Low-Complexity Encoder Framework for Window-Level Rate Control Optimization

Long Xu, Sam Kwong, *Senior Member, IEEE*, Yun Zhang, and Debin Zhao

***Abstract*—There exists a tradeoff between visual quality smoothness and buffer constraint in rate control. Thus, a window model and a window-level rate control algorithm were proposed to handle such a tradeoff recently. However, the extra computational complexity and encoding delay were introduced due to a pre-analysis process in the window-level rate control. In this paper, the window model is first improved for a more convenient usage in practical encodings. Second, a new low-complexity encoder framework is proposed based on an open-loop motion estimation (ME). Third, a new memory-efficient algorithm is proposed to eliminate the drift error of open-loop ME. Fourth, the window-level rate control algorithm is improved to adapt to the proposed encoder framework. The experimental results show that the visual quality smoothness is much improved by the proposed algorithm against the traditional one of H.264/AVC. In addition, the improvement of the window-level rate control is better than the previous one in terms of both bit control accuracy and visual quality smoothness. Besides, the proposed encoder could be used to any other rate control algorithms with pre-analysis to reduce computational complexity.**

***Index Terms*—Rate control, video coding, window model, window-level rate control.**

I. Introduction

RATE CONTROL is very important for a video coding standard to be applied to many industrial applications, such as live TV broadcasting and video streaming over network. Rate control is employed for both delivery of video program over network and storage of video content in hardware medium. It aims to achieve the best perceptual video quality under bandwidth, buffer, total volume, and computational complexity constraints. Rate control of source encoding is usually classified into constant bit rate (CBR) control [1], [2] and variable bit rate (VBR) control [3]–[6]. To maintain a constant average bit rate in a short term, CBR adopts the uniform bit allocation among different coding units irrespective of individual picture's characteristic. Thus, there have to be large quantization differences among the coding units with diverse characteristics, which usually results in the fluctuation of perceptual visual quality. On the contrary, VBR is designed to regulate its output bit rate according to the varying characteristics of video content in order to achieve perceptually consistent visual quality in a long term. In VBR, we need to pre-analyze video sequence to capture video content changes along time. The target bits will be distributed to various scenes, allocating more bits to complex scene and fewer bits to simple scene. Traditionally, VBR is realized by two-pass or multipass encoding processes, which requires more delay and computational complexity. Hence, it would need more efforts than CBR when applying to real-time encoding.

In H.264/AVC, a rate control scheme [1] with quadratic rate-distortion (R-D) model and interframe mean absolute difference (MAD) prediction is employed. For quantization parameter (QP) decision of H.264/AVC, there is an inherent dilemma when rate control and rate distortion optimization (RDO) [7], [8] are both enabled. The MAD of each basic coding unit which might be a macroblock (MB), frame or slice, should be provided before QP decision. However, the problem is that the MAD is not available until the coding unit was encoded. Therefore, a predicted MAD is used instead of the actual MAD in rate controls [1], [2] of H.264/AVC. When scene changes or high motions happen, the predicted MAD will be much different from the actual one. When it is used to calculate QP, large QP variation and significant visual quality fluctuation would be caused. To handle this problem, some researchers improved the existing rate control of H.264/AVC by optimizing MAD prediction in [9], [10]. In [11], the authors employed a low-pass filter to smooth frame distortion for MPEG-4 video encoder. In [12], a sequence-based bit allocation model was proposed to track nonstationary characteristics of input video. In [13], the picture quality variation and buffer fluctuation were formulated into an optimization problem to get as smooth as possible picture quality while maintaining buffer smoothness. In [14], the authors proposed a two-stage QP decision for better visual quality and bit control accuracy.

The traditional rate control of H.264/AVC [1], [2] cannot obtain the information of future frames for bit allocation. It predicted the characteristics of future frames from previously coded frames with the assumption of the stationarity of input signal, which is usually not true to the nature video signals. In case the input signal is not stationary, the rate control

Manuscript received November 21, 2011; revised February 12, 2012; accepted February 22, 2012. Date of publication March 14, 2012; date of current version January 30, 2013. This work was supported by the Hong Kong Research Grants Council General Research Fund, under Project 9041495 (CityU 115109), National Nature Science Foundation of China under Grants 61102088, and the Major State Basic Research Development Program of China (973 Program) under Grant 2009CB320905.

L. Xu was with the Department of Computer Science, City University of Hong Kong, Kowloon, Hong Kong. He is currently with the Automation Department, University of Science and Technology Beijing, Beijing, 100083, China (e-mail: lxu@ustb.edu.cn).

S. Kwong is with the Department of Computer Science, City University of Hong Kong, Kowloon, Hong Kong (e-mail: cssamk@cityu.edu.hk).

Y. Zhang is with the Shenzhen Institutes of Advanced Technology, CAS, Shenzhen, 518005, China (e-mail: yunzhang@cityu.edu.hk).

D. Zhao is with the Department of Computer Science, Harbin Institute of Technology, Harbin 150001, China (e-mail: dbzhao@hit.edu.cn).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TIE.2012.2190960

may fail. In addition, the R-D efficiency and visual quality would become worse. Thus, the pre-analysis tools [15]–[17] were widely used to handle the nonstationarity of input signal, where both the encoding efficiency and rate control accuracy were much improved. In [15], only 16×16 intermode was performed to get the mapping between the number of zero coefficients and QP in pre-analysis. In [16], the variance of each MB was computed from the temporally predicted residue using only 16×16 intermode in pre-analysis. In [17], a pre-analysis which was similar to that of [15] was used to obtain ρ-QP tables. Thus, the extra computational complexity and encoding delay were introduced by the pre-analysis. In this paper, a new encoder framework is proposed to handle these problems, where the pre-analysis is integrated into motion estimation (ME) of a standard encoder. Hence, the extra computational complexity and encoding delay were completely removed from the previous window-level rate control algorithm. In addition, a memory-efficient residue correction method is proposed to free the proposed encoder framework from drift error. Then, a new window-level rate control is proposed based on the new encoder framework.

The rest of this paper is organized as follows. In Section II, the window model is improved for the sake of convenient usage of it in practical encoding task. In Section III, a new encoder framework and an improved window-level rate control algorithm are presented in detail. Section IV shows the experimental results and analyses. A conclusion is given in the last section.

II. Improved Window Model

In practice, there is a tradeoff between visual quality smoothness and buffer constraint in rate control. Even though the accurate characteristics of videos can be obtained by the aforementioned pre-analysis, without enough buffer resources, the visual quality smoothness still would be compromised. The tradeoff was firstly studied in [17] by a so-called window model. In [17], the bounds on frame QP variation (ΔQ) and frame bits variation (ΔR) were used to represent visual quality smoothness and buffer constraint. Using window model, ΔR and ΔQ given by the application are converted into window size (L) which is controllable during encoding process. In [17], the frame QP variation was supposed to be a Gaussian random distribution, represented by $\xi(\omega)(\mu_\xi = 0, \sigma_\xi^2)$. According to Central Limit Theorem [18], the average of $\{\xi_k(\omega)\}(k = 1, 2, \ldots, n)$ converges to a Gaussian random variable. From the Law of Large Numbers [18]

$$P\left(\left|\frac{1}{n}\sum_{k=1}^{n}\xi_k - \mu_\xi\right| < \Delta Q\right) = \frac{1}{\sqrt{2\pi}}\int_{-\Delta Q(\sqrt{n}/\sigma_\xi)}^{\Delta Q(\sqrt{n}/\sigma_\xi)} e^{-\frac{t^2}{2}}\,dt \quad (1)$$

was derived in [17]. In (1), ΔQ represents the given bound of QP variation. Let (1) equal to p, the integral limit of (1) can be calculated by looking up the standardized normal table. Assume $\Delta Q(\sqrt{n}/\sigma_\xi) = \sqrt{\alpha}$, replacing n by L, then $L - \Delta Q$ model [17] was derived as

$$L = \alpha\frac{\sigma_\xi^2}{\Delta Q^2}. \quad (2)$$

According to R-D model, the frame bits variation aroused by ΔQ was calculated from

$$\delta R = -\frac{\beta}{Q}\Delta Q. \quad (3)$$

Based on (2) and (3), a window model [17] was proposed as

$$L = \alpha\frac{\sigma_\eta^2}{\Delta Q^2}\min\left\{1, \left(\frac{\left(\frac{\beta}{Q}\right)}{\frac{\Delta Q}{\Delta R}}\right)^2\right\} \quad (4)$$

where α and β were the model parameters. For a certain application, ΔR and ΔQ were given as the encoding requirements. Thus, L could be calculated from (4).

The model efficiency of (4) was proved there by a proposed window-level rate control algorithm. However, a practical encoding task usually assumes the buffer constraint instead of bit rate fluctuation ΔR. A conversion from the given buffer constraint to bit rate fluctuation was needed. In this paper, the window model is improved by using the buffer constraint directly replacing of bit rate fluctuation. In our improved window model, only $L - \Delta Q$ model (5) is employed. However, ΔR is replaced by buffer size Z. The improved window model is more convenient than [17] since a practical application usually assumes a buffer constraint Z. In [11], the authors proved that the encoder buffer of the proposed rate control was subject to

$$|Z_e(n)| \le \frac{1}{2}\Theta\sum_{j=1}^{L} a_j \times j \quad (5)$$

where $Z_e(n)$ represented the buffer occupancy for the nth frame, Θ was such a constant that $|\log_2\sigma^2(j)| \le \Theta$ was true for all frames, where $\sigma^2(j)$ was the variance of residue of the jth frame after intra- or interprediction. $\{\alpha_j; \alpha_j > 0, \sum_{j=1}^{L}\alpha_j = 1\}$ was an L-tap averaging filter for smoothing the distortion of frame encodings. Let $W_{\max} = \Theta\sum_{j=1}^{L} a_j \times j$, (5) can be rewritten as $|Z_e(n)| \le (1/2)W_{\max}$. It was regarded as the upper bound of encoder buffer that was needed for video streaming usage of the rate control of [18]. Particularly, if an arithmetic averaging filter was employed in (5), then

$$W_{\max} = \frac{L+1}{2}\Theta \quad (6)$$

which suggested that the larger the filter length (L) and variation of scene activity (Θ), were the larger encoder buffer was needed for the video streaming.

The function of filter length L mentioned above is the same as that of the window size of (4). In this paper, the window size L is firstly derived from $L - \Delta Q$ model of (2). Then $W_{\max}$ can be deduced from (6). If $W_{\max} \le Z$, L from (2) will be the final window size which conforms to both ΔQ and Z; otherwise, L from (2) is further decreased. Substituting Z for $W_{\max}$ in (6), and combining with (2), the proposed window model is deduced as

$$L = \begin{cases} \alpha\frac{\sigma_\xi^2}{\Delta Q^2}; & \text{if } W_{\max} \le Z \\ \beta \times \frac{Z}{\Theta} - 1; & \text{else} \end{cases} \quad (7)$$

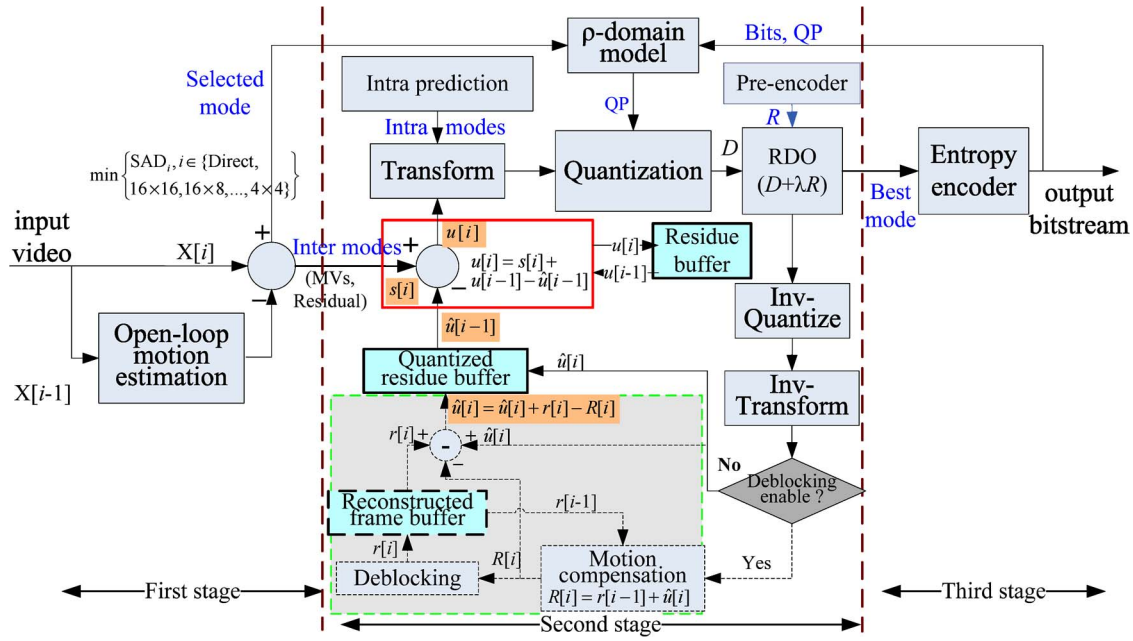

Fig. 1. Proposed encoder framework with open-loop ME.

where the model parameters α and β are manually initialized and adaptively updated for each window. σ_ξ^2 is the variance of the assumed Gaussian distribution of QP variations. From (7), the given ΔQ and Z decide L. In turn, L makes the quality and bits variations of actual encoding satisfy the given ΔQ and Z. In addition, we also update window size adaptively by statistics of bits and QP usages after coding a window.

III. Window-Level Rate Control Scheme for Open-Loop ME Based Encoder

The delivery of video content over industrial networks is usually under the given buffer constraint, bit rate constraint [19], [20]. The window model is a theoretical model mapping the visual quality smoothness and buffer constraint to window size which can be dynamically regulated during the encoding process. It enables us to determine the best efficiency subject to the given requirements in a theoretical way. To achieve such efficiency, the corresponding rate control algorithm based on window model should be provided. In [17], a window-level ρ-domain rate control was proposed to be an example of usage of window model. In the ρ-domain model [15], the percentage of zero coefficients quantized (ρ) is directly related to the bit rate (R) as

$$R = \theta(1-\rho). \tag{8}$$

It can be established immediately after a pre-analysis with only 16 × 16 mode ME and DCT transform. The rate control algorithm of [17] could achieve smooth visual quality as much as possible under a given buffer constraint. However, additional computational complexity was introduced due to the pre-analysis although the increase of complexity was limited. In addition, there was a start-up delay in algorithm [17] due to pre-analysis of a window. Furthermore, only 16 × 16 interprediction mode is used in pre-analysis in order to reduce computational cost. Thus, the mismatch between 16 × 16 mode of pre-analysis and actual best mode may occur frequently, which eventually reduces the efficiency of the proposed algorithm in both bit control accuracy and R-D performance.

In this section, a new proposed encoder framework and the corresponding window-level rate control on the new encoder framework are presented in detail. These two proposals are expected to efficiently handle the flaws of [17]. In addition, the encoder framework can be used to any other rate controls with pre-analysis process.

A. Proposed Encoder Framework

Traditionally, a hybrid encoder framework, such as a standard H.264/AVC encoder, has a close-loop process which consists of ME, transform, quantization, dequantization and inverse transform. In such a framework, the ME is performed on the reconstructed frames obtained from the close-loop, so it is called close-loop ME. Open-loop ME means that ME is performed by reference to original input frames instead of the reconstructed frames. However, the original input frames are not available in decoder side in compression, so a drift error is caused. Open-loop ME was usually used in hardware accelerator and parallel algorithms of an encoder, because it provided a hardware-friendly partitioning of an encoder. From our analytical derivations below, the drift error of open-loop ME can be easily removed by rewriting residue before entropy encoder. In addition, the memory cost of the proposed drift error-free algorithm is very low.

The proposed encoder framework is shown in Fig. 1. From Fig. 1, an encoder is divided into three stages or modules, i.e., open-loop ME, mode decision and entropy encoder. The first module of open-loop ME performs ME and motion compensation at original input frames, and produces residue and

motion vectors (MVs) for each prediction mode. The second module selects a mode with minimum R-D cost among all mode candidates. Then, the third module, i.e., entropy encoder, writes the best mode into bitstream. It should be noted that the residual signal from the first module is not the real residue written into bitstream. It will be rewritten in the second module. Thus, the final output bitstream absolutely conforms to the standard.

B. Proposed Drift Error-Free Algorithm

Assume that $X[i]$ represents the ith original input frame, $s[i]$ is the residue between $X[i]$ and $X[i-1]$, which is derived from the first module, $u[i]$ is the real residue between $X[i]$ and its reconstructed reference $r[i-1]$, $\hat{u}[i]$ is the quantized $u[i]$, we have

$$\begin{cases} s[i] = X[i] - X[i-1] \\ u[i] = X[i] - r[i-1] \end{cases}. \tag{9}$$

According to the relation between reconstructed frame $r[i-1]$ and its reference frame $r[i]$, we have

$$r[i] = r[i-1] + \hat{u}[i]. \tag{10}$$

Based on (9) and (10), the residue $s[i]$ from the first module is rewritten as

$$\begin{aligned} u[i] &= X[i] - r[i-1] \\ &= (X[i] - X[i-1]) + (X[i-1] - r[i-1]) \\ &= s[i] + [X[i-1] - (r[i-2] + \hat{u}[i-1])] \\ &= s[i] + u[i-1] - \hat{u}[i-1]. \end{aligned} \tag{11}$$

From (9), the output bitstream of entropy encoding of $s[i]$ cannot be decoded properly because the original frame $X[i-1]$ is not available in decoder. However, the entropy encoding of $u[i]$ from (11) is absolutely complied with the standard. From (11), $u[i]$ is modified from $s[i]$ without the needs of $X[i]$ and $r[i-1]$, while the conventional close-loop ME based encoder gets $u[i]$ by using $u[i] = X[i] - r[i-1]$ with the needs of both $X[i]$ and $r[i-1]$. The operation of (11) is highlighted in a red box of Fig. 1 without the needs original input signal $X[i]$ but only residual signal $s[i]$. Therefore, a small bandwidth of data flow is between the first module and the second one, which is an appealing attribute in the hardware accelerator of an encoder. From (11), there only need to store one previous residue $u[i-1]$ and its reconstruction $\hat{u}[i-1]$ for the residue correction of current frame, so very little memory cost is required for both software and hardware. In addition, the encoding procedure accesses $X[i]$, $u[i]$, and $\hat{u}[i]$ only one time, thereby leading to an efficient off-chip memory access in hardware.

The best advantage of the proposed encoder framework is to increase the parallel processing capability from one MB to one frame. Thus, more data could be processed in a clock cycle in parallelism. The encoder framework and the corresponding rate control algorithm can be easily adapted to industrial products of real-time video encoders, such as live TV broadcasting, online video streaming, particularly high-definition video encoders. In addition, since ME is time consuming, it is often accelerated by the hardware realization in industrial usage [21], [22]. While other modules of an encoder, such as RDO, rate control, and entropy encoder which are usually reconfigured for different applications, are implemented in software. Thus, the proposed encoder framework consisting of three stages could satisfy the purpose of software-hardware partitioning of an encoder.

TABLE I
PROPOSED WINDOW-LEVEL RATE CONTROL ALGORITHM BASED ON THE PROPOSED ENCODER FRAMEWORK

Symbol	Meaning
L	*Window size*
r	*Bit rate*
f	*Frame rate*
B	*Bits quota of a window*
B_j	*Remaining bits before encoding the j-th frame*
b_j	*Generated bits for coding the j-th frame*
θ	*Model parameters of ρ-domain R-D model (8)*
ρ	*The number/percentage of zero coefficients*
α, β	*The parameters of window model (7)*
$\sigma^2(j)$	*The variance of residual signal u[j] from (9) for the j-th frame*
Θ	*The maximum variance of residual signals*

TABLE II
SYMBOLS USED IN THE PROPOSED RATE CONTROL ALGORITHM

Step 1: If the first window, L is initialized to be the Group of Picture (GOP) length; otherwise, compute window size L from (8);

Step 2: Perform open-loop ME for the current window, record MVs and residual signal $s[j]$ (j=0,1,…,L-1) for each candidate mode of each MB;

Step 3: Perform residual signal correction according to (11), compute $\sigma^2(j)$ for each residual frame $u[j]$ (j=0,1,…,L-1);

Step 4: Select the best mode with minimum SAD to be the input of rate control, build ρ-QP tables $\rho_j(QP)$ (j=0,1,…,L-1) for the frames of current window, and the sum of them is the window-level ρ-QP table $\rho_w(QP)$ referring to (12);

Step 5: Compute the total bits of the current window $B = (r/f) \times L$;

Step 6: B_j (j=0,1,…,L-1) is initialized to B, i.e., $B_0 = B$ before encoding of a window;

Step 7: Compute QP of the j-th frame according to (8):
Firstly compute ρ by using ρ-domain model (8): $B_j = \theta_j(1-\rho_j)$; secondly, get QP of the j-th frame by looking up ρ-QP table $\rho_w(QP)$ in (12);

Step 8: Perform encoding for the j-th frame using the obtained QP above;

Step 9: Update remaining bits of current window as $B_{j+1} = B_j - b_j$, update ρ-QP tables by subtracting the ρ-QP table of the j-th frame $\rho_j(QP)$ from $\rho_w(QP)$;

Step 10: Update θ as $\theta_{j+1} = b_j/(1-\rho_j)$ with the actual coding bits b_j. If the last frame of current window is reached, go to Step 11; else go to Step 6;

Step 11: Update window model parameters α and β as:

$$\alpha = L \times \Delta Q^2 \Big/ \sigma_\xi^2 \text{ , where } \sigma_\xi^2 = \sum_{j=0}^{L-1} (Q_j - \frac{1}{L}\sum_{j=0}^{L-1} Q_j)^2 ;$$

$$\beta = \frac{(L+1) \times \Theta}{Z},$$

where $\Theta = \max\{\sigma^2(j), j = 0,1,...,L-1\}$;

Step 12: If the sequence ends, terminates the procedure; else go to Step 1.

TABLE III
COMPARISONS OF TIME COMPLEXITY OF SOFTWARE IMPLEMENTATION

Sequence	Bit rate (kbps)	Traditional rate control		ρ-domain		Kamaci's algorithm		Close-loop		Open-loop	
		Total time (s)	ME time (s)	Total time (s)	ME time (s)	Total time (s)	ME time (s)	Total time (s)	ME time (s)	Total time (s)	ME time (s)
Foreman	2000	343.05	174.83	433.14	239.13	352.97	179.98	359.38	178.76	342.48	176.12
	1000	326.35	176.05	414.16	238.91	338.04	177.60	342.86	180.85	326.22	178.18
	500	315.04	178.22	401.02	240.98	315.25	180.03	331.65	184.27	315.13	181.55
	300	308.67	179.86	393.56	243.75	318.38	182.75	325.53	188.12	308.95	185.34
News	2000	339.46	160.86	424.93	223.60	349.53	168.89	349.08	163.62	333.16	161.20
	1000	321.32	163.44	408.12	228.31	325.79	167.70	331.95	166.15	316.30	163.69
	500	307.21	165.88	394.89	231.06	308.06	177.29	319.99	172.24	304.31	169.69
	300	298.83	170.07	385.75	233.00	303.63	174.76	311.53	172.94	296.02	170.38
Silent	2000	353.37	159.23	445.26	224.28	357.30	167.45	368.27	163.23	351.97	160.82
	1000	330.48	162.46	417.71	223.12	335.20	166.61	343.17	165.56	327.27	163.11
	500	311.73	164.48	396.48	228.35	317.20	166.64	322.84	168.90	307.16	166.40
	300	301.25	166.59	385.48	230.31	304.38	177.63	312.67	173.48	297.19	170.92
Average		**320.48**		**407.4**		**327.15**		**334.91**		**318.84**	
				27.12%		***2.08%***		***4.24%***		***-0.51%***	
Harbour	10000	6121.71	3177.29	7782.23	4357.28	6131.03	3193.13	6397.19	3184.48	6131.42	3158.39
	8000	6032.14	3182.59	7668.48	4361.85	6038.00	3200.60	6303.59	3186.80	6025.25	3179.18
	5000	5899.95	3201.70	7502.18	4389.10	5914.67	3219.19	6165.45	3207.19	5901.37	3198.24
	2000	5699.60	3254.21	7246.36	4460.78	5719.35	3270.11	5956.08	3256.69	5706.23	3251.02
Crew	10000	6043.30	3403.08	7685.63	4666.70	6051.09	3407.23	6315.25	3411.91	6054.21	3401.27
	8000	5974.16	3401.95	7593.23	4664.06	5992.75	3406.62	6243.00	3409.12	5968.25	3398.25
	5000	5863.48	3415.31	7453.82	4680.43	5866.99	3419.67	6127.34	3418.22	5869.28	3420.01
	2000	5715.95	3459.02	7268.93	4742.10	5719.69	3463.41	5973.17	3465.69	5718.26	3456.89
Night	10000	4585.32	2425.37	5831.15	3324.64	4594.62	2435.64	4791.66	2434.86	4583.68	2415.38
	8000	4526.60	2428.92	5754.69	3330.17	4528.80	2430.64	4730.30	2437.69	4513.29	2438.26
	5000	4426.89	2430.70	5629.29	3332.72	4445.30	2439.18	4626.10	2432.22	4438.64	2429.56
	2000	4284.63	2468.19	5444.78	3382.06	4292.03	2481.72	4477.44	2473.27	4294.01	2455.28
Average		**5431.14**		**6905.06**		**5505.25**		**5675.55**		**5433.66**	
				27.14%		***1.36%***		***4.50%***		***0.46%***	

C. Proposed Window-Level Rate Control Algorithm

As the conventional usage of ρ-domain algorithm, the relationship between the number of zero coefficients ρ and QP should be recorded for each MB beforehand. Usually, a pre-analysis process is used to get ρ and QP relations. However, there is no need to do such a process for building ρ and QP relation in the proposed encoder framework. Thus, the extra computational complexity of ρ-domain model is completely removed from the proposed algorithm. From Fig. 1, the first module of open-loop ME produces residues of each mode for each MB. We select the best one with the minimum sum of absolute difference (SAD) to be the input of rate control. The frame-level 1-QP table is $\rho_j(QP) = \sum_{m=0}^{m=N-1} \rho_m^{MB}(QP)$, where $\rho_m^{MB}(QP)$ represents ρ-QP table of the mth MB. Assume the ρ-QP tables of all frames in a window $\{\rho_j(QP), j = 0, 1, \ldots L-1\}$, the window-level ρ-QP table is obtained by the sum of all frame's ρ-QP tables, i.e.,

$$\rho_w(QP) = \sum_{j=0}^{j=L-1} \rho_j(QP). \tag{12}$$

With the pre-built ρ-QP tables in (12), the original window-level rate control algorithm [17] is improved based on the proposed new encoder framework in Table I. The symbols used in the proposed rate control algorithm are summarized in Table II.

IV. EXPERIMENTAL RESULTS AND DISCUSSIONS

The proposed window-level rate control algorithm is realized on JM14.0 of H.264/AVC under conditions: *Profile/Level: 100/40, Reference frames: 2, Full ME*, Search range: *16, RDO on* and *CABAC, IPPP* encoding structure. For exhibiting the advantages of the proposed algorithm, the traditional rate control algorithm [1], [2], ρ-domain algorithm [15], Kamaci's algorithm [23], and the window-level algorithm [17] are benchmarks. We name our proposed rate control algorithm "open-loop" because of open-loop ME. The original one [17] is called "close-loop" accordingly in the following content.

First, the extra computational complexity of pre-analysis is completely removed from the original window-level rate control [17] by using the proposed encoder framework in Fig. 1, which enables the usage of the window-level rate control in real-time video encoding. The implementing time of software indicates that 0.51% time saving averagely of open-loop algorithm less than traditional algorithm [1] at frame level for CIF sequences as Table III shown. The ρ-domain algorithm [15] is 27% more complex than traditional one because it is with many times operations of looking-up tables, updating parameters at MB-level leaving aside pre-analysis. Kamaci's algorithm is comparable to the traditional one. The MB level of [1] is over 15% more complex than it at frame level.

Second, the comparisons of R-D performance and bit control error are performed among the traditional rate control [1], [2], ρ-domain [15], Kamaci's algorithm [23], close-loop [17], and

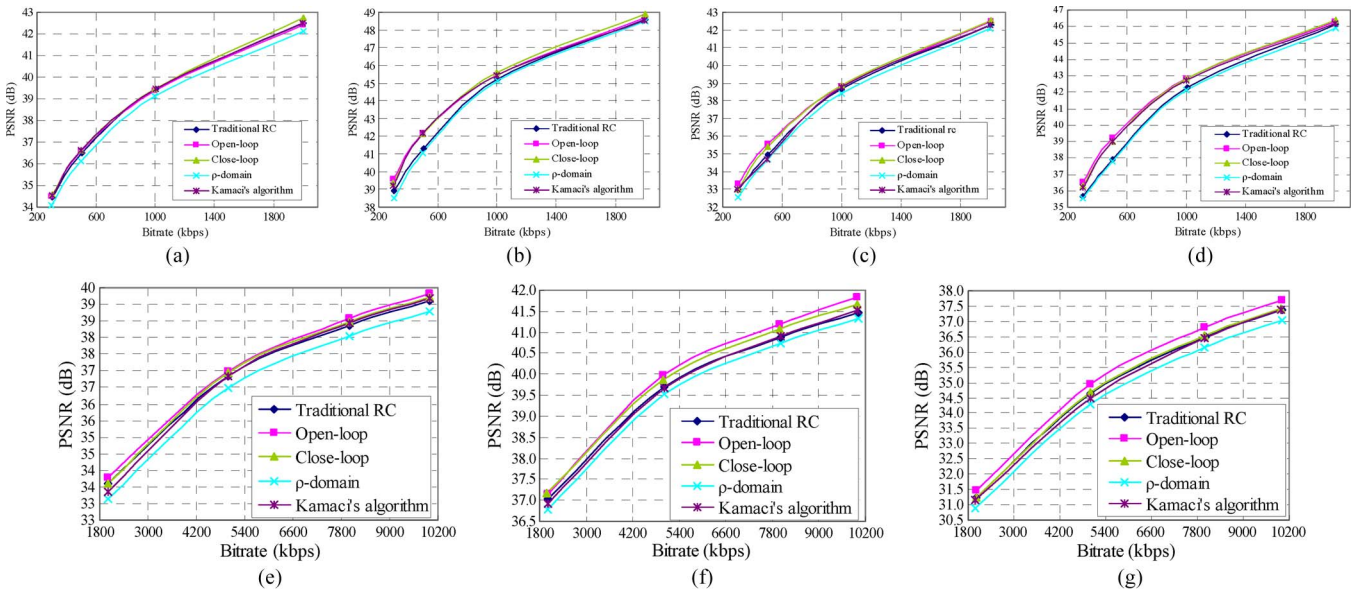

Fig. 2. Comparison of R-D performance between the proposed algorithm and benchmarks [(a)–(d) are CIF sequences, (e)–(g) are 720 P (1280 × 720) HD sequences]. (a) "Foreman"; (b) "News"; (c) "Tennis"; (d) "Silent"; (e) "Night"; (f) "Crew"; (g) "Harbour".

TABLE IV
PERFORMANCE COMPARISON FOR ALL TESTING ALGORITHMS IN TERMS OF BIT CONTROL ERROR AND PSNR GAIN

Resolution	Sequence	Target bit rate (kbps)	Traditional rate control			ρ-domain			Kamaci's algorithm			Close -loop			Open -loop		
			Bit rate (kbps)	PSNR (dB)	Bit control error (%)	Bit rate (kbps)	PSNR (dB)	Bit control error (%)	Bit rate (kbps)	PSNR (dB)	Bit control error (%)	Bit rate (kbps)	PSNR (dB)	Bit control error (%)	Bit rate (kbps)	PSNR (dB)	Bit control error (%)
CIF	Foreman	2000	2001.77	42.49	0.09	2000.83	42.14	0.04	1998.99	42.48	-0.05	1999.94	42.77	0.00	2002.42	42.43	0.12
		1000	1003.43	39.45	0.34	998.83	39.12	-0.02	1001.38	39.46	0.24	999.84	39.45	-0.02	1000.90	39.41	0.09
		500	501.77	36.48	0.35	497.82	36.13	-0.44	501.00	36.62	0.20	500.26	36.62	0.05	500.42	36.64	0.08
		300	303.08	34.46	1.03	299.53	34.08	-0.16	300.90	34.55	0.30	301.44	34.60	0.48	300.35	34.58	0.12
	News	2000	2001.70	48.57	0.09	2010.90	48.20	0.55	1999.43	48.55	-0.03	1999.81	48.91	-0.01	1998.88	48.70	-0.06
		1000	1001.16	45.24	0.12	1000.83	44.93	0.18	999.67	45.45	0.07	1000.10	45.57	0.01	999.73	45.49	-0.03
		500	501.66	41.29	0.33	502.76	41.29	0.55	500.12	42.16	0.02	500.29	42.11	0.06	499.78	42.22	-0.04
		300	301.11	38.92	0.37	301.90	38.67	0.63	300.06	39.21	0.02	300.30	39.38	0.10	300.07	39.61	0.02
	Tennis	2000	2000.24	42.31	0.01	2005.12	41.98	0.26	1999.69	42.31	-0.02	2001.19	42.52	0.06	2000.26	42.49	0.01
		1000	1000.65	38.65	0.06	999.20	38.27	-0.08	998.65	38.75	-0.14	1000.60	38.89	0.06	998.90	38.84	-0.11
		500	500.16	34.95	0.03	499.99	34.61	0.00	499.08	34.72	-0.18	500.79	35.40	0.16	501.43	35.57	0.29
		300	300.57	32.98	0.19	298.42	32.52	-0.53	299.71	33.01	-0.10	301.76	33.00	0.59	300.91	33.31	0.30
	Silent	2000	2001.88	46.15	0.09	1994.40	45.76	-0.28	2000.13	46.19	0.01	2000.38	46.42	0.02	1999.25	46.29	-0.04
		1000	1002.51	42.31	0.25	997.83	41.86	-0.12	1000.50	42.73	0.15	1000.29	42.82	0.03	1000.07	42.83	0.01
		500	502.52	37.92	0.50	498.77	37.55	-0.25	501.13	38.99	0.23	499.92	38.98	-0.02	500.13	39.26	0.03
		300	301.58	35.65	0.53	300.07	35.22	0.02	300.93	36.19	0.31	299.96	36.29	-0.01	300.31	36.51	0.10
Average				**39.86**	**0.27**		**39.52**	**0.02**		**40.09**	**0.06**		**40.23**	**0.1**		**40.26**	**0.06**
							-0.34			***0.23***			***0.37***			***0.40***	
720P	Night	10000	10001.67	39.10	0.02	10009.68	38.80	0.10	9999.40	39.18	-0.01	9999.87	39.22	0.00	9992.79	39.33	-0.07
		8000	8000.36	38.36	0.00	8000.17	38.05	0.00	7995.29	38.42	-0.06	7999.89	38.47	0.00	8002.39	38.56	0.03
		5000	5000.95	36.83	0.02	4992.78	36.47	-0.14	4998.84	36.83	-0.02	4998.10	36.94	-0.04	5000.83	36.99	0.02
		2000	2003.66	33.60	0.18	1994.03	33.15	-0.30	1999.36	33.37	-0.03	2000.95	33.61	0.05	2001.01	33.79	0.05
	Crew	10000	10007.22	41.47	0.07	10013.80	41.34	0.14	9994.74	41.51	-0.05	9995.13	41.65	-0.05	9992.94	41.82	-0.07
		8000	8006.89	40.88	0.09	8011.27	40.74	0.14	8013.48	40.91	0.17	7998.49	41.06	-0.02	7994.15	41.20	-0.07
		5000	5006.23	39.70	0.12	5007.53	39.52	0.15	5007.03	39.67	0.14	4994.18	39.87	-0.12	4996.95	39.99	-0.06
		2000	2002.82	37.04	0.14	2003.08	36.78	0.15	2002.71	36.92	0.14	1996.85	37.17	-0.16	1997.91	37.15	-0.10
	Harbour	10000	10003.49	37.37	0.03	10007.44	37.05	0.07	9991.02	37.37	-0.09	9997.35	37.41	-0.03	10001.49	37.70	0.01
		8000	8001.01	36.49	0.01	7997.51	36.13	-0.03	7995.46	36.45	-0.06	7997.60	36.52	-0.03	7998.95	36.80	-0.01
		5000	5000.80	34.65	0.02	5000.40	34.28	0.01	4998.73	34.50	-0.03	5001.53	34.68	0.03	5000.18	34.94	0.00
		2000	2000.74	31.24	0.04	1996.48	30.90	-0.18	1999.84	31.15	-0.01	2000.59	31.25	0.03	2000.58	31.52	0.03
Average				**37.23**	**0.06**		**36.93**	**0.01**		**37.19**	**0.01**		**37.32**	**-0.03**		**37.48**	**-0.02**
							-0.30			***-0.04***			***0.09***			***0.25***	

open-loop. The ρ-domain algorithm is inferior to others in terms of R-D efficiency. The proposed algorithm achieves an obvious PSNR improvement over the others. It is 0.40 dB and 0.25 dB better than the traditional one on CIF and HD sequences, respectively. In addition, open-loop is a little better than close-loop at low bit rate with regard to R-D performance. The PSNR comparisons at each given bit rate are shown in Fig. 2. Table IV summarizes the experimental results, where open-loop achieves

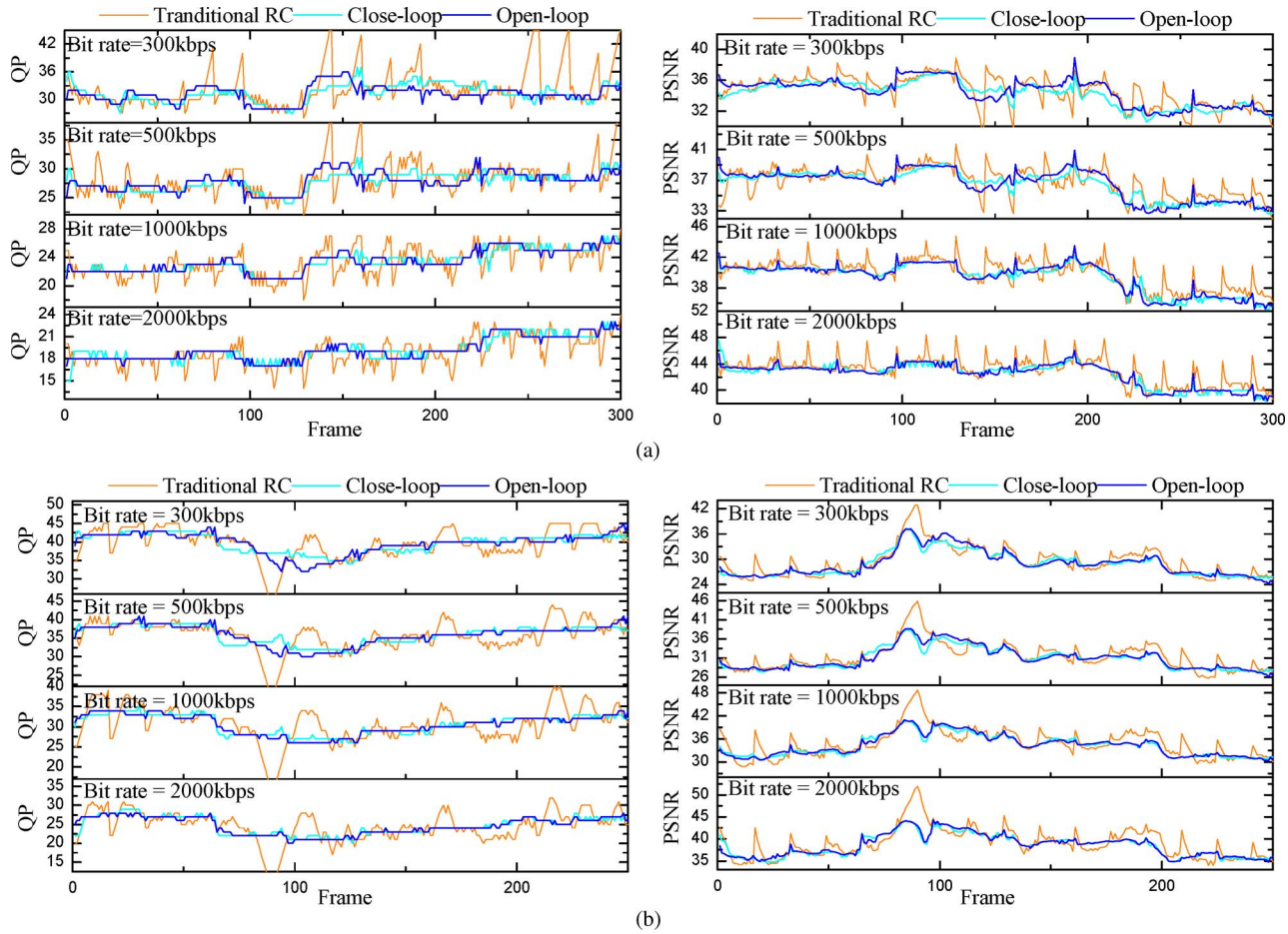

Fig. 3. Comparisons of frame QP/PSNR curves among traditional and two window-level rate control algorithms (smooth QP/PSNR indicates smooth/consistent visual quality at the situations of single and similar scenes). (a) "Foreman"; (b) "Football."

an average of 0.03 dB and 0.16 dB PSNR gain on CIF and HD sequences, respectively, over close-loop. Open-loop at low bit rate is better than it at high bit rate with respect to R-D performance. Comparing with close-loop, open-loop achieves 0.23 dB, 0.31 dB, and 0.22 dB PSNR gains for "News," "Tennis," and "Silent," respectively at 300 kbps. However, there is little PSNR loss at high bit rate for open-loop. At low bit rate, open-loop ME can obtain more consistent and reasonable MVs than close-loop ME, because the motion alignments in open-loop ME are mostly along the real trajectories of the objects' movements in nature video sequence. Thus, the information of residue, such as ρ-QP table in this work, is more accurate and stable for open-loop method. Based on a better ρ-QP table, the better bit control accuracy, visual quality smoothness, and R-D efficiency can be therefore achieved. While for close-loop ME, the serious visual quality degradation of references would much affect the accuracy and efficiency of ME when bit rate is very low. Open-loop and close-loop are comparable with respect to average R-D performance. As far as the bit control error concern, the proposed open-loop scheme is the best among all testing algorithms with bit control error below 0.06% and 0.03% for CIF and HD sequences, respectively.

Third, both open-loop and close-loop can achieve significantly smooth picture quality compared to the benchmarks [1], [2], [15], [23]. In Fig. 3, the frame QP/PSNR curves of traditional rate control, close-loop and open-loop are compared. The significantly smooth picture quality can be observed for open-loop and close-loop from Fig. 3. The QP/PSNR curves of close-loop and open-loop are more similar at medium and high bit rates by observing Fig. 3, which can be explained to some extent by the related ρ-QP characteristics as shown in Fig. 4. From Fig. 4, there exist high correlations between the black, blue, and green curves which represent ρ-QP relations of open-loop, close-loop with QP of 16 and 28, respectively, i.e., they seem to be parallel in the figure. At low bit rate encoding, open-loop is a little better than close-loop with respect of R-D performance. The reason lies in that the MVs of open-loop are much more consistent than those of close-loop at such a situation, which could promise a better encoding achievement.

Fourth, only 16×16 mode of pre-analysis may not be consistent of actual modes of final encoding. Thus, to select a mode by SAD from open-loop ME to build ρ-QP tables is better than that of close-loop using pre-analysis with only 16×16 blocks. From our experiments, more than 50% MBs select non-16×16 block from high to medium bit rate, which can be proved by the block partitions of decoded frames shown in Figs. 5 and 6. Even at low bit rate, the percentage of non-16×16 blocks is still significant. The percentage of MBs selecting 16×16 bock is listed in Table V, where the same conclusion as Figs. 5 and 6 can be obtained. Moreover, to use the proposed encoder framework may result in a fast mode decision by making decisions among several candidate modes

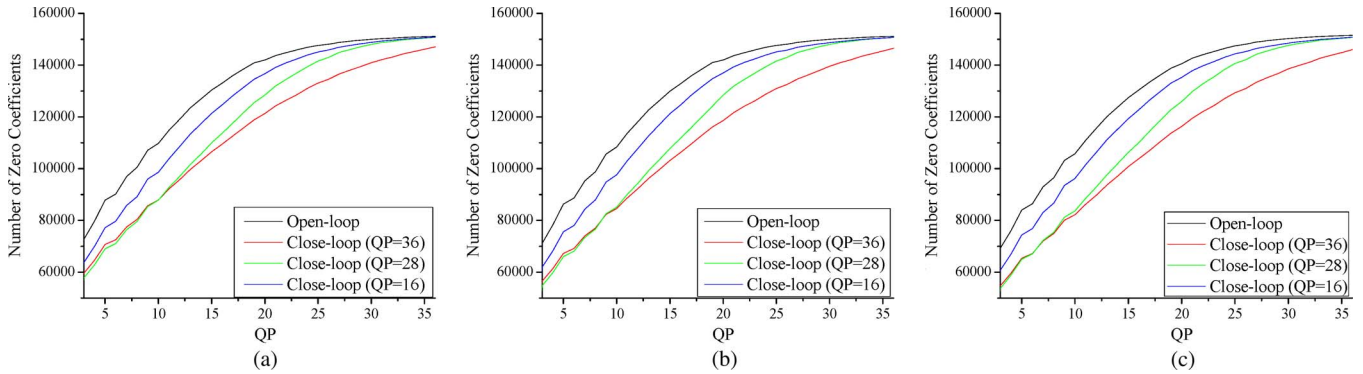

Fig. 4. ρ-QP relations of "Foreman" [(a), (b) and (c) are for the 10, 20, and 30th frame, respectively; encodings of QP 16, 28 and 36 indicate high, media and low bit rate encodings, respectively, corresponding PSNRs of them are about 45, 36, and 31 dB, respectively].

Fig. 5. Block partition of the 10th, 30th, 50th, and 60th frames of "Foreman" (CIF) [A bigger square grid represents a MB, a small one is the partitions of a MB; (a)–(d) is coded at 2 Mbps and (e)–(h) is at 500 kbps].

with smaller SAD instead of all modes. Before RDO, SAD of each mode has been available due to the mode selection for building ρ-QP tables of rate control.

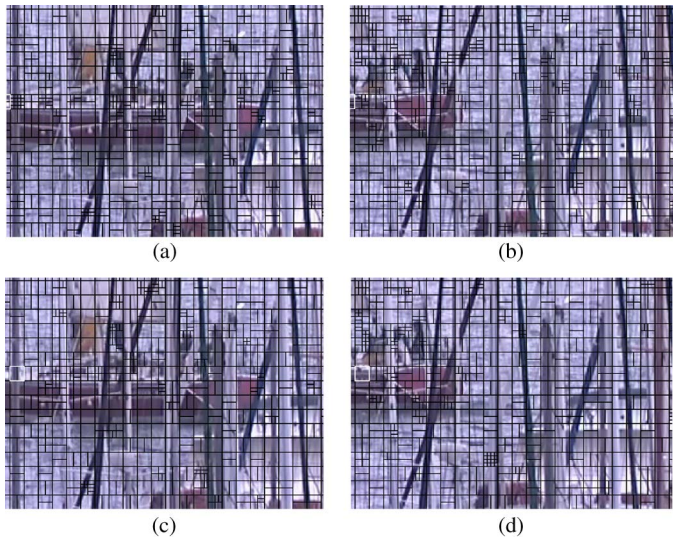

Fig. 6. Block partition of the 10th and 100th frames of "Harbour" (720 P) [A bigger square grid represents a MB, a small one is the partitions of a MB; (a)–(b) is coded at 10 Mbps and (c)–(d) is at 5 Mbps;].

TABLE V
PERCENTAGE (%) OF MBS WITH 16 × 16 BLOCK ("< 16 × 16" INDICATES 16 × 16 MODE AND SKIP/DIRECT MODE)

Sequence	2Mbps		500kbps	
	16×16 mode	<16×16 mode	16×16 mode	<16×16 mode
Foreman	25.09	32.61	31.19	45.84
Football	31.76	31.76	22.88	42.15
	10Mbps		5Mbps	
Harbour	42.26	50.84	43.02	54.92
Night	31.55	53.08	28.00	65.08

V. CONCLUSION

In this paper, the window model is firstly improved for a more convenient usage in rate control. Secondly, a low-complexity encoder framework is proposed to reduce the extra computational complexity of pre-analysis in rate control. The encoder framework is based on open-loop ME. The drift error of open-loop ME is well handled by introducing a new residue correction method. Thirdly, the original window-level rate control algorithm is improved based on the proposed new encoder framework. The proposed rate control can afford smooth picture quality as much as possible under given buffer

constraint. In addition, it is without extra computational complexity compared to the traditional rate control of H.264/AVC. Furthermore, the proposed encoder framework can be used to reduce the overhead complexity of other rate control algorithms using pre-analysis.

REFERENCES

[1] S. W. Ma, Z. G. Li, and F. Wu, "Proposed draft adaptive rate control," in *Proc. 8th Meeting Joint Video Team (JVT) of ISO/IEC MPEG & ITU-T VCEG, Doc. JVT-H017r3*, Geneva, Switzerland, May 20–26, 2003.

[2] Y. Liu and Z. G. Li, "A novel rate control scheme for low delay video communication of H.264/AVC standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 17, no. 1, pp. 68–78, Jan. 2007.

[3] M. R. Pickering and J. F. Arnold, "A perceptually efficient VBR rate control algorithm," *IEEE Trans. Image Process.*, vol. 3, no. 5, pp. 527–532, Sept. 1994.

[4] P. H. Westerink, R. Rajagopalan, and C. A. Gonzales, "Two-pass MPEG-2 variable-bit-rate encoding," *IBM J. Res. Develop.*, vol. 43, no. 4, pp. 471–488, Jul. 1999.

[5] B. Han and B. F. Zhou, "VBR rate control for perceptual consistent video quality," *IEEE Trans. Consum. Electron.*, vol. 54, no. 4, pp. 1912–1919, Nov. 2008.

[6] Y. Yu, J. Zhou, Y. Wang, and C. W. Chen, "A novel two pass VBR coding algorithm for fixed-size storage application," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 11, no. 3, pp. 345–356, Mar. 2001.

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

[8] T. Wiegand and B. Girod, "Lagrange multiplier selection in hybrid video coder control," in *Proc. Int. Conf. Image Process.*, Thessaloniki, Greece, Oct. 2001, pp. 542–545.

[9] J. P. Dong and N. Ling, "A model parameter and MAD prediction scheme for H.264 macroblock layer rate control," in *Proc. ISCAS*, May 18–21, 2008, pp. 628–631.

[10] S. J. Wu, Y. Q. Huang, and T. Ikenaga, "A macroblock-level rate control algorithm for H.264/AVC video coding with context-adaptive MAD prediction model," in *Proc. ICCMS*, Feb. 20–22, 2009, pp. 124–128.

[11] Z. H. He, W. Zeng, and C. W. Chen, "Low-pass filtering of rate-distortion functions for quality smoothing in real-time video communication," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 15, no. 8, pp. 973–981, Aug. 2005.

[12] B. Xie and W. Zeng, "A sequence-based rate control framework for consistent quality real-time video," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 16, no. 1, pp. 56–71, Jan. 2006.

[13] Z. Z. Chen and K. N. Ngan, "Towards rate-distortion tradeoff in real-time color video coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 17, no. 2, pp. 158–167, Jan. 2007.

[14] S. D. Hu, H. L. Wang, S. Kwong, and C.-C. Jay Kuo, "Novel rate-quantization model-based rate control with adaptive initialization for spatial scalable video coding," *IEEE Trans. Ind. Electron.*, vol. 59, no. 3, pp. 1673–1684, Mar. 2011.

[15] Z. H. He, Y. K. Kim, and S. K. Mitra, "Low-delay rate control for DCT video coding via ρ-domain source modeling," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 11, no. 8, pp. 928–940, Aug. 2001.

[16] J. R. Corbera and S. M. Lei, "Rate control in DCT video coding for low-delay communications," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 9, no. 1, pp. 172–185, Feb. 1999.

[17] L. Xu, D. B. Zhao, X. Y. Ji, S. Kwong, and W. Gao, "Window-level rate control for smooth picture quality and smooth buffer occupancy," *IEEE Trans. Image Process.*, vol. 20, no. 3, pp. 723–734, Mar. 2011.

[18] F. Cucker and D. X. Zhou, *Learning Theory: An Approximation Theory Viewpoint*. Cambridge, U.K.: Cambridge Univ. Press, 2007.

[19] P. C. Ting and M. C. Yuang, "Dynamic multi-threshold rate control mechanisms for supporting ABR traffic in ATM networks," *IEEE Trans. Ind. Electron.*, vol. 45, no. 1, pp. 60–68, Feb. 1998.

[20] J. S. Blanes, L. Almerdia, R. Marau, and P. Pedreiras, "On-line QoS management for multimedia real-time transmission in industrial network," *IEEE Trans. Ind. Electron.*, vol. 58, no. 3, pp. 1061–1071, Mar. 2011.

[21] M.-H. Lee, K.-K. Shyu, P.-L. Lee, C.-M. Huang, and Y.-J. Chiu, "Hardware implementation of EMD using DSP and FPGA for on-line signal processing," *IEEE Trans. Ind. Electron.*, vol. 58, no. 6, pp. 2473–2481, Jun. 2010.

[22] S. Jung and S. S. Kim, "Hardware implementation of a real-time neural network controller with a DSP and an FPGA for nonlinear systems," *IEEE Trans. Ind. Electron.*, vol. 54, no. 1, pp. 265–271, Feb. 2007.

[23] N. Kamaci, Y. Altunbasak, and R. M. Mersereau, "Frame bit allocation for the H.264/AVC video coder via Cauchy-density-based rate and distortion models," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 15, no. 8, pp. 994–1006, Aug. 2005.

Long Xu received the M.S. degree in applied mathematics from Xidian University, Xi'an, China, in 2002, and the Ph.D. degree from the Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China.

He was a Postdoc Researcher at the Department of Computer Science of City University of Hong Kong, Kowloon, Hong Kong from August 2009 to July 2011. And now, he is with the Automation Department, University of Science and Technology Beijing, Beijing, China. His research interests include image/video coding, wavelet-based image/video coding, and computer vision.

Sam Kwong (M'93–SM'04) received the B.S. and M.S. degrees in electrical engineering from the State University of New York at Buffalo, Buffalo, NY, in 1983, The University of Waterloo, Waterloo, ON, Canada, in 1985, and the Ph.D. degree from the University of Hagen, Germany, in 1996.

From 1985 to 1987, he was a Diagnostic Engineer with Control Data Canada. He joined Bell Northern Research Canada as a Member of Scientific Staff. In 1990, he became a Lecturer in the Department of Electronic Engineering, City University of Hong Kong, Kowloon, Hong Kong, where he is currently a Professor at the Department of Computer Science. His research interests are video and image coding and evolutionary algorithms.

Yun Zhang received the B.S. and M.S. degrees in electrical engineering from Ningbo University, Ningbo, China, in 2004 and 2007, respectively, and the Ph.D. degree in computer science from Institute of Computing Technology, Chinese Academy of Sciences (CAS), Beijing, China, in 2010.

From 2009 to 2011, he was a Postdoc Researcher with the Department of Computer Science, City University of Hong Kong, Kowloon, Hong Kong. In 2010, he joined the Shenzhen Institutes of Advanced Technology, CAS, as an Assistant Professor. His research interests are multiview video coding, video object segmentation, and content-based video processing.

Debin Zhao received the B.S., M.S., and the Ph.D. degrees in computer science from the Harbin Institute of Technology, Harbin, China, in 1985, 1988, and 1998, respectively.

He was a Research Fellow in the Department of Computer Science, City University of Hong Kong, Kowloon, Hong Kong, from 1989 to 1993. And now, he is a Professor in the Department of Computer Science of the Harbin Institute of Technology. His research interests include data compression, image processing, and human-machine interface. He has published two books and over 60 scientific papers.