A NEW RATE CONTROL SCHEME FOR H.264 VIDEO CODING

Peng Yin and Jill Boyce

peng.yin@thomson.net, jill.boyce@thomson.net
Corporate Research, Thomson Inc., Princeton, NJ 08540, USA

ABSTRACT

The H.264/MPEG-4 AVC video coding standard is very promising due to its high coding efficiency. In this paper, we propose a new constant bit rate control method based on the rate-distortion model of TMN8. The novelty of our approach is to use simple preprocessing to achieve the target bit rate more accurately, better target bit allocation and buffer management for both frame-level and macroblock-level rate control, improved perceptual quality, and adoption of virtual frame skipping. Simulations show that our method can meet the target bitrate very accurately, even for content with scene changes and scene transitions, and achieves both good subjective and objective quality. In addition, our approach adds little complexity to the encoder, and can therefore be used for a real-time encoder.

1. INTRODUCTION

The H.264 (or JVT, MPEG-4 AVC) [1] video coding standard has gained more and more attention recently. It employs a number of new technologies to significantly improve the coding efficiency, such as intra-prediction coding, multiple reference pictures and variable block size for motion estimation and compensation, 4x4 integer transform, in-loop deblocking filter, etc. With all these features, the complexity of the H.264 video coding becomes much higher than its previous standards, such as H.263 and MPEG2.

Rate control plays an important role in video coding. In many applications, video sequences must be transmitted over constant bitrate channels. Therefore, rate control has to be used to regulate the variable bitrate of the coded stream. The topic has been widely studied for older standards, but not for H.264. The new features of H.264 involve more complexity, which makes it more difficult to present a good rate control scheme while at the same time not overly increasing the coding complexity.

Recently, several papers have addressed rate control in H.264 [2][3]. Different rate-distortion (R-D) models are discussed, such as a quadratic R-D model in [2], and a linear model in [3]. In this paper, we shall adopt and enhance the model used in TMN8 of H.263+[4]. This model uses Lagrangian optimization to minimize distortion subject to the target bitrate constraint. The scheme is simple and is known to be able to achieve both high quality and accurate bit rate. To adapt the model into H.264 and to further improve the performance, we have to consider several issues. First, rate-distortion optimization (RDO) (i.e., rate-constrained motion estimation and mode decision) is a widely accepted approach in H.264 for mode decision and motion estimation, where the quantization parameter (QP) (used to decide λ in the Lagrangian optimization) needs to be decided before RDO is performed[5]. But the TMN8 model requires the statistics of prediction error signal (residue) to estimate QP, which means that motion estimation and mode decision needs to be performed before QP is made, thus resulting in a chicken and egg dilemma. Second, TMN8 is targeted at low delay applications. But H.264 can be used for various applications, such as video conferencing, broadcast, film studio, etc. Therefore a new bit allocation and buffer management scheme is needed. Third, TMN8 adapts the QP at macroblock (MB) level. Though a constraint is made on the QP difference (DQUANT) between current MB and last coded MB, large QP variations within the same picture can be observed and has a negative subjective effect. In addition, it is known that using constant QP for the whole image may save additional bits for coding DQUANT, thus achieving higher PSNR for very low bit rate. Finally, H.264 uses 4x4 integer transform and if the codec uses some thresholding techniques such as in JM reference software [6], details may be lost. Therefore, it is useful to adopt the perceptual model in the rate control to maintain more details.

In this paper, we shall present new techniques to resolve the above issues. We first briefly discuss TMN8 model in Section 2. Our new rate control scheme is given in Section 3, followed by simulation result and conclusion. Only baseline profile, containing *I* and *P* frames, is addressed.

2. TMN8 RATE CONTROL SCHEME

The TMN8 rate control uses a frame-layer rate control to select the target number of bits for the current frame and a MB-layer rate control to select the value of QP for the MBs [4].

In the frame-layer rate control, the target number of bits for the current frame is determined by

$$B = R/F - \Delta, \tag{1}$$

$$\Delta = \begin{cases} W/F, & W > Z \bullet M \\ W - Z \bullet M, & otherwise \end{cases} \tag{2}$$

$$W = \max(W_{prev} + B' - R/F, 0), \tag{3}$$

where B is the target number of bits for a frame, R is the channel rate in bits per second, F is the frame rate in frames per second, W is the number of bits in the encoder buffer, M is the maximum buffer size, W_{prev} is the previous number of bits in the buffer, B' is the actual number of bits used of encoding the previous frame, and $Z=0.1$ is set by default to achieve the low delay.

The MB-layer rate control selects the value of the quantization step size for all the MBs in a frame, so that the sum of the MB bits is close to the frame target B. The optimal quantization step size Q_i^* for MB i in a frame can be determined by

$$Q_i^* = \sqrt{\frac{AK}{\beta_i - AN_i C}} \quad \frac{\sigma_i}{\alpha_i} \quad \sum_{k=1}^{N} \alpha_k \sigma_k, \tag{4}$$

0-7803-8554-3/04/$20.00 ©2004 IEEE.

where K is the model parameter, A is the number of pixels in a MB, N_i is the number of MBs that remain to be encoded in the frame, σ_i is the standard deviation of the residue in the *ith* MB, α_i is the distortion weight of the *ith* MB, C is the overhead rate, and β_i is the number of bits left for encoding the frame by setting $\beta_1 = B$ at the initialization stage.

3. PROPOSED RATE CONTROL SCHEME

In this section, we shall present our new rate control scheme in H.264. We shall focus on the issues raised in Section 1. Similar to TMN8, a frame-layer rate control to select the target number of bits for the current frame and a MB-layer rate control to select the value of QP for the MBs is used. In addition, a pre-processing stage is added.

3.1. Preprocessing Stage

From equation (4), we can see that the TMN8 model requires the knowledge of standard deviation of the residue to estimate QP. However, RDO requires knowledge of the QP to perform motion estimation and mode decision thus to produce the residue. To overcome this dilemma, [2] uses the residue of the collocated MB in the most recently coded picture with the same type to predict that of the current MB, and [3] employs a two-step encoding, where the QP of the previous picture (QP_{prev}) is first used to generate the residue, and then the QP of current MB is estimated based on the residue. The former approach is simple, but it lacks precision. The latter approach is more accurate, but it requires multiple encoding, thus adding too much complexity.

In our approach, the residue of each picture is estimated in the preprocessing stage. Experiments show that a simple preprocessing is sufficient to give a good estimation of the residue. To reduce the complexity, for *I* pictures we only test the 3 most probable intra16x16 modes (vertical, horizontal and DC mode) and MSE (Mean Square Error) of the prediction residual is used to select the best mode. The spatial residue is then generated using the best mode. It should be noted that we use the original pixel values for intra prediction instead of reconstructed ones, simply because the reconstructed pixels are not available. For *P* pictures, we perform a rate-constrained motion search using only the 16x16 block type and 1 reference picture with fast motion estimation [5]. The temporal residue is generated using the best motion vector in this mode. The average QP of the previously coded picture with the same coding type is used to decide λ on rate-constrained motion search. The experiment shows that by constraining the difference of QP between previous coded picture and current picture, the λ based on QP_{prev} has minor impact on motion estimation. The side advantage of this approach is that the motion vectors generated during the preprocessing stage can be used as initial motion vectors in the motion estimation during the normal encoding.

3.2. Frame-layer rate control

TMN8 is targeted to low-delay and low bit rate applications, which assume to encode only *P* pictures after the first *I* picture, hence the bit allocation model as shown in equation (1) should be re-defined to adapt to the various applications which use more frequent I pictures. The QP estimation model by equation (4) can result in large QP variation within one image, thus a frame-level QP is better first estimated to put a constraint on the variation of MB QP. In addition, for very low bit rate, due to the overhead of coding the DQUANT, it may be more efficient to use a constant picture QP. So a good rate control scheme should allow rate control at both the frame-level and the MB-level.

In this section, we shall first propose a new bit allocation scheme. Then we shall present a simple scheme to decide a frame-level QP.

In many applications, e.g. real-time encoders, the encoder does not know the total number of frames that need to be coded beforehand, or when scene changes will occur. Thus we adopted a GOP layer rate control to allocate target bits for each picture. The H.264 standard does not actually contain Group of Pictures, but the terminology is used here to represent the distance between I pictures. The length of the GOP is indicated by N_{GOP}. If $N_{GOP} \to \infty$, we set $N_{GOP} = F$, which corresponds to one second's length of frames. Notation $BG_{i,j}$ is used to indicate the remaining bits in the GOP *i* after coding picture *j-1*, equaling to

$$BG_{i,j} = \begin{cases} \min(RG_{i-1} + \frac{R}{F} * N_{GOP}, \frac{R}{F} * N_{GOP} + M * 0.2) & j = 0 \\ BG_{i,j-1} - B'_{i,j-1} & otherwise \end{cases} \quad (5)$$

In the above equation, RG_{i-1} is the number of remaining bits after GOP *i-1* is coded, given by $RG_{i-1} = R/F * N_{coded} - B_{coded}$, where B_{coded} is the used bits and N_{coded} is the number of coded pictures after GOP *i-1* is finished. $B_{i,j}$ and $B'_{i,j}$ is the target bits and actual used bits for frame *j* of GOP *i*, respectively. In equation (5), we add one constraint on the total number of bits allocated for the GOP *i* to prevent buffer overflow when the complexity level of the content varies dramatically from one GOP to another. For example, consider a scenario where the previous GOP was of very low complexity, e.g. all black, so the buffer fullness level would go quite low. Instead of allocating all of the unused bits from the previous GOP to the current GOP, the unused bits are distributed over several following GOPs by not allowing more than *0.2M* additional bits to an individual GOP. The target frame bit $B_{i,j}$ is then allocated according to the picture type. If the *jth* picture is *P*, the initial target bits is $B^P_{i,j} = BG_{i,j}/(K^I N^I + N^P)$, where K^I is the bit ratio between *I* picture and *P* picture, which can be estimated using a sliding window approach, N^I is the remaining number of *I* pictures in GOP *i* and N^P is that of *P* pictures. Otherwise, $B^I_{i,j} = K^I B^P_{i,j}$. Since *P* pictures are used as the references by subsequent *P* pictures in the same GOP, we shall allocate more target bits for *P* pictures that are at the beginning of the GOP to ensure the latter *P* pictures can be predicted from the references of better quality and the coding quality can be improved. We use a linear weighted *P* picture target bit allocation as follows:

$$B^P_{i,j} += R/F * 0.2 * (N_{GOP} - 2j)/(N_{GOP} - 2) \quad (6).$$

An additional constraint is added to better meet the target bits for a GOP as $B_{i,j} = B_{i,j} + 0.1 * B_{diff}$, where B_{diff} is defined as $B_{diff} = B_{i,j-1} - B'_{i,j-1}$, and $B_{diff} = sign(B_{diff}) \min(|B_{diff}|, R/F)$.

In our rate control, we aim at 50% buffer occupancy. To prevent the buffer overflow or underflow, the target bits need to

be jointly adapted with buffer level. The buffer level W is updated at the end of coding each picture by equation (3). In our approach, instead of using real buffer level to adjust the target bits, a virtual buffer level W' given by $W' = \max(W, 0.4M)$ is proposed. This helps prevent the scenario that if the previously coded pictures are of very low complexity such as black scenes and consume very few bits, the buffer level will become very low. If we use the real buffer level to adjust target frame bits as in equation (7), we may allocate too many bits to the frame, which will cause QP to decrease very quickly. But after a while, when the scene returns normal, the low QP will easily cause the buffer to overflow. Hence we need to either increase QP dramatically or skip the frames. This causes the temporal quality to vary significantly. So a constraint is added in our approach for the buffer level, which is used to adjust the bits as

$$B_{i,j} = B_{i,j} * (2M - W') / (M + W') \qquad (7).$$

To guarantee a minimum level of quality, we set $B_{i,j} = \max(0.6 * R / F, B_{i,j})$. To further avoid the buffer overflow and underflow, we set buffer safety top margin W_T and bottom margin W_B for I picture as $W_T^I = 0.75M$ and $W_B^I = 0.25M$. As for P pictures, compliant with equation (6) and to allow enough buffer for the next I picture in the next GOP, we set $W_T^P = (1 - ((0.4 - 0.2)/(N-1) * j + 0.2)) * M$, and $W_B^P = 0.1M$. The final target bits are determined as follows. We set $W_{VT} = W + B_{i,j}$, $W_{VB} = W_{VT} - R/F$. If $W_{VT} > W_T$, $B_{i,j} -= W_{VT} - W_T$, else if $W_{VB} < W_B$, $B_{i,j} += W_B - W_{VB}$. We note that if a scene change detector is employed, we shall encode the picture at the scene change to be an I picture and a new GOP starts from this I picture. The above scheme can still be adopted.

We propose a new scheme to decide frame-level QP based on equation (4). We modify (4) as

$$\hat{Q}_i = \sqrt{\frac{AK}{B - \hat{C}} \frac{\sigma_i}{\alpha_i} \sum_{k=1}^{N} \alpha_k \sigma_k}, \qquad (8)$$

where $\hat{C}$ is the overhead from last coded picture with the same type, σ_i is estimated in the preprocessing stage as in Section 3.1. Two approaches can be used to get frame-level constant QP, denoted as QP_f. The first approach is to set $\alpha_i = \sigma_i$, so that all the MB QPs are equal. The second method is to use the same α_i as that of the MB level, as defined in the next section, then use the mean, median or mode of the histogram of the $\hat{Q}_i$ values to find the QP_f. We adopt the second method to better match the MB QP. The frame-level quantization step size is decided by the mean of the $\hat{Q}_i$ values, $\hat{Q}_f = \sum_{i=1}^{N} \hat{Q}_i / N$. We note that there is an approximate conversion between the quantization parameter QP and quantization step size Q by $Q = 2^{(QP-6)/6}$. To reduce the temporal quality variation between adjacent pictures, we set $QP_f = \max(QP_f' - D_f, \min(QP_f, QP_f' + D_f))$, where QP_f' is the frame QP of last coded picture, and $D_f = \begin{cases} 2 & W < 0.7M \\ 4 & otherwise \end{cases}$. Since scene changes usually cause higher buffer levels, we take advantage of temporal masking effect and set D_f to be a higher value when a scene change occurs.

3.3. MB-layer rate control

The TMN8 MB layer rate control is adapted in our approach [4]. The novelty lies in two aspects. The first aspect is about the adaptive selection of weighted distortion α_i to get a better perceptual quality. The second aspect is to reduce the variation of the MB QPs in the same picture.

For low detail content, such as ocean wave, a lower QP is required to keep the details. But from an RDO point of view, a higher QP is preferred, because the lower detail content tends to give a higher PSNR. To keep a balance, we adopt different settings of α_i for I and P pictures, respectively. For I picture, a higher distortion weight is given to the MBs with less detail, so that the detail can be better retained. Accordingly, we set

$$\alpha_i = (\sigma_i + 2\sigma_{avg}) / (2\sigma_i + \sigma_{avg}), \text{ where } \sigma_{avg} = \sum_{i=1}^{N} \sigma_i / N.$$

For P picture, a higher distortion weight is given to the MBs with more residue errors as in [4]. Accordingly,

$$\alpha_i = \begin{cases} 2B/AN(1-\sigma_i) + \sigma_i, & B/AN < 0.5 \\ 1, & otherwise \end{cases}$$

In this way, better perceptual quality is maintained for I picture and can be propagated to the following P pictures, while higher objective quality is still kept as in [4]. To prevent large variation of the spatial visual quality inside one picture, we set $QP_i = \max(QP_f - 2, \min(QP_i, QP_f + 2))$. If a frame level rate control is used, $QP_i = QP_f$. We should note that even when we use a frame level rate control scheme, we still need to update the parameters as in [4].

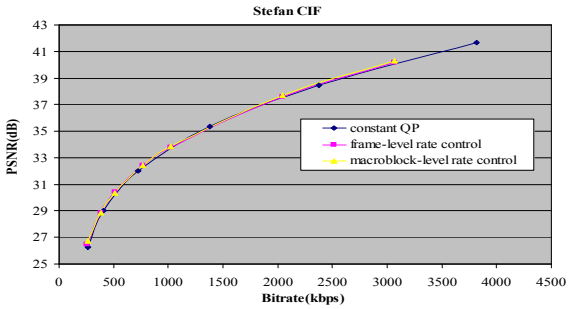

Figure 1 Rate Distortion Curve for Stefan

3.4. Virtual frame skipping

After encoding one picture, we shall update W by equation (3). If $W > 0.9M$, the next frame is virtually skipped until the buffer level is below $0.9M$. Virtual frame skipping is to code every MB in the P picture to be SKIP mode. In this way, we can syntactically keep a constant frame rate. If the current frame is decided to be a virtual skipped frame, we set $QP_f = QP_f' + 2$.

In summary, our rate control scheme consists of the following steps: preprocessing, frame target bits allocation and frame-level constant QP estimation, MB-level QP estimation, buffer updates and virtual frame skipping control. Our approach can allow both frame-level and MB-level rate control.

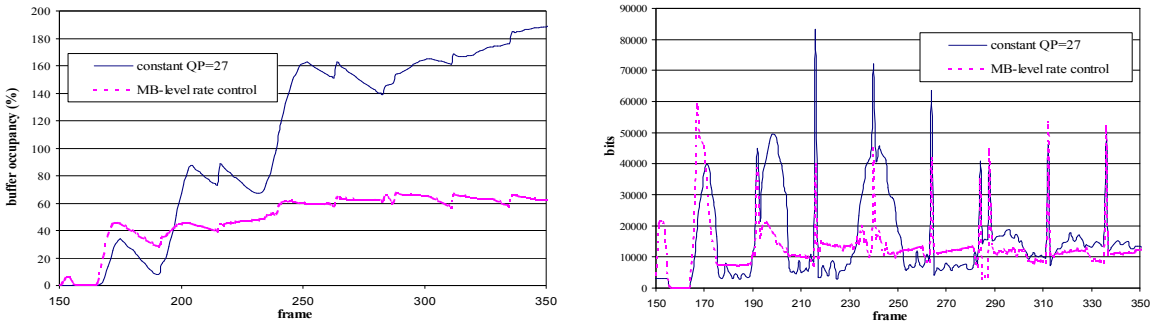

Figure 2 Performance of MB-Level Rate Control for Time-Machine (left: buffer occupancy, right: frame-by-frame bitrate result)

4. EXPERIMENTAL RESULT

We implement our rate control scheme in a custom H.264 encoder, which uses the mode decision and motion estimation described in [5]. Here we show some baseline profile results (IPP…IP) for 3 typical CIF sequences: *News*, *Foreman* and *Stefan*, and for a QVGA movie trailer *Time Machine* [7], which contains many scene transitions. In the simulation, we coded the CIF sequence at 30fps ($F = 30$), with $N_{GOP} = 30$, $M = R$, search range of +/- 32 and 1 reference frame. Table 1 shows that both our frame-level rate control and MB-level rate control scheme can achieve the target bitrate very accurately, with a slight advantage for the MB-level method.

Table 1 Bitrate Achievement of Proposed Scheme

sequence	target bitrate(kbps)	achieved bitrate(kbps)	
		frame	MB
News	64	64.01	63.97
Foreman	128	128.60	128.01
Stefan	384	384.91	384.04
Time-Machine	290	292.80	293.66

During the development of our algorithm, no public H.264 rate control algorithm is available for comparison. The known TMN8 algorithm is not appropriate either, as it targets only on low delay applications. So we compare our result with that using constant QP for the whole sequence. Figure 1 shows the rate distortion curve for *Stefan* with and without rate control. The average PSNR result of both rate control scheme is comparable to that of constant QP. To illustrate the performance of our rate control scheme dealing with various content, Figure 2 illustrates the buffer occupancy and used bits per frame of a movie trailer *Time-Machine* [7] at 270kbps, 24f/s with $N_{GOP} = 24$. Constant *QP=27* has approximately same bitrate over entire sequence. Hence we use it as a basis for comparison. Figure 2 shows one subset of the sequence. Frame 150-164 is an all-white scene, followed by a fade-in from 165-175. Then a dissolve happens from 192-200, then another dissolve from 234-253, followed by a scene change at frame 285 and then panning afterwards. From the figure, we can see that using our rate control scheme, the buffer gradually achieves its stability after the all-white scene. No buffer overflow or frame skipping occurs. Figure 3 compares the visual quality of a zooming part of the ocean wave in an *I* image of *Container* using frame-level rate control and MB-level rate control. We can observe more waves in the image using MB-level rate control that adjusts the MB QP based on the proposed perceptual metrics.

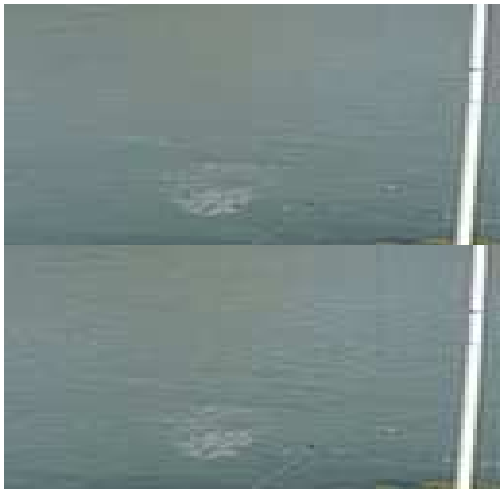

Figure 3 Visual Quality Comparison between Frame-level and MB-level Rate Control of 30th frame for Container CIF, 384kbps, 30fps, GOP size 30 (upper: frame, lower: MB)

5. CONCLUSION

In this paper, we propose a new rate control scheme for H.264 codec. Our approach can achieve the targeted bit rate very accurately, even for content with scene transitions and high complexity variation, while requiring minor additional complexity. In the future work, we shall look into the problem of using look-ahead buffer to further improve the performance.

6. REFERENCES

[1]T. Wiegand, G. J. Sullivan, G. Bjontegaard, and A. Luthra, "Overview of the H.264/AVC Video Coding Standard," *IEEE. Trans. on CSVT*, pp. 560-576, July, 2003.

[2]S. Ma and W. Gao, etc, "Adaptive Rate Control with HRD Consideration," *JVT-H014*, 8th meeting, Geneva, May,2003

[3]Z. He and T. Chen, "Linear Rate Control for JVT Video Coding," *ITRE*, Newark, NJ, 2003.

[4]J. Ribas-Corbera and S. Lei, "Rate Control in DCT Video Coding for Low-Delay Communications", *IEEE Trans. on CSVT*, Feb., 1999.

[5]P. Yin, H. C. Tourapis, A. Tourapis and J. Boyce, "Fast Mode Decision and Motion Estimation for H.264", *ICIP2003*

[6]JM Reference Software version 7.5b, http://bbs.hhi.de/suehring/tml/download

[7]http://www.apple.com/trailers/dreamworks/the_time_machine