

Rate Control for H.264 with Two-Step Quantization Parameter Determination but Single-Pass Encoding

Xiaokang Yang,¹ Yongmin Tan,¹ and Nam Ling²

¹*Institute of Image Communication and Information Processing, Shanghai Jiao Tong University, Shanghai 200030, China*

²*Department of Computer Engineering, Santa Clara University, Santa Clara, CA 95053-0566, USA*

Received 1 August 2005; Revised 27 June 2006; Accepted 16 July 2006

We present an efficient rate control strategy for H.264 in order to maximize the video quality by appropriately determining the quantization parameter (QP) for each macroblock. To break the chicken-and-egg dilemma resulting from QP-dependent rate-distortion optimization (RDO) in H.264, a preanalysis phase is conducted to gain the necessary source information, and then the coarse QP is decided for rate-distortion (RD) estimation. After motion estimation, we further refine the QP of each mode using the obtained actual standard deviation of motion-compensated residues. In the encoding process, RDO only performs once for each macroblock, thus one-pass, while QP determination is conducted twice. Therefore, the increase of computational complexity is small compared to that of the JM 9.3 software. Experimental results indicate that our rate control scheme with two-step QP determination but single-pass encoding not only effectively improves the average PSNR but also controls the target bit rates well.

Copyright © 2006 Hindawi Publishing Corporation. All rights reserved.

1. INTRODUCTION

H.264/MPEG-4 AVC is the latest international video coding standard developed by Joint Video Team (JVT) of ISO Motion Picture Expert Group and ITU-T Video Coding Expert Group [1–5]. As in other video standards such as MPEG-2 [6] and H.263 [7, 8], rate control remains as an open but important issue for H.264/AVC. A rate control scheme that is able to maximize the video quality and at the same time meets the rate constraints is much desired for H.264/AVC.

In comparison with other video standards, there are several challenges for rate control in H.264 [9–12], due to its unique features. The first one is the well-known chicken-and-egg dilemma in the rate-distortion optimization (RDO) process [10], which is briefly described as follows. In H.264, quantization-parameter- (QP-) dependant RDO technique is adopted in the process of best prediction mode selection [11, 13]. To perform RDO, QP should be decided first. But in order to perform rate control, QP can only be obtained according to the coding complexity and number of target bits that are calculated by motion-compensated residues after RDO mode decision. This imposes a big problem for rate control in H.264. Secondly, due to more delicate prediction modes adopted in H.264 than those in previous standards, the number of header bits fluctuates greatly from Inter 16×16 to Inter 4×4 [11, 12]. Thus, a good overhead model

is necessary for accurate rate control. Thirdly, better mode selection in H.264 often leads to small motion-compensated residues [11]. As a result, a large number of macroblocks will be quantized to zero.

Although several rate control algorithms have recently been proposed to cope with these problems [9, 12, 14], the proper method for rate control in H.264/AVC has not been fully explored. A predictive rate control scheme [9] has been adopted in H.264/AVC reference software JM 9.3 [15]. The general idea of the rate control scheme is as follows: after preencoding of the macroblock using the QP of previously encoded macroblock, the block activity is measured by the sum of absolute differences (SAD). Using a linear model that captures the connection between the QP, buffer occupancy, and the block activity, the QP is then determined based on buffer occupancy and block activity. The macroblock is re-encoded using the obtained QP if the difference between the two QPs exceeds a specific threshold. Up to 20% of the MBs need to be encoded twice. Furthermore, linear modeling of the relation between QPs, buffer occupancy, and the block activity may not achieve the best performance. In [12], a solution of the chicken-and-egg dilemma between rate control and RDO in H.264 is given, and hence different bits to different modes are allocated so that the bad situation for the quadratic rate-distortion (RD) model is deviated. Although the solution can keep the peak signal-to-noise ratio (PSNR) smoother than that of [9] and generalized bit rate matches

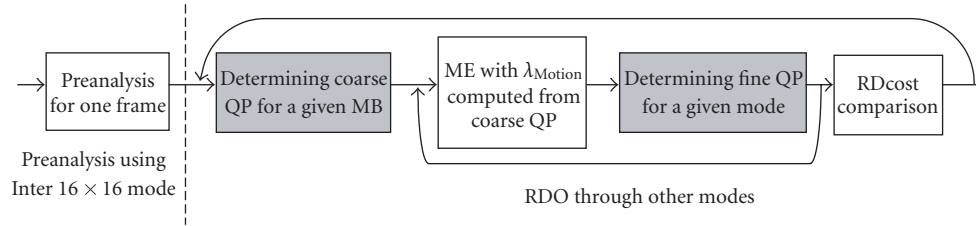


FIGURE 1: Illustration of the basic ideas for the proposed rate control scheme.

the target bit rate accurately, the PSNR improvement is insignificant. In [16–18], a PSNR-and-MAD-based frame complexity estimation is proposed to allocate the bits more accurately among frames. Two special cases of scene change and small texture bits are taken into account when determining QP at frame layer. A frame skipping decision is also used to proactively drop a simple frame in order to make room for the later more complex frames. However, this rate control scheme does not pay much attention to QP determination at the macroblock layer. In [19], a frame-layer rate control scheme is presented, which computes the Lagrange multiplier for mode decision by using a quantization parameter which may be different from that used for encoding.

In this paper, we propose an RDO-based rate control scheme for H.264 with two-step QP determination but single-pass encoding in order to maximize the video quality by appropriately determining QP for each macroblock, which is based on our previous work [11]. To break the chicken-and-egg dilemma resulting from QP-dependent rate-distortion optimization (RDO) in H.264, a pre-analysis phase is conducted to gain the necessary source information, and then the coarse QP is decided for R-D estimation. After QP-dependant motion estimation (with coarse QP), we further refine the QP of each mode based on the obtained actual standard deviation of motion-compensated residues. Using the actual standard deviation, each possible mode's QP can be calculated. Thus, these QPs are used in the comparison of each mode's rate-distortion (RD) cost (RDcost). The encoder chooses the mode having the minimum value. Thus, carefully selected QPs can ensure accurate bits allocation to individual MBs according to their actual needs. The introduction of QP refinement process is helpful to achieve a good video quality given the bit budget. In addition, the header bits and coefficient bits are separately estimated so that the rate control accuracy is further enhanced. In the encoding process, RDO only performs once for each macroblock, thus one-pass, while QP determination is conducted twice. Therefore, the increase of computational complexity is small compared to that of the JM 9.3 software. Experimental results indicate that our rate control scheme not only effectively improves the average PSNR but also controls the target bit rates well.

The rest of this paper is organized as follows. In Section 2, we derive models for bit rate and distortion estimation. In Section 3, our proposed rate control algorithm is presented

in detail, including the solutions to the aforementioned difficulties and the two-step QP decision with single-pass encoding. Section 4 gives experimental results. Finally, Section 5 concludes the paper.

2. MODELING RATE AND DISTORTION

Figure 1 shows the basic ideas of the overall rate control process of our algorithm, which comprises of two major steps. Firstly, pre-analysis is performed to break the chicken-and-egg dilemma, thus obtaining the source information, which is used in determining the coarse QP for QP-dependent motion estimation. Secondly, RDO mode decision is conducted at the macroblock layer to select the best prediction mode for individual macroblock. The refined QP of each possible mode is determined and used in the RDcost comparison. After RDO, current macroblock is encoded with the selected mode and its corresponding refined quantization parameter.

To determine QP, an R-D model usually estimates the rate and distortion based on some measurements of frames or macroblocks. In this paper, we choose the R-D model of our previous work [11] in which the header bits, the coefficient bits, and distortion of each macroblock are estimated. They are briefly described as follows.

2.1. Preanalysis using Inter 16×16 mode header bits estimation

Pre-analysis phase is performed by motion estimation for Inter 16×16 mode. To break the chicken-and-egg dilemma in order to get the required information, all MBs in current frame are preencoded before the RDO mode decision. Among the possible seven modes (i.e., Intra 4×4 , Intra 16×16 , Skip, Inter 16×16 , Inter 16×8 , Inter 8×16 , and Inter 8×8), we choose the simplest Inter 16×16 to perform preanalysis. After this preanalysis, the source information, such as the standard deviations of motion-compensated residues, RDcost of each macroblock for Inter 16×16 mode, is obtained. These measurements are used in the R-D model to decide the number of target bits for every frame and the coarse QP for individual macroblock.

In this implementation, the QP for preanalyzing the first inter-predicted frame is the same as the fixed QP set in configuration file of JM 9.3 for each encoding. In other

inter-predicted frames, the average QP from all MBs of the previously inter-predicted frame is used to preanalyze current frame.

2.2. Header bits estimation

Most existing R-D models only consider the transform coefficient bits in the estimation of the rate for a macroblock. Header bits are simply represented by a constant value. This is a reasonable simplification for previous standards such as MPEG-2 and H.263, because the header bits are relatively few in number due to the simplicity of prediction modes in these standards. However, header bits form a significant portion of H.264/AVC bitstream [11]. Therefore, the number of header bits needs to be estimated separately from coefficient bits for accurate rate estimation. In this paper, we use the following simple but effective model to estimate the number of header bits for one macroblock:

$$H_i = C \times \text{com}_i \quad (1)$$

with

$$\text{com}_i = \begin{cases} \frac{H_{\text{trd}}}{C}, & \sigma_i^2 \leq \sigma_{\text{trd}}^2, \\ (\log(\sigma_i^2))^2, & \text{else,} \end{cases} \quad (2)$$

where H_i is the number of header bits for the i th macroblock in the current frame. σ_i is the predicted standard deviation of motion-compensated residues for Inter 16×16 . In the following, we refer to the standard deviation of the motion-compensated residue obtained in the pre-analysis phase as *predicted standard deviation* since it may be different from the *actual standard deviation* if RDO selects other mode rather than the Inter 16×16 mode as the prediction mode. H_{trd} and σ_{trd}^2 are the averages of all recorded H_i and σ_i^2 , which are explained below. C is a constant that implies the linear relation between H_i and com_i , which is used to separate the following two situations so that (1) looks more compact.

Two situations are considered in our header bits model. (1) When encoding the previous frame, we record H_i and σ_i^2 of the MBs whose H_i is smaller than a predefined constant ($= 11$). After encoding the previous frame, we calculate the averages of all recorded H_i and σ_i^2 , which are referred to as H_{trd} and σ_{trd}^2 , respectively. During the encoding of current frame, if $\sigma_i^2 \leq \sigma_{\text{trd}}^2$ for a macroblock, we then conclude that this macroblock will produce a small number of header bits and H_i is directly estimated by H_{trd} . (2) Otherwise, the number of header bits of a macroblock is linear to $[\log(\sigma_i^2)]^2$. Furthermore, C is adaptively updated macroblock by macroblock during the encoding process to make the model more robust, which is discussed below. Further explanation of (1) and (2) is given as follows.

We use Inter 16×16 mode in the pre-analysis to compute the motion-compensated residues. A good prediction of the MB by Inter 16×16 will result in a small predicted standard

deviation. So the chances are that Inter 16×16 will be selected as the best prediction mode. In contrast, a large predicted standard deviation implies a bad prediction and RDO may quite possibly select other modes such as Intra 4×4 or Inter 8×8 to do the prediction. In this sense, the prediction mode selected by RDO is, to some extent, dependent on the predicted standard deviation. On the other hand, as we know, in H.264, the number of header bits strongly depends on its prediction mode (e.g., Inter 16×16 has only one motion vector while Inter 8×8 may have up to 16 motion vectors). From the above analysis, we can say that the number of header bits depends on the predicted standard deviation as well. The larger the predicted standard deviation, the higher the possibility that header-bits-expensive modes, such as Inter 8×8 , will be used. In other words, the number of header bits increases with the predicted standard deviation, as is suggested by (2).

2.3. Coefficient bits estimation

The rate-quantization model proposed in [21] is used to estimate the coefficient bits estimation:

$$F_i = AK \frac{\sigma_i^2}{Q_i^2}, \quad (3)$$

where F_i denotes the bit required for encoding the DCT coefficients of i th macroblock; σ_i denotes the standard deviation of motion-compensated residues; Q_i is the quantization step size; A is the number of the pixels in a macroblock (i.e., $16 \times 16 = 256$); K is a constant and can be set to $e/\ln 2$ if the DCT coefficients are Laplacian distributed and independent [21]. However, since the DCT coefficients may not follow the Laplacian distribution strictly, it is better to adaptively update the value of K , macroblock by macroblock and frame by frame. More details are discussed in the Section 3.3.

2.4. Distortion estimation

The following well-known distortion-quantization model [15] is used to measure the distortion of encoded macroblocks:

$$D = \frac{1}{N} \sum_{i=1}^N \alpha_i^2 \frac{Q_i^2}{12}, \quad (4)$$

where N is the total number of macroblocks in one frame; α_i is distortion weight of i th macroblock, which can be used to incorporate the importance or weight of that macroblock's distortion. However, in this implementation, these weights are used to reduce the bit overhead caused by recording each macroblock's QP individually at low bit rates.

If the values of QP for sequential macroblocks are differentially encoded in a raster-scan order, frequent QP changes between macroblocks consume too many bits. This effect is negligible at high bit rates but may become increasingly

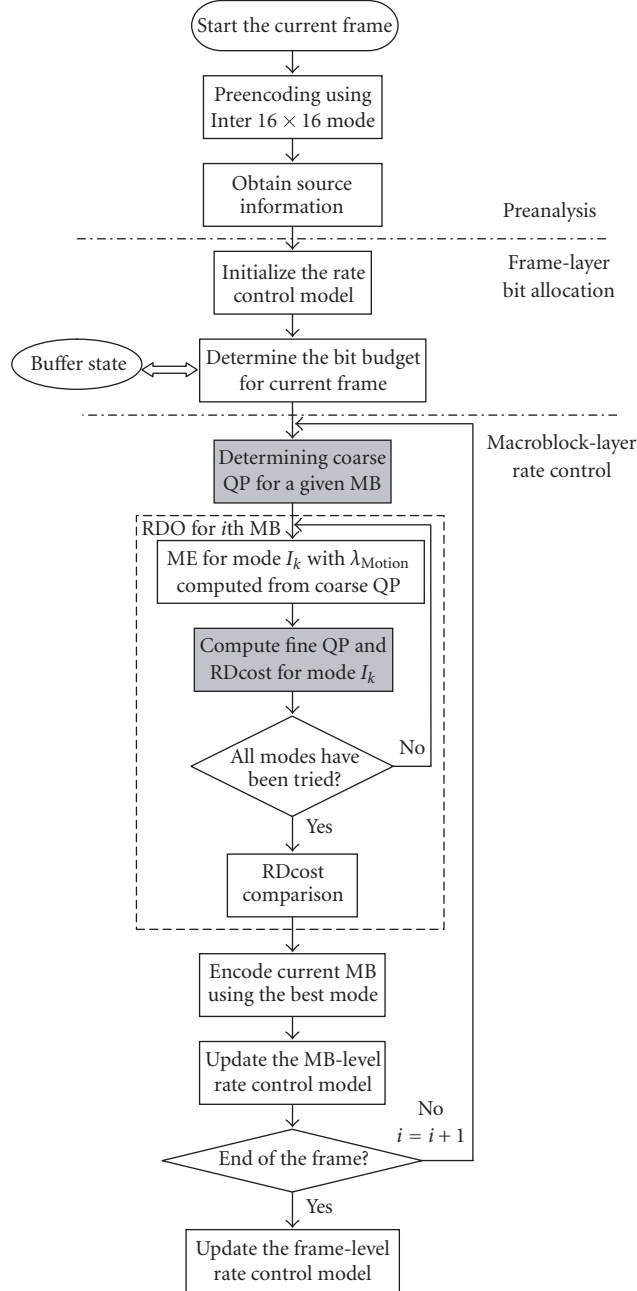


FIGURE 2: A flowchart of the proposed rate control scheme.

significant at low bit rates. We therefore try to control the dynamic range of QP by simply setting the values of α_i . At lower bit rates, α_i is determined from the respective standard deviation of residues σ_i by the method proposed in [15]. At higher bit rates (above 0.5 bits/pixel), all of α_i are set to 1.

3. OUR PROPOSED RATE CONTROL SCHEME

Figure 2 shows the flowchart of the proposed rate control scheme. The three major steps are the above-mentioned pre-

analysis, frame-layer bit allocation, and macroblock-layer rate control.

3.1. Pre-analysis

Through pre-analysis using Inter 16×16 mode, we obtain the necessary source information for R-D estimation before the RDO. The predicted information is used to determine the bit budget for frames and the coarse QPs for macroblocks.

3.2. Frame-layer bit allocation

In [9], a fluid flow traffic model was proposed to compute the target bit for the current coding frame. Although this model can achieve accurate bit-rate control, it only considers the buffer states (or rate) but without the consideration of distortion, thus may limit the quality improvement. In our previous work [11], we proposed a frame-layer bit allocation scheme by integrating both rate-distortion cost and target bit rate. The scheme can be divided into two steps.

First, we determine the number of target bits for current frame without considering the buffer state using the following equation:

$$B = \left[1 + \frac{\hat{P} - P_n}{2} \right] \times \frac{J_{\text{cur}} - J_{\text{prev},0}}{\hat{f} - J_{\text{prev},0}} \times \frac{R}{f}, \quad (5)$$

where R is the available channel bandwidth. f is the frame rate. J_{cur} is the RDcost of current frame, which is defined as the sum of the RDcost of all the MBs in the current frame. It is noticed that macroblock-layer rate control is still not enabled at this moment. Remembering that in the pre-analysis stage we use Inter 16×16 mode for pre-encoding, so J_{cur} is actually the RDcost of current frame under the Inter 16×16 mode. \hat{f} is the average RDcost of the encoded frames in the group of pictures (GOP), the GOP size is 100 frames. $J_{\text{prev},0}$ is the sum of RDcost of all the zero-coefficient macroblocks in the previous frame. Zero-coefficient macroblock refers to a macroblock whose coefficients are all quantized to zeros after the transform and quantization. P_n is the average PSNR of the recent n frames, which is computed using a sliding window (length is 8) method. \hat{P} is the average PSNR of the encoded frames again in the GOP.

Second, the target number of bits for a frame is further adjusted according to the buffer state in a similar way to the fluid flow traffic model [11, 20]:

$$B = \begin{cases} \frac{R}{f} + \lambda_1 \left(B - \frac{R}{f} \right), & B > \frac{R}{f} \ \& \ L > 0.2M, \\ \frac{R}{f} + \lambda_2 \left(B - \frac{R}{f} \right), & B < \frac{R}{f} \ \& \ L < 0.2M, \end{cases} \quad (6)$$

where M is the buffer size and L is the currently observed buffer fullness. The strength of the restriction depends on the parameters of λ_1 and λ_2 , which are determined from the normalized buffer fullness (L/M) via

$$\begin{aligned} \lambda_1 &= \frac{0-1}{1-0.2} \times \left(\frac{L}{M} - 0.2 \right) + 1 \quad \left(0.2 \leq \frac{L}{M} \leq 1 \right), \\ \lambda_2 &= \frac{1-0}{0.2-0} \times \left(\frac{L}{M} - 0.2 \right) + 1 \quad \left(0 \leq \frac{L}{M} \leq 0.2 \right). \end{aligned} \quad (7)$$

As we can see, λ_1 and λ_2 linearly range from 0 to 1 according to the current buffer state. The two functions converge at point (0.2, 1), which means that there is no constraint imposed when L/M is 0.2. On the other hand, stronger restriction is imposed when the buffer level is extremely high or low. It should be noticed that these controlling points of linear function can be adjusted to meet the variant requirement and buffer condition.

3.3. Macroblock-layer rate control

3.3.1. Determining coarse QP

We mainly focus our discussion on the low delay situation where the macroblock-layer rate control is more critical. We consider the IPPP...GOP structure. The most crucial task of macroblock-layer rate control is to determine the QP for every individual macroblock. For I frame, the method in the JM 9.3 reference software is also used to determine the QPs in this implementation. In the following, we only discuss the QP determination for P frames.

The optimized quantization step size Q_i^* for i th MB can be determined by minimizing the overall distortion D subject to a given bit budget B , namely, minimizing the RDcost as follows:

$$\begin{aligned} \text{cost} &= D + \lambda \left[\sum_{i=1}^N (F_i + H_i) - B \right] \\ &= \frac{1}{N} \sum_{i=1}^N \alpha_i^2 \frac{Q_i^2}{12} + \lambda \left[\sum_{i=1}^N \left(AK \frac{\sigma_i^2}{Q_i^2} + C \times \text{com}_i \right) - B \right]. \end{aligned} \quad (8)$$

This kind of optimization problem can be solved by Lagrangian optimization technique [21]:

$$Q_i^* = \sqrt{\frac{AK_{i-1}}{B_i - C_i \sum_{j=i}^N \text{com}_j} \frac{\sigma_i}{\alpha_i} \sum_{j=i}^N \alpha_j \sigma_j}. \quad (9)$$

It is noticed that σ_i in the equation is the standard deviation of motion-compensated residues of the Inter 16×16 mode in the pre-analysis phase. Formula (9) is used to compute the coarse QP of each macroblock. The parameters K_{i-1} and C_i are recursively updated (MB by MB) during the encoding of the successive macroblocks; more details are given in Section 3.3.5.

From (9), we can see that if α_i approaches σ_i very closely, the term σ_i/α_i becomes 1 and thus all of the quantization steps in one frame are approximately equal. The range of QP is then reduced. So it gives a good explanation to the aforementioned distortion weights determination.

3.3.2. Motion estimation

The resultant Q_{Coarse} (i.e., Q_i^*) and $\lambda_{\text{Motion}} = 0.85 \times 2^{(Q_{\text{Coarse}} - 12)/3}$ are used in motion estimation to search for the best motion vectors for each macroblock under a certain mode.

3.3.3. Quantization parameter refinement

From Section 2, we know that the coefficient model is based on the actual standard deviation of the motion-compensated residues. Clearly, the standard deviation obtained in the pre-analysis may be different from the actual standard deviation if the RDO process selects another prediction mode rather than Inter 16×16 . This will result in some error of QP calculation to some extent, especially for high-motion videos and

their high bit rates because there are fewer chances for Inter 16×16 to be selected in such situation.

We observe that for mode I_k , the standard deviation of motion-compensated residues $\sigma_i^*(I_k)$ can be obtained easily after motion estimation (ME) in the loop of the RDO process. Then, the QP of each mode, denoted as QP_{I_k} , can be calculated using (9), where we just replace σ_i with $\sigma_i^*(I_k)$. After all modes are checked by RDO, the encoder uses QP_{I_k} in the comparison of RDcost to choose a best prediction mode (I_{best}) for the current macroblock.

3.3.4. Encoding of MBs using the best mode

To encode the i th macroblock with the best mode I_{best} , we define $S_i = \sum_{j=i}^N \alpha_j \sigma_j$, $T_i = \sum_{j=i}^N \text{com}_j$ and rewrite (9) as follows:

$$Q_i(I_{\text{best}}) = \sqrt{\frac{AK_i}{B_i - C_i T_i} \frac{\sigma_i^*(I_{\text{best}})}{\alpha_i}} S_i, \quad (10)$$

where B_i is the unused number of target bits for the remaining macroblocks from i th to N th in the current frame. K_i and C_i are the updated values of R-D model parameters K and C after encoding the first $(i - 1)$ macroblocks. In this way, we can compute the QPs of each macroblock via updating the required parameters macroblock by macroblock when the macroblocks are processed sequentially in one frame.

3.3.5. Updating some parameters of R-D model

(1) Updating B_i

B_{i+1} is updated as follows:

$$B_{i+1} = \left(B - \sum_{j=1}^i b_j \right) \times \frac{N-i}{N} + \left(\frac{\sum_{j=i+1}^N J_j}{\sum_{j=1}^i J_j} \times \sum_{j=1}^i b_j \right) \times \frac{i}{N}, \quad (11)$$

where J_j is the R-D cost of j th macroblock obtained in the pre-analysis stage; b_j is the actual number of encoding bits used for j th macroblock. We adopt the weighted average method to improve the accuracy and robustness of bit allocation. On the right-hand side of the equation, the first term indicates the unused bit budget for the remaining macroblocks to be encoded while the second term is to update the bit allocation according to the actual R-D cost of the macroblocks. Such updating according to the actual encoding results is necessary during the scan over all macroblocks.

(2) Updating K_i

- (a) Compute the K'_i after encoding the current macroblock:

$$K'_i = \frac{F_i \times (Q_i^*)^2}{256\sigma_i^2}. \quad (12)$$

- (b) If $K'_i > 0$ and $K'_i \leq 4.5$, compute the average K of the macroblocks encoded so far:

$$\bar{K}_i = \frac{\bar{K}_{i-1}(l-1) + K'_i}{l}, \quad (13)$$

where l is the number of macroblocks encoded so far whose K'_i is within $[0, 4.5]$.

Otherwise, we regard the current value of K'_i as an ineffective estimation and just skip this step. So \bar{K}_i remains unchanged after encoding the current macroblock in this situation.

- (c) Find the weighted average of the initial estimate K_1 with \bar{K}_i :

$$\bar{K}_i = \bar{K}_i \left(\frac{i}{N} \right) + \frac{K_1(N-i)}{N}, \quad (14)$$

where K_1 is the average K of the previous frame. It is used to improve the accuracy of the estimation of K , since when only the first few macroblocks in the current frame have been encoded (i.e., i is small), \bar{K}_i is the average of only a few values and hence is not a robust estimate of K for the current frame. Then the updated \bar{K}_i is used in (9) and (10).

(3) Updating C_i

- (a) Compute the C'_i after encoding the current macroblock:

$$C'_i = \frac{\sum_{j=1}^i (b_j - F_j)}{\sum_{j=1}^i \text{com}_j}, \quad (15)$$

where $\sum_{j=1}^i (b_j - F_j)$ is the total number of header bits used for encoding the first i macroblocks.

- (b) Find the average C''_i of all the encoded macroblocks in the current frame:

$$C''_i = C''_{i-1} \times \frac{i-1}{i} + C'_i \times \frac{1}{i}. \quad (16)$$

- (c) Find the weighted average of the initial estimate C_1 with C''_i :

$$C'_i = C''_i \times \frac{i}{N} + C_1 \times \frac{N-i}{N}, \quad (17)$$

where C_1 is the average C of the previous frame. This method of weighted average is used for the same reason as (14). Then the updated C'_i is used in (9) and (10).

3.3.6. Implementation issue related to RDO options

When our scheme was integrated into the JM 9.3 software, two different situations were considered: RDO on and RDO off (whether to apply RDO technique in mode decision process or not), which led to a little difference in the realization of our algorithm.

(1) RDO off

When the RDO option was switched to off, it implied that RDcost value comparison was not conducted for mode decision. Only the values of SAD or SATD (when Hadamard

transform was set) for each mode were compared to select the best prediction mode. Therefore, we just examined the standard deviation of motion-compensated errors for the best mode and updated its QP.

(2) RDO on

It was more complicated when the RDO option was switched to on. The mean absolute difference (MAD) for each mode should be calculated in order to perform QP refinement. Firstly, motion estimation was performed. All modes were checked in order. Motion estimations for Inter 16×16 , 8×16 , and 16×8 were performed in one loop, then Inter 8×8 with transform size 8×8 , and lastly Inter 8×8 with transform size 4×4 (8×8 , 8×4 , 4×8 , 4×4 partitions). The motion vectors and reference frames of each mode were decided in the motion search process. We used them to obtain the MAD of each mode. Then, the QP of each mode was easily calculated according to our algorithm. Secondly, RDcost value comparison was performed to get the best macroblock mode, where we used each mode's refined QP instead of coarse QP.

It was noticed that RDO technique was already used in the loop over 8×8 subpartitions with transform size 4×4 . For all four 8×8 subblocks in a 16×16 macroblock, the best block mode should be decided among modes 4, 5, 6, and 7 (8×8 , 8×4 , 4×8 , 4×4) through the comparison of RDcost value. After that, some variables were updated if the best mode had been changed. Therefore we also applied our algorithm here. Similarly, we obtained the MAD of 8×8 subblock and then introduce the small-sized refined QP for RDcost comparison. For QP refinement, the QP range was restricted in a reasonable range, that is, the coarse QP ± 4 to prevent too high QP fluctuation between neighboring macroblocks.

Another issue was how many parameters of the rate control model in (9) should be updated with different modes. In fact, many model variables were associated with the standard deviation of motion-compensated residues $\sigma_i^*(I_k)$. But we believed that there was no need to modify them because they were less dominative than $\sigma_i^*(I_k)$ in deciding the refined QP. Another reason was that most of these variables were introduced in the pre-analysis phase at the frame layer, such as the number of target bits and the number of header bits. Though these parameters had some errors if we did not recalculate them, it was also unsuitable to update them at the macroblock layer during the encoding process. Hence we only traced the change of each mode's MAD and ignored other parameters that had indirect relations with the standard deviations of motion-compensated residues. So in our implementation, the only difference between (9) and (10) is $\sigma_i^*(I_k)$.

In the encoding process, the QP calculation is conducted twice in all. First, coarse QP is obtained to compute the Lagrange multiplier parameter for motion estimation. Second, QPs are further refined for different modes, which are used for R-D cost comparison in the RDO process. The final QP of the macroblock (i.e., the best mode's corresponding refined QP) becomes more accurate and conforms to the actual R-D performance of the macroblock for more effective

TABLE 1: Test sequences.

Test sequence	Size	Frame rate	QP range	Sequence length	Frames encoded	Frame type
Carphone	QCIF	30	20–44	382	100	IPPP
News	QCIF	30	20–44	300	100	IPPP
Foreman	QCIF	30	20–44	300	100	IPPP
Silent	QCIF	30	20–44	300	100	IPPP
Mother_daughter	QCIF	30	20–44	300	100	IPPP
Salesman	QCIF	30	20–44	449	100	IPPP
Paris	CIF	30	20–44	1065	150	IPPP
Stefan	CIF	30	20–44	300	150	IPPP
City	D1	30	20–44	300	100	IPPP

TABLE 2: Test conditions.

MV resolution	1/4 pel
Hadamard	ON
RDO	OFF/ON
Search range	16
Restricted search range	2
Reference frame	5
Symbol mode	CABAC
Slice mode	OFF
Frame skip	2

and accurate rate control. The RDO process does not need to be performed again like that in JVT-F086 [22], hence we call it two-step QP determination but single-pass encoding.

3.3.7. Computational complexity analysis

The possible computational complexity overhead of our method may come from the pre-analysis stage where the Inter 16×16 mode is performed to obtain the source information. However, since the results obtained in pre-analysis can be stored for use in the following RDO process, there is no need to implement Inter 16×16 again during the RDO. Thus, pre-analysis will only change the algorithm flow and the overall computational complexity has only a possibly negligible increase when RDO option is switched on. As for the RDO off situation, the encoding complexity increases about 30% in terms of the total encoding time.

4. RESULTS AND DISCUSSIONS

The proposed rate control scheme was implemented onto the H.264 JM 9.3 encoder [23]. In this section, nine typical sequences of various resolution sizes and motion measurements were tested as listed in Table 1. The encoder configuration is shown in Table 2. The performance of our proposed scheme is evaluated in comparison with the original encoder JM 9.3 and the existing rate control functionality in the JM 9.3. We also compared the proposed approach with the approach that does not refine the QP for mode decision. In the

TABLE 3: Performance comparison (QP for FQP is 44 and the first I frame QP for rate control schemes is 40, RDO on).

Test Sequence	Scheme	PSNR-Y (dB)	QP	R (bps)	GAIN (dB)	ΔR (%)
Carphone	JM 9.3 FQP	26.46	44	7430	—	—
	JM 9.3 RC	26.88	40	7540	0.42	1.48%
	PRC w/o QP refinement	27.06	40	7520	0.6	1.21%
	RC with QP refinement	27.36	40	7620	0.9	2.56%
News	JM 9.3 FQP	25.45	44	5890	—	—
	JM 9.3 RC	26.12	40	5960	0.67	1.19%
	PRC w/o QP refinement	26.64	40	5820	1.19	-1.19%
	RC with QP refinement	26.81	40	5920	1.36	0.51%
Silent	JM 9.3 FQP	25.92	44	5050	—	—
	JM 9.3 RC	26.94	40	5090	1.02	0.79%
	PRC w/o QP refinement	26.9	40	4890	0.98	-3.17%
	RC with QP refinement	27.11	40	5130	1.19	1.58%
Mother_daughter	JM 9.3 FQP	27.85	44	2600	—	—
	JM 9.3 RC	28.09	40	2580	0.24	-0.77%
	PRC w/o QP refinement	28.39	40	2590	0.54	-0.38%
	RC with QP refinement	28.59	40	2640	0.74	1.54%
Salesman	JM 9.3 FQP	25.55	44	2800	—	—
	JM 9.3 RC	26.1	40	2800	0.55	0.00%
	PRC w/o QP refinement	26.22	40	2840	0.67	1.43%
	RC with QP refinement	26.46	40	2890	0.91	3.21%
Foreman	JM 9.3 FQP	26.01	44	9990	—	—
	JM 9.3 RC	25.89	40	10060	-0.12	0.70%
	PRC w/o QP refinement	26.01	40	9830	0	-1.60%
	RC with QP refinement	26.22	40	9920	0.21	-0.70%
Paris	JM 9.3 FQP	24.15	44	28630	—	—
	JM 9.3 RC	25.2	40	28790	1.05	0.56%
	PRC w/o QP refinement	25.02	40	28210	0.87	-1.47%
	RC with QP refinement	25.23	40	28320	1.08	-1.08%
Stefan	JM 9.3 FQP	24.14	44	72080	—	—
	JM 9.3 RC	24.13	40	72270	-0.01	0.26%
	PRC w/o QP refinement	24.17	40	71840	0.03	-0.33%
	RC with QP refinement	24.33	40	72130	0.19	0.07%
City	JM 9.3 FQP	26.16	44	68680	—	—
	JM 9.3 RC	25.69	40	69000	-0.47	0.47%
	PRC w/o QP refinement	25.16	40	67510	-1	-1.70%
	RC with QP refinement	25.44	40	68030	-0.72	-0.95%

simulation, we first encoded the sequence using fixed quantization parameter to determine the target bit rate. Then the same video was encoded once again using the rate control scheme in JM 9.3 and our rate control algorithm, respectively. The obtained PSNRs and the bit rates are compared.

We adopt the method in [20] to determine the starting quantization parameter QP_0 . It is predefined based on the available channel bandwidth and the GOP length. In our implementation, the QP for the first I frame is 4 lesser than that for the fixed-QP scheme. The same starting QP is used in the JM 9.3 rate control scheme for a fair comparison of PSNR.

Tables 3 to 6 list the comparison of the experimental results among JM 9.3 rate control (RC), the proposed rate

control without QP refinement (PRC w/o QP refinement), and the proposed rate control with QP refinement (PRC with QP refinement). We analyzed the performances of these three rate control schemes with JM 9.3 fixed QP (FQP) as benchmark, where each of the video sequences was encoded at seven different bit rates with JM 9.3 for fixed QPs ranged from 20 to 44 (the QPs were kept unchanged for all the frames). For the other three rate control schemes, the QPs in the tables were only used for I frames and the QPs in P frames were dynamically adjusted by the aforementioned algorithm during the encoding process. R is the overall bit rate.

As observed from Tables 3 to 6, our rate control scheme with QP refinement outperforms the existing rate control

TABLE 4: Performance comparison (QP for FQP is 36 and the first I frame QP for rate control schemes is 32, RDO on).

Test sequence	Scheme	PSNR-Y (dB)	QP	R (bps)	GAIN (dB)	ΔR (%)
Carphone	JM 9.3 FQP	31.5	36	21790	—	—
	JM 9.3 RC	31.64	32	21930	0.14	0.64%
	PRC w/o QP refinement	31.91	32	21730	0.41	-0.28%
	RC with QP refinement	32.09	32	21750	0.59	-0.18%
News	JM 9.3 FQP	30.95	36	16300	—	—
	JM 9.3 RC	30.98	36	16400	0.03	0.61%
	PRC w/o QP refinement	31.91	32	16030	0.96	-1.66%
	RC with QP refinement	31.98	32	16050	1.03	-1.53%
Silent	JM 9.3 FQP	30.63	36	14990	—	—
	JM 9.3 RC	31.5	32	15060	0.87	0.47%
	PRC w/o QP refinement	31.49	32	14680	0.86	-2.07%
	RC with QP refinement	31.63	32	14790	1	-1.33%
Mother_daughter	JM 9.3 FQP	32.44	36	7660	—	—
	JM 9.3 RC	32.17	32	7730	-0.27	0.91%
	PRC w/o QP refinement	32.38	32	7590	-0.06	-0.91%
	RC with QP refinement	32.49	32	7740	0.05	1.04%
Salesman	JM 9.3 FQP	30.1	36	9600	—	—
	JM 9.3 RC	30.67	32	9680	0.57	0.83%
	PRC w/o QP refinement	30.79	32	9390	0.69	-2.19%
	RC with QP refinement	30.96	32	9500	0.86	-1.04%
Foreman	JM 9.3 FQP	30.86	36	24940	—	—
	JM 9.3 RC	30.69	32	25010	-0.17	0.28%
	PRC w/o QP refinement	30.68	32	24390	-0.18	-2.21%
	RC with QP refinement	30.82	32	24660	-0.04	-1.12%
Paris	JM 9.3 FQP	29.6	36	96880	—	—
	JM 9.3 RC	30.34	32	97390	0.74	0.53%
	PRC w/o QP refinement	30.62	32	95640	1.02	-1.28%
	RC with QP refinement	30.82	32	96210	1.22	-0.69%
Stefan	JM 9.3 FQP	29.22	36	279360	—	—
	JM 9.3 RC	29.08	32	279380	-0.14	0.01%
	PRC w/o QP refinement	29.19	32	278920	-0.03	-0.16%
	RC with QP refinement	29.38	32	279840	0.16	0.17%
City	JM 9.3 FQP	30.54	36	197580	—	—
	JM 9.3 RC	29.86	32	198490	-0.68	0.46%
	PRC w/o QP refinement	29.86	32	189680	-0.68	-4.00%
	RC with QP refinement	30.08	32	192870	-0.46	-2.38%

functionality in JM 9.3 in terms of PSNR in most cases. The average PSNR improvement is 0.63 dB over JM 9.3 FQP, and 0.28 dB over JM 9.3 RC for the 36 experiments when RDO was on, while the bit rate inaccuracy is less than 2%. Besides, we can also obviously see the significant effect of QP refinement step adopted in our scheme. The average gain is

0.25 dB compared to the approach without QP refinement for mode decision. The tables only list the PSNRs of the luminance component. In fact, the PSNRs of the two chrominance components are improved much more than that of the luminance component. Similar experimental results have been achieved in the case of “RDO off,” but, however, are not

TABLE 5: Performance comparison (QP for FQP is 28 and the first I frame QP for rate control schemes is 24, RDO on).

Test sequence	Scheme	PSNR-Y (dB)	QP	R (bps)	GAIN (dB)	ΔR (%)
Carphone	JM 9.3 FQP	36.91	28	69054	—	—
	JM 9.3 RC	37.34	24	69340	0.43	0.41%
	PRC w/o QP refinement	37.23	24	68010	0.32	-1.51%
	RC with QP refinement	37.32	24	68560	0.41	-0.72%
News	JM 9.3 FQP	36.84	28	45350	—	—
	JM 9.3 RC	37.12	24	45520	0.28	0.37%
	PRC w/o QP refinement	37.56	24	44360	0.72	-2.18%
	RC with QP refinement	37.82	24	44544	0.98	-1.78%
Silent	JM 9.3 FQP	35.83	28	44238	—	—
	JM 9.3 RC	37.2	24	44350	1.37	0.25%
	PRC w/o QP refinement	37.15	24	43050	1.32	-2.69%
	RC with QP refinement	37.3	24	43190	1.47	-2.37%
Mother_daughter	JM 9.3 FQP	37.63	28	25615	—	—
	JM 9.3 RC	37.62	24	25820	-0.01	0.80%
	PRC w/o QP refinement	37.64	24	25440	0.01	-0.68%
	RC with QP refinement	37.79	24	25560	0.16	-0.21%
Salesman	JM 9.3 FQP	35.6	28	30067	—	—
	JM 9.3 RC	36.51	24	30190	0.91	0.41%
	PRC w/o QP refinement	36.7	24	29880	1.1	-0.62%
	RC with QP refinement	36.96	24	30590	1.36	1.74%
Foreman	JM 9.3 FQP	36.08	28	68941	—	—
	JM 9.3 RC	36.35	24	68970	0.27	0.04%
	PRC w/o QP refinement	36.05	24	67840	-0.03	-1.60%
	RC with QP refinement	36.17	24	68050	0.09	-1.29%
Paris	JM 9.3 FQP	35.61	28	297250	—	—
	JM 9.3 RC	36	24	298330	0.39	0.36%
	PRC w/o QP refinement	36.74	24	293120	1.13	-1.39%
	RC with QP refinement	36.97	24	294440	1.36	-0.95%
Stefan	JM 9.3 FQP	35.33	28	951880	—	—
	JM 9.3 RC	35.2	24	951570	-0.13	-0.03%
	PRC w/o QP refinement	34.94	24	944390	-0.39	-0.79%
	RC with QP refinement	35.18	24	947620	-0.15	-0.45%
City	JM 9.3 FQP	35.77	28	854440	—	—
	JM 9.3 RC	35.53	24	854750	-0.24	0.04%
	PRC w/o QP refinement	35.24	24	829920	-0.53	-2.87%
	RC with QP refinement	35.48	24	840530	-0.29	-1.63%

presented in this paper to save the page space. Figures 3 and 4 show frame-by-frame PSNR curve comparison in the encoding process for “Salesman” and “Paris” in the case of “RDO on.”

Interestingly, our scheme is relatively more effective for the sequences tested with low bit rates and low motion be-

cause Inter 16×16 mode is more likely to be selected by RDO in such situations. Thus, the inaccuracies resulted from the inconsistency of different prediction modes in the pre-analysis stage and RDO stage are avoided as much as possible. But thanks to the QP refinement algorithm, the performances of those high motion and high bit rate sequences are

TABLE 6: Performance comparison (QP for FQP is 20 and the first I frame QP for rate control schemes is 16, RDO on).

Test sequence	Scheme	PSNR-Y (dB)	QP	R (bps)	GAIN (dB)	ΔR (%)
Carphone	JM 9.3 FQP	42.83	20	172570	—	—
	JM 9.3 RC	42.65	16	172710	-0.18	0.08%
	PRC w/o QP refinement	42.42	16	172590	-0.41	0.01%
	RC with QP refinement	42.64	16	173210	-0.19	0.37%
News	JM 9.3 FQP	42.95	20	106360	—	—
	JM 9.3 RC	43.07	16	106420	0.12	0.06%
	PRC w/o QP refinement	43.24	16	103370	0.29	-2.81%
	RC with QP refinement	43.4	16	104560	0.45	-1.69%
Silent	JM 9.3 FQP	42.2	20	105620	—	—
	JM 9.3 RC	43.01	16	105730	0.81	0.10%
	PRC w/o QP refinement	43.07	16	103490	0.87	-2.02%
	RC with QP refinement	43.24	16	104210	1.04	-1.33%
Mother_daughter	JM 9.3 FQP	43.4	20	74480	—	—
	JM 9.3 RC	43.15	16	74600	-0.25	0.16%
	PRC w/o QP refinement	43.33	16	73150	-0.07	-1.79%
	RC with QP refinement	43.58	16	74030	0.18	-0.60%
Salesman	JM 9.3 FQP	42.06	20	80570	—	—
	JM 9.3 RC	42.58	16	80880	0.52	0.38%
	PRC w/o QP refinement	42.86	16	78720	0.8	-2.30%
	RC with QP refinement	43.02	16	79660	0.96	-1.13%
Foreman	JM 9.3 FQP	42.05	20	176710	—	—
	JM 9.3 RC	41.81	16	176730	-0.24	0.01%
	PRC w/o QP refinement	41.46	16	171370	-0.59	-3.02%
	RC with QP refinement	41.7	16	173450	-0.35	-1.84%
Paris	JM 9.3 FQP	41.78	20	743400	—	—
	JM 9.3 RC	41.88	16	743610	0.1	0.03%
	PRC w/o QP refinement	41.66	16	732880	-0.12	-1.42%
	RC with QP refinement	41.87	16	735820	0.09	-1.02%
Stefan	JM 9.3 FQP	41.59	20	2224480	—	—
	JM 9.3 RC	41.1	16	2223810	-0.49	-0.03%
	PRC w/o QP refinement	40.67	16	2192330	-0.92	-1.45%
	RC with QP refinement	40.96	16	2204640	-0.63	-0.89%
City	JM 9.3 FQP	41.43	20	3668610	—	—
	JM 9.3 RC	41.2	16	3665000	-0.23	-0.10%
	PRC w/o QP refinement	40.96	16	3599590	-0.47	-1.88%
	RC with QP refinement	41.28	16	3642920	-0.15	-0.70%

also improved. In our future work, we may try to use Inter 8×8 mode for preencoding to obtain more accurate source information for the sequences.

5. CONCLUSION

We have presented a novel RDO-based rate control algorithm for H.264. The major difficulties in H.264 rate control

have been addressed. The pre-analysis stage is used to break the chicken-and-egg dilemma. Robust header bits prediction model and coefficient bits prediction model are established by adaptively updating the model parameters. The frame-layer bit allocation is simple and effective. By using the two-step QP determination but single-pass encoding scheme at the macroblock-layer rate control, each macroblock's QP is further refined and thus highly conformed to its actual

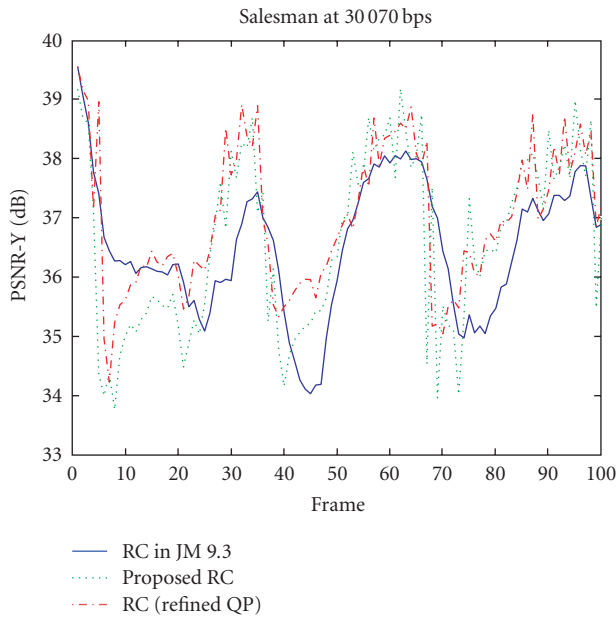


FIGURE 3: PSNR comparison frame-by-frame of “Salesman,” number of target bits = 30 070 bps (RDO on).

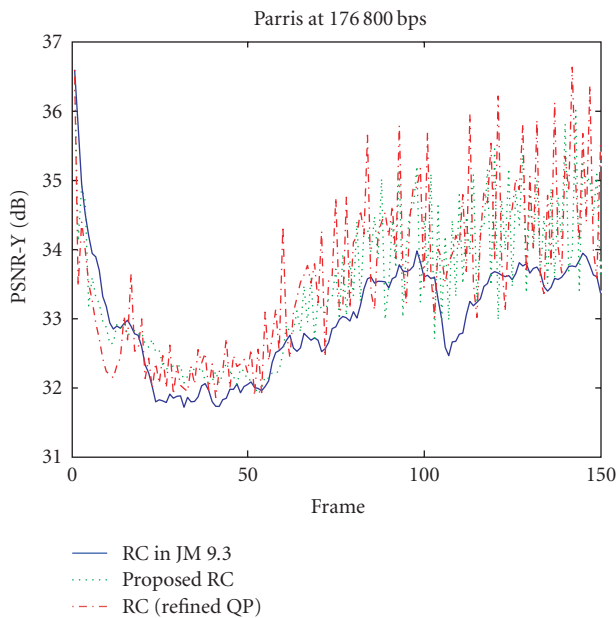


FIGURE 4: PSNR comparison frame-by-frame of “Paris,” number of target bits = 176 800 bps (RDO on).

needs. As shown by the test results, our proposed rate control scheme significantly outperforms the original JM 9.3 with fixed QP and the existing rate control scheme in JM 9.3 in terms of PSNR improvement, while maintaining the bit accuracy.

ACKNOWLEDGMENTS

This work was supported by National Natural Science Foundation of China under Grants no. 60332030 and no. 60502034, and Shanghai Rising-Star Program under Grant no. 05QMX1435.

REFERENCES

- [1] ISO-IEC/JTC1/SC29/WG11, *Information technology—coding of audio-visual objects—part 10: advanced video coding* Final Draft International Standard, ISO/IEC FDIS 14 496-10, December 2003.
- [2] T. Wiegand, “Draft ITU-T recommendation and final draft international standard of joint video specification (ITU-T Rec. H.264 — ISO/IEC 14496-10 AVC),” in *Joint Video Team (JVT) of ISO/IEC MPEG and ITU-T VCEG, VT-G050*, Pattaya, Thailand, March 2003.
- [3] T. Sikora, “Trends and perspectives in image and video coding,” *Proceedings of the IEEE*, vol. 93, no. 1, pp. 6–17, 2005.
- [4] T. Wiegand, G. J. Sullivan, G. Bjontegaard, and A. Luthra, “Overview of the H.264/AVC video coding standard,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 13, no. 7, pp. 560–576, 2003.
- [5] G. J. Sullivan and T. Wiegand, “Video compression—from concepts to the H.264/AVC standard,” *Proceedings of the IEEE*, vol. 93, no. 1, pp. 18–31, 2005.
- [6] ISO-IEC/JTC1/SC29/WG11, “Generic coding of moving pictures and associated audio information: video,” ISO/IEC 13818-2, November 1994.
- [7] ITU-T Study Group 15, “Draft of recommendation H.263: video coding for low bitrate communication,” Tech. Rep., ITU-T, Geneva, Switzerland, May 1996.
- [8] P. H. Hsu and K. J. R. Liu, “A predictive H.263 bitrate control scheme based on scene information,” in *Proceedings of the IEEE International Conference on Multimedia & Expo (ICME '00)*, pp. 1735–1738, New York, NY, USA, July–August 2000.
- [9] S. W. Ma, W. Gao, P. Gao, and Y. Lu, “Rate control for advance video coding (AVC) standard,” in *Proceedings of the IEEE International Symposium on Circuits and Systems (ISCAS '03)*, vol. 2, pp. 892–895, Bangkok, Thailand, May 2003.
- [10] S. W. Ma, W. Gao, F. Wu, and Y. Lu, “Rate control for JVT video coding scheme with HRD considerations,” in *Proceedings of the IEEE International Conference on Image Processing (ICIP '03)*, vol. 3, pp. 793–796, Barcelona, Spain, September 2003.
- [11] P. Li, X. K. Yang, and W. S. Lin, “Buffer-constrained R-D model-based rate control for H.264/AVC,” in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP '05)*, vol. 2, pp. 321–324, Philadelphia, Pa, USA, March 2005.
- [12] J. F. Xu and Y. He, “A novel rate control for H.264,” in *Proceedings of the IEEE International Symposium on Circuits and Systems (ISCAS '04)*, vol. 3, pp. 809–812, Vancouver, BC, Canada, May 2004.
- [13] T. Wiegand, H. Schwarz, A. Joch, F. Kossentini, and G. Sullivan, “Rate-constrained coder control and comparison of video coding standards,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 13, no. 7, pp. 688–703, 2003.
- [14] Z. G. Li, F. Pan, K. P. Lim, et al., “Adaptive frame layer rate control for H.264,” in *Proceedings of the IEEE International Conference on Multimedia & Expo (ICME '03)*, vol. 1, pp. 581–584, Baltimore, Md, USA, July 2003.

- [15] H.264/AVC reference software JM 9.3, <http://ftp3.itu.int/av-arch/jvt-site>.
- [16] M. Jiang and N. Ling, "On enhancing H.264/AVC video rate control by PSNR-based frame complexity estimation," *IEEE Transactions on Consumer Electronics*, vol. 51, no. 1, pp. 281–286, 2005.
- [17] M. Jiang, X. Yi, and N. Ling, "Frame layer bit allocation scheme for constant quality video," in *Proceedings of the IEEE International Conference on Multimedia & Expo (ICME '04)*, vol. 2, pp. 1055–1058, Taipei, Taiwan, June 2004.
- [18] M. Jiang and N. Ling, "Low-delay rate control for real-time H.264/AVC video coding," *IEEE Transactions on Multimedia*, vol. 8, no. 3, pp. 467–477, 2006.
- [19] M. Jiang and N. Ling, "On lagrange multiplier and quantizer adjustment for H.264 frame-layer video rate control," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 16, no. 5, pp. 663–669, 2006.
- [20] Z. G. Li, W. Gao, F. Pan, et al., "Adaptive rate control with HRD consideration," in *Joint Video Team of ISO/IEC and ITU, JVT-H014, 8th Meeting*, pp. 23–27, Geneva, Switzerland, May 2003.
- [21] J. Ribas-Corbera and S. Lei, "Rate control in DCT video coding for low-delay communications," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 9, no. 1, pp. 172–185, 1999.
- [22] S. W. Ma, W. Gao, Y. Lu, and H. Lu, "Proposed draft description of rate control on JVT standard," in *Joint Video Team (JVT) of ISO/IEC MPEG & ITU-T VCEG, JVT-F086, 6th Meeting*, Awaji, Japan, December 2002.
- [23] A. M. Tourapis, K. Sühring, and G. Sullivan, "H.264/MPEG-4 AVC reference software enhancements," in *Joint Video Team (JVT) of ISO/IEC MPEG & ITU-T VCEG. (ISO/IEC JTC1/SC29/WG11 and ITU-T SG16 Q.6), JVT-N008, 14th Meeting*, Hong Kong, China, January 2005.

Xiaokang Yang received the B.S. degree from Xiamen University, Xiamen, China, in 1994, the M.S. degree from Chinese Academy of Sciences, Shanghai, China, in 1997, and the Ph.D. degree from Shanghai Jiao Tong University, Shanghai, China, in 2000. From September 2000 to March 2002, he worked as a Research Fellow in Centre for Signal Processing, Nanyang Technological University, Singapore. From April 2002 to October 2004, he was a Research Scientist in the Institute for Infocomm Research (I²R), Singapore. He is currently an Associate Professor and the Director Assistant of the Institute of Image Communication and Information Processing, Department of Electronic Engineering, Shanghai Jiao Tong University, Shanghai, China. He has published over 70 refereed papers, and has filed 6 patents. His current research interests include networked multimedia processing, media retrieval, perceptual visual processing, digital television, and pattern recognition. He received the Microsoft Young Professorship Award 2006, the Best Young Investigator Paper Award at IS&T/SPIE International Conference on Video Communication and Image Processing (VCIP2003), and awards from A-STAR and Tan Kah Kee foundations. He is currently a Senior Member of IEEE, and a Member of Visual Signal Processing and Communications Technical Committee of the IEEE Circuits and Systems Society. He is the Special Session Chair of Perceptual Visual Processing of IEEE ICME2006. He is currently the Technical Program Cochair for IS-CAS '07 and the Program Cochair for SiPS '07.



Yongmin Tan received the B.S. degree in electronic engineering from Shanghai Jiao-tong University, Shanghai, China, in 2005. He is currently working toward the M.S. degree in the Institute of Image Communication and Information Processing, Department of Electronic Engineering, Shanghai Jiao Tong University, Shanghai, China. His current research interests include scalable video coding, video processing, and rate control.



Nam Ling received the B.Eng. degree (electrical engineering) from the National University of Singapore, and the M.S. and Ph.D. degrees (computer engineering) from the University of Louisiana at Lafayette, USA. He is currently a Full Professor with the Department of Computer Engineering and the Associate Dean (Graduate Studies and Research) for the School of Engineering at Santa Clara University, California, USA. He has more than 120 publications, including a book, in the fields of video coding and systolic arrays. He was named IEEE Distinguished Lecturer for 2002–2003, and received the 2003 IEEE ICCE Best Paper Award (First Place Winner) for his joint work on MPEG-4 face animation. He and his team's proposals on motion estimation and related methods were adopted into the H.264/MPEG-4 AVC video coding international standard. He served as an editor for several journals. He served as the Chair for two IEEE technical committees. He was the General Chair for the IEEE Hot Chips Symposium in 1995. He is currently the Technical Program Cochair for ISCAS '07 and the Program Cochair for SiPS '07. He also served as the Program Chair/Cochair for DCV '02 and SiPS '00.

