Ratio,j} & , \; 0.5 \le MAD_{Ratio,j} < 1.5 \\ 1.5 & , \quad MAD_{Ratio,j} \ge 1.5 \end{cases} \tag{5}$$

$$MAD_{Ratio,j} = \begin{cases} 1 & , \quad j = 1 \\ \dfrac{MADP_j}{\sqrt{\dfrac{1}{j-1} \times \sum\limits_{l=1}^{j-1} MADA_l^2}} & , \quad j \ge 2 \end{cases} \tag{6}$$

2.2 Adaptive Lagrange Multiplier Adjustment

The effect of λ_{MODE} in selecting modes is described in [9]. When an MB is under coding, assume two different modes M_1 and M_2. Their rate-distortion cost (RD_cost) are decided by (7), where $D_{M\_1} < D_{M\_2}$, $R_{M\_1} > R_{M\_2}$, and $J_{M\_1} < J_{M\_2}$.

$$\begin{aligned} J_{M\_1} &= D_{M\_1} + \lambda_{MODE} \bullet R_{M\_1} \\ J_{M\_2} &= D_{M\_2} + \lambda_{MODE} \bullet R_{M\_2} \end{aligned} \tag{7}$$

Suppose λ_{MODE} is increased, it is obvious that the RD_cost increment of M_1 is larger than that of M_2 because of $R_{M\_1} > R_{M\_2}$. So the best mode for current MB has the possibility to change from M_1 to M_2 if λ_{MODE} is large enough. From this, we can find that a larger λ_{MODE} corresponds to higher distortion and lower bit-rate. Contrarily, a smaller λ_{MODE} corresponds to lower distortion and higher bit-rate. In other words, we can change the rate and distortion of upcoming picture by modifying λ_{MODE}. As mentioned in section 1, reasonable allocation of bits is very important for whole video sequence. Since bits utilized for each frame will greatly affect the succeeding frames. And this kind of effectiveness will be propagated or even magnified throughout the entire coding process. In one-pass coding, we can never know the exact complexity of each frame before real coding. So it is a reasonable choice to consider information of already utilized, current predicted, and remaining bits to modify λ_{MODE} in a universal point of view. Based on this idea, the proposed λ_{MODE} adjustment is given by (8). f_R is a function represents bits usage measurement determined by context-adaptive bits analysis. In order to obtain this function, we first compare the bits usage among current P frame, all pre-coded P frames, and all remaining ones by (9) and (10).

$$\lambda_{MODE} = 0.85 \times f_R \times 2^{(QP-12)/3} \tag{8}$$

$$Bit_{Ratio,j} = \frac{\omega_j}{\nu_j} \qquad \text{for } j > 1 \tag{9}$$

$$\omega_j = \frac{\dfrac{1}{j-1} \times \left(\sum\limits_{l=1}^{j-1} B_{f,l}\right)}{\dfrac{T_P}{N_P}}, \qquad \nu_j = \frac{T_j}{\dfrac{T_r(j)}{N_r}} \tag{10}$$

The $Bit_{Ratio,j}$ computed by (9) is the combinatorial bit ratio for the j^{th} P frame. $B_{f,j}$ denotes the actual bits utilized for encoding j^{th} frame, T_P is the total target bits for all P frames, and T_j represents the predicted target bits for j^{th} frame. In (10), ω_j and ν_j are backward ratio and forward ratio, representing previously coded P frames bit status and the relationship between current and remaining frames, respectively. ω_j is the measurement of the bit usage of "the past". When ω_j is large, it means that the coded frames may very likely have utilized too many bits, and we should increase λ_{MODE} to limit the bit usage upon the upcoming frame. By contraries, if it is small, the rest frames may have rich bit resource to be coded, and the corresponding λ_{MODE} should also be small. The operator ν_j is a factor represents the proportion of current frame in bits allocation among all not-yet-coded frames. Larger the ν_j is, more complexity the current frame holds compared with other remaining ones. Of course, this frame should be allocated more bits to encode. Thus, the change of ν_j is inverse to the trend of λ_{MODE}. By using these forementioned operators, we can get the definition of f_R by (11).

$$f_R(j) = Bit_{Ratio,j} \bullet \sqrt{\frac{Bit_{Ratio,j}}{Bit_{Ratio,j-1}}} \tag{11}$$

Here $Bit_{Ratio,j-1}$ is for the previous $((j\text{-}1)^{\text{th}})$ frame. In (11), the change of Bit_{Ratio} is also considered by using ratio (current frame Bit_{Ratio} to previous Bit_{Ratio}) square root. The idea of introducing this part into calculation is that when the Bit_{Ratio} becomes larger than last frame, it means the modification effectiveness should be further strengthened. Contrarily, when Bit_{Ratio} becomes smaller, λ_{MODE} should also be restricted in order to avoid excessive control.

The position of current frame in the sequence is also considered in our method. Generally, since coding is based on a frame by frame flow, anterior frames often get more bits for coding, and the bits left for remaining frames usually decrease gradually. This is not reasonable for a sequence with different features. For example, if the latter part holds high complexity, the coding quality of the overall sequence will drop significantly. Thus, f_R is scaled by (12), where a_0 and a_1 are constants with typical values of 1.2 and 0.2, respectively. The final f_R is defined by (13) and utilized to adjust corresponding Lagrange multiplier λ_{MODE} in equation (8). The constant values in (12) and (13) are obtained statistically based on sample experiments.

$$f_R(j) = (a_0 - \frac{a_1}{N_P} \times j) \bullet f_R(j) \tag{12}$$

$$f_R(j)_{final} = \begin{cases} 1.0 & , \quad f_R(j) < 1.1 \\ 1.0 + 0.4 \times (f_R(j) - 1.1) & , \; 1.1 \le f_R(j) < 1.6 \\ 1.2 & , \quad f_R(j) \ge 1.6 \end{cases} \tag{13}$$

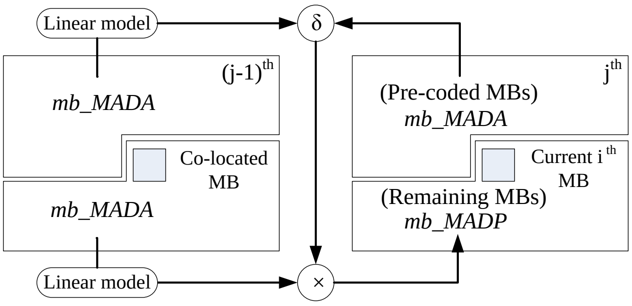


Figure 2: Remaining MBs' complexity prediction.

It should be noted that in this stage, λ_{MODE} will be calculated with all QP values covering the entire QP variance range (from QP minimum to the QP maximum). This is executed at the beginning of each frame to update the look-up-table (LUT) of λ_{MODE} values. MBs in current frame search for its λ_{MODE} according to the computed QP in section 2.3 to act mode decision and other process.

2.3 MB Layer Bit Allocation and QP Adjustment

The original calculation of target bits $mb\_T_{i,j}$ for current i^{th} MB (i: $1\sim N_{mb}$, where N_{mb} is the total number of MBs in one frame) in j^{th} frame is defined by (14) in recommended algorithm described in JVT-G012 [2].

$$mb\_T_{i,j} = T_j \times \frac{mb\_MADP_{i,j}^2}{\sum_{l=i}^{N_{mb}} \left(c_{i,j} \times mb\_MADA_{l,j-1} + d_{i,j}\right)^2} \tag{14}$$

The $mb\_MADP_{i,j}$ and $mb\_MADA_{i,j}$ are predicted MAD and actual MAD values of i^{th} MB in j^{th} frame, respectively. They are computed by the method in [3]. $c_{i,j}$ and $d_{i,j}$ are MB layer parameters in linear model, which will be updated after coding each MB. However, the complexity prediction for remaining MBs is not accurate since it only utilizes their co-located MBs in previous frame.

Our proposal is shown in Figure 2. All MBs in previous frame and all pre-coded MBs in current frame are utilized. Firstly, by using pre-coded MBs in current frame and their co-located ones, ratio δ is computed by (15). It is defined as a measurement of complexity change from last frame ($(j\text{-}1)^{th}$)to current one (j^{th}). Second, the remaining MBs' complexity is predicted by (16) using their co-located ones and $\delta_{i,j}$. Finally, target bits $mb\_T_{i,j}$ is calculated in (17) by new estimation of complexity proportion. $mb\_MADP_r(j)$ stands for predicted sum of MAD for remaining MBs in j^{th} frame. It is calculated using linear model in [2] and the ratio $\delta_{i,j}$ introduced by our proposal in (15).

$$\delta_{i,j} = \frac{\sum_{m=1}^{i-1}\left(mb\_MADA_{m,j}\right)}{\sum_{l=1}^{i-1}\left(c_{i,j} \times mb\_MADA_{l,j-1} + d_{i,j}\right)} \tag{15}$$

$$mb\_MADP_r(j) = \delta_{i,j} \times \sum_{n=i}^{N_{mb}}\left(c_{i,j} \times mb\_MADA_{n,j-1} + d_{i,j}\right) \tag{16}$$

$$mb\_T_{i,j} = T_j \times \frac{mb\_MADP_{i,j}}{mb\_MADP_r(j)} \tag{17}$$

Since each frame is coded MB by MB, the remaining bit budget for un-coded MBs is often getting less and less. Hence $mb\_T_{i,j}$ is further scaled by (18), where e_0 and e_1 are two constants with typical values of 0.8 and 0.4, respectively. It is easy to find that (18) has a form similar to (12) since both of them are created for saving bit budget at the beginning of coding for the latter part. The main difference is that f_R is inverse to bit usage so that the minus operation is utilized.

$$mb\_T_{i,j} = (e_0 + \frac{e_1}{N_{mb}} \times i) \bullet mb\_T_{i,j} \tag{18}$$

By subtracting the header bits, the texture bits $mb\_B_{i,j}$ for encoding is further computed by (19). To avoid it to be too small, a lower bound is made by (20).

$$mb\_B_{i,j} = mb\_T_{i,j} - mb\_H_{i,j} \tag{19}$$

$$mb\_B_{i,j} = \max\{mb\_B_{i,j}, \frac{u}{F_r \bullet N_{mb} \bullet MINVALUE}\} \tag{20}$$

Here, the $mb\_H_{i,j}$ is the average number of header bits generated by coded MBs in current j^{th} frame, $MINVALUE$ is a constant with typical value of 4. When $mb\_B_{i,j}$ is smaller than the lower bound, the computed $QP_{i,j}$ does no longer satisfy the bits constraints. So we do the adjustment based on (21). This adjusted $QP_{i,j}$ will be utilized in mode decision and other posterior processes for current MB.

$$QP_{i,j} = QP_{i,j} + 1 \tag{21}$$

3. OVERALL RATE CONTROL ALGORITHM

Based on the forementioned three key stages in section 2, a R-D optimized multi-stage rate control scheme is proposed. The basic framework is similar with JVT-G012 [2]. Figure 3 shows the flowchart of the proposed scheme. It gives out the detail explanation of Figure1 mentioned in the first section. The processes with bold font are executed with proposed algorithm while others are remained the conventional work in [2]. At the first stage, the global complexity analysis is added to enhance the accuracy of frame layer bit allocation. Average of remaining bits is replaced by complexity estimation based bit prediction. The work in second stage is to adaptively adjust the Lagrange multiplier λ_{MODE} based on previ-

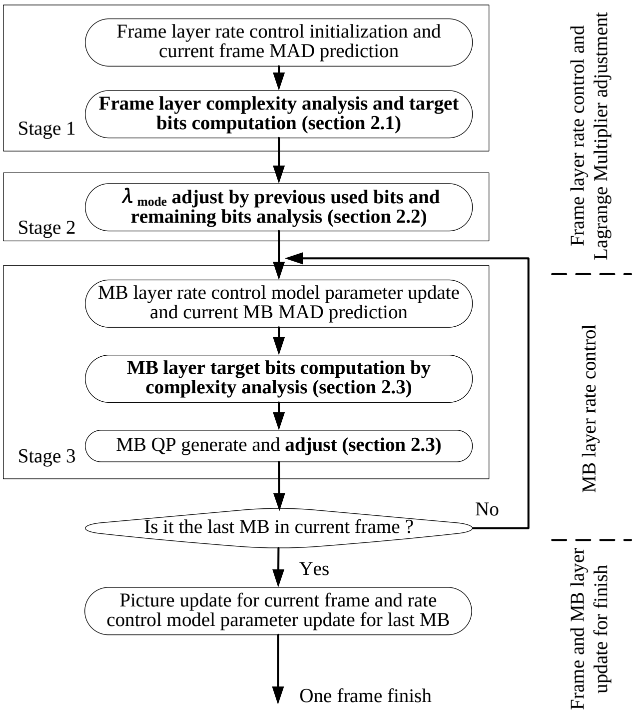


Figure 3: Flowchart of proposed RDO based rate control scheme.

ous used bits and remaining bits analysis. It controls the bit generation by the effect on mode decision. Different from the existing work such as [9], a combination of forward and backward bit measurements is proposed to optimize the final result. Moreover, the operation is applied for the frame layer instead of the MB layer to achieve an overall encoding system level adjustment. MB layer rate control is carried out at stage three. In our scheme, a method considering complexity movement between current and previous frames is utilized to compute MB target bits. And the computed QP for current MB is finally adjusted according to the number of MB target texture bits. After encoding each MB, the MB layer model parameters are updated. Finally, after finishing encoding all of the MBs in current frame, frame layer model parameters are updated for next frame.

4. SIMULATION RESULTS

The proposed rate control scheme is tested by the recent H.264/AVC reference software JM13.2 [3]. The widely utilized recommended algorithm in JVT-G012 [2] is selected as a benchmark since all of its simulation parameter information is open and it has already been implemented in JM software. Four QCIF and four CIF sequences in 4:2:0 format are coded with H.264/AVC baseline. With the frame rate 30 fps, 300 frames are coded for each sequence with the GOP structure of IPPP mode (first frame is intra coded *I* frame, and the remaining frames are *P* frames) for real-time applications. CAVLC is chosen as symbol mode. The search range is set to be 16 for QCIF and 32 for CIF, respectively. One reference frame is adopted. All other parameters are carefully selected for both JVT-G012 [2] and proposed algorithm to be equivalent.

Table 1 shows the simulation results. It can be found that, our scheme out performs [2] on both coding quality (represented by peak-signal-noise-ratio (PSNR)) and bit-rate control under various target bit-rate constraints. Gains up to 0.79 dB and 0.89 dB are observed for QCIF and CIF sequences, respectively. Since this paper targets real-time applications such as videophone or video conference, better performance can be achieved for sequences with low-motion or containing indoor scenes. For those sequences with high-motion scene, the improvement can also be observed though they are not so significant. Take “foreman.cif” for example, during encoding process, there exists a period that the camera moves quite fast and the gain on PSNR is only about 0.2 dB. For “football.qcif” (simulation result is not given for space limitation), which is a typical high-motion sequence, the PSNR remains almost the same as that of JVT but the bit-rate can be controlled much closer to the target. The most possible reason for this kind of result is that when objects or background move fast, it becomes much harder for rate control to estimate the frame or MB complexity. Example of PSNR performance (“salesman.qcif” at 48Kbps) is drawn in Figure 4. The R-D curves for all tested eight sequences are also shown in Figure 5. Both QCIF and CIF sequences obtain optimized R-D performance by using our rate control scheme. Moreover, PSNR gains are also reflected in the improvement of subjective quality of reconstructed frames, as shown in Figure 6. It is obvious that the visual distortion is reduced by using our scheme.

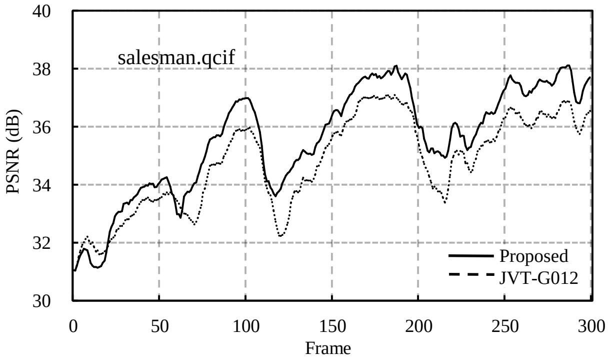
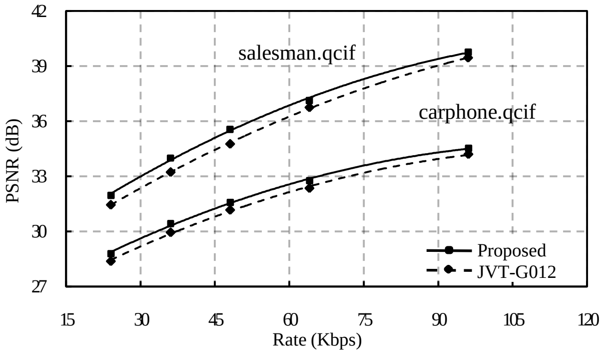
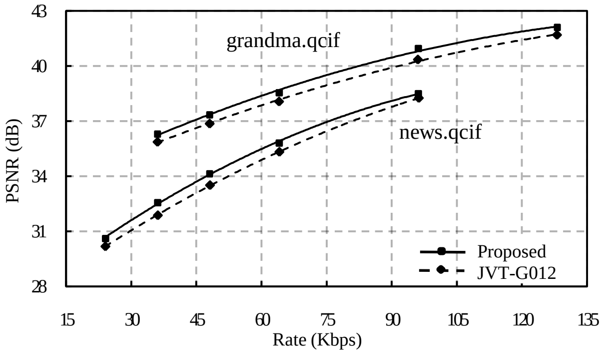


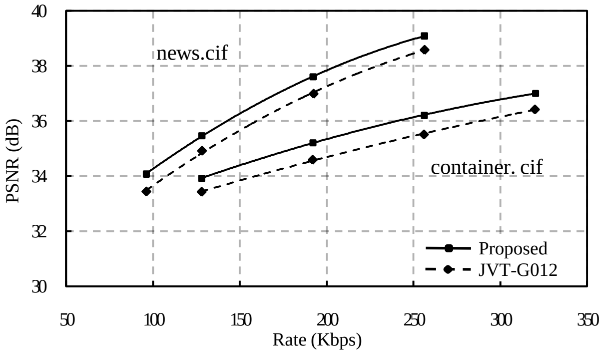
Figure 4: PSNR of salesman.qcif at 48Kbps



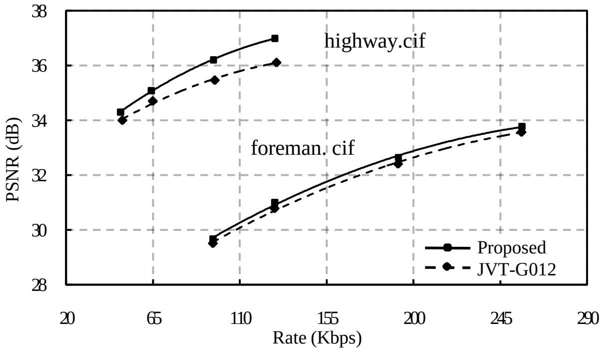
(a) salesman.qcif, carphone.qcif.



(b) grandma.qcif, news.qcif.



(c) news.cif, container.cif.



(d) highway.cif, foreman.cif

Figure 5: R-D curves.

Table 1: Simulation Results

Target (Kbp)	**JVT-G012**		**Proposed**			**Target (Kbp)**	**JVT-G012**		**Proposed**		
	Rate (Kbps)	**PSNR (dB)**	**Rate (Kbps)**	**PSNR (dB)**	**ΔPSNR (dB)**		**Rate (Kbps)**	**PSNR (dB)**	**Rate (Kbps)**	**PSNR (dB)**	**ΔPSNR (dB)**
salesman (QCIF, 300frms, 30Hz)						**news (QCIF, 300frms, 30Hz)**					
24	24.05	31.46	24.03	31.97	+0.51	24	24.07	30.19	24.06	30.62	+0.43
36	36.07	33.23	36.04	34.00	+0.77	36	36.12	31.88	36.07	32.57	+0.69
48	48.05	34.77	48.03	35.56	+0.79	48	48.17	33.51	48.07	34.14	+0.63
64	64.04	36.74	64.03	37.13	+0.39	64	64.17	35.33	64.08	35.82	+0.49
96	96.06	39.45	96.04	39.76	+0.31	96	96.29	38.25	96.14	38.50	+0.25
carphone (QCIF, 300frms, 30Hz)						**grandma (QCIF, 300frms, 30Hz)**					
24	24.04	28.38	24.04	28.80	+0.42	36	36.05	35.87	36.06	36.30	+0.43
36	36.06	29.95	36.05	30.43	+0.48	48	48.10	36.87	48.07	37.35	+0.48
48	48.07	31.17	48.07	31.58	+0.41	64	64.09	38.06	64.08	38.55	+0.49
64	64.08	32.35	64.07	32.77	+0.42	96	96.07	40.35	96.12	40.96	+0.61
96	96.13	34.20	96.10	34.54	+0.34	128	128.12	41.69	128.15	42.11	+0.42
news (CIF, 300frms, 30Hz)						**container (CIF, 300frms, 30Hz)**					
96	96.27	33.45	96.17	34.09	+0.64	128	128.04	33.43	128.07	33.92	+0.49
128	128.33	34.92	128.19	35.47	+0.55	192	192.06	34.59	192.17	35.22	+0.63
192	192.57	36.98	192.31	37.61	+0.63	256	256.06	35.51	256.27	36.21	+0.70
256	256.49	38.58	256.41	39.09	+0.51	320	320.11	36.42	320.34	37.01	+0.59
foreman (CIF, 300frms, 30Hz)						**highway (CIF, 300frms, 30Hz)**					
96	96.09	29.51	96.10	29.67	+0.16	48	49.14	34.00	48.16	34.30	+0.30
128	128.11	30.78	128.15	31.00	+0.22	64	64.96	34.70	64.14	35.09	+0.39
192	192.11	32.41	192.21	32.65	+0.24	96	97.02	35.46	96.38	36.21	+0.75
256	256.11	33.57	256.28	33.77	+0.20	128	129.07	36.10	128.23	36.99	+0.89

(a) 122nd in "salesman" at 48Kbps (b) 186th in "carphone" at 36Kbps

Figure 6: Comparison of reconstructed frames with rate control in JM13.2 (Left) and with proposed scheme (Right).

5. CONCLUTION

In this paper, a R-D optimized multi-stage rate control scheme has been proposed for H.264/AVC. Both frame and MB layer bit allocation are improved by accurate complexity analysis. The Lagrange multiplier λ_{MODE} is adaptively adjusted for mode decision in each frame by using previous used bits and remaining bits information. These techniques are adopted in three key stages of rate control covering both frame and MB layers. Simulation results show that the visual distortion is reduced, and the encoder gains up to 0.89 dB in PSNR by our scheme compared with the recent rate control algorithm in JVT-G012 [2].

ACKNOWLEDGMENT

This work was supported by fund from CREST, JST.

REFERENCES

[1] JVT, *Advanced Video Coding (AVC) -3rd Edition*, ITU-T Rec. H.264 and ISO/IEC 14496-10 (MEPG-4 Part 10). 2004.

[2] Z.G. Li, F. Pan, K.P. Lim, G. Feng, X. Lin, and S. Rahardja, "Adaptive basic unit layer rate control for JVT," *JVTG012, 7th Meeting*, Pattaya, Thailand, March 2003.

[3] JVT, "H.264/AVC reference software (JM13.2)," http://www.hhi.fraunhofer.de/en/departments/image-processing.html.

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

[5] X. Li, N. Oertel, A. Hutter, and A. Kaup, "Rate-distortion optimized frame level rate control for H.264/AVC," *2007 European Signal Processing Conference*, Lausanne, Switzerland, Aug 2008.

[6] T. Wiegand, H. Schwarz, A. Joch, F. Kossentini, and G.J. Sullian, "Rate constrained coder control and comparison of video coding standards," *IEEE Trans. Circuits and Syst. Video Technol.*, vol. 13, pp. 688-703, July 2003.

[7] M. Jiang, X. Yi, and N. Ling, "Improved frame-layer rate control for H.264 using MAD ratio," *IEEE Intl. Symp. on Circuit and Syst.*, vol. 3, pp. 813-816, Vancouver, Canada, May 2004.

[8] Z.G. Li, C. Zhu, N. Ling, X.K. Yang, G.N. Feng, S. Wu, and F. Pan, "A unified architecture for real-time video-coding systems," *IEEE Trans. Circuit and Syst. Video Technol.*, vol. 13, no. 6, pp. 472-487, June 2003.

[9] M. Jiang and N. Ling, "On lagrange multiplier and quantizer adjustment for H.264 frame-layer video rate control," *IEEE Trans. Circuits and Syst. Video Technol.*, vol. 16, no. 5, pp. 663-669, MAY 2006.

[10] M. Jiang and N. Ling, "An improved frame and macroblock layer bit allocation scheme for H.264 rate control," *Proc. 2005 IEEE Intl. Symp. on Circuits and Syst.*, pp. 1501-1504, Kobe, Japan, May 2005.