A Universal Rate Control Scheme for Video Transcoding

Long Xu, Sam Kwong, *Senior Member, IEEE,* Hanli Wang, *Member, IEEE,* Yun Zhang, Debin Zhao, and Wen Gao, *Fellow, IEEE*

***Abstract*—Video transcoding is proposed for the bitrate adaption, spatial and/or temporal resolutions adaption, and video format conversion. In video streaming application, it converts videos at server to the compatible versions demanded by networks or clients' devices, so that the videos can be delivered over networks and displayed in the clients' devices successfully. This paper provides a universal rate control scheme for various video transcoding purposes. First, a new rate-distortion (R-D) model is established theoretically for better representing the real R-D feature of transcoding. Second, a window-level rate control algorithm is proposed for providing smooth visual quality with compliant buffer constraint by utilizing the two-pass R-D model and a new proposed sliding window buffer control strategy. Finally, a universal rate control scheme for transcoding is developed based on the established R-D model and the proposed window-level rate control algorithm. The extensive experimental results demonstrate that as compared to other state-of-the-art rate control algorithms for transcoding, the proposed scheme can achieve more bit control accuracy with the average mismatch below 0.2%, and much more consistent visual quality with 0.1 dB–0.3 dB peak-to-signal noise ratio improvement in average, while with low computational complexity.**

***Index Terms*—Bitrate adaption, rate control, video transcoding, window-level rate control.**

I. Introduction

THE AIM OF transcoding is to convert an already compressed bitstream to another version in order to meet the requirements of network transmissions and/or the needs of users. It is often performed when the target device has to convert incompatible or obsolete data to its own supported or latest formats. On the one hand, there are still many MPEG-2 video sources recorded in the past decades. As H.264/AVC achieves better compression performance over all previous video coding standards, the demand of transcoding to H.264/AVC from other video formats is increasing. On the other hand, video streams over heterogeneous networks become more popular. It requires that a video source should have a great diversity of bitstreams for the specific network transmission need. However, only a few bitstreams are recorded for a video source due to the limited storage space. The delivery of video sources over networks is usually with different characteristics, and it should be able to adapt to the specific network conditions. In addition, the diverse terminals usually have different decoding/display capabilities, so the transcodings of bitrate, spatial and/or temporal resolutions are also needed. Furthermore, the users may access video sources via different physical medium such as broadband channels, phone cables, or mobile wireless channels. It is necessary to adapt the video bitstreams to different transmission bandwidth.

Generally, there were three classical structures of transcoding: cascade, open-loop, and close-loop structures in the literature [1]. The cascade transcoder included a standard decoder followed by a standard encoder. It was most efficient among all these transcoders from the aspect of rate-distortion (R-D) performance. However, its complexity was significantly higher than a standard encoder. Open-loop transcoder [2], [3] was directly implemented on discrete cosine transform (DCT) coefficients of encoded bitstreams, and it was with the lowest computational complexity. However, severe drifting error [1] made it infeasible to many applications. Close-loop transcoder [4]–[6] introduced a decoding loop into the open-loop structure after the quantization of transcoding. The output of the decoding loop was used to reconstruct reference frames for transcoding the subsequent frames. Thus, the problem of drifting error propagation has been addressed in the close-loop structure. However, the coding efficiency of close-loop transcoder was low due to the reuse of modes and motion vectors (MVs) of encoded bitstreams. Many efforts on designing new transcoders have been made to achieve a good tradeoff between R-D performance and computational complexity in recent years. A frequency-domain close-loop transcoder was presented in [5]. A hybrid open-loop and close-loop transcoder

Manuscript received February 1, 2011; revised June 12, 2011; accepted August 16, 2011. Date of publication September 15, 2011; date of current version April 2, 2012. This work was supported in part by the Hong Kong Research Grants Council General Research Fund, under Project 9041495 (CityU 115109), in part by the Major State Basic Research Development Program of China (973 Program), under Grant 2009CB320905, in part by the National Natural Science Foundation of China, under Grant 60736043, in part by the Program for New Century Excellent Talents in University of China, under Grant NCET-10-0634, and in part by the Shanghai Pujiang Program, under Grant 11PJ1409400. This paper was recommended by Associate Editor E. Steinbach.

L. Xu and S. Kwong are with the Department of Computer Science, City University of Hong Kong, Kowloon, Hong Kong (e-mail: lxu@jdl.ac.cn; cssamk@cityu.edu.hk).

H. Wang is with the Department of Computer Science and Technology and Key Laboratory of Embedded System and Service Computing, Ministry of Education, Tongji University, Shanghai 200092, China (e-mail: hanliwang@tongji.edu.cn).

Y. Zhang is with the Shenzhen Institute 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).

W. Gao is with the School of Electronics Engineering and Computer Science, Peking University, Beijing 100871, China (e-mail: wgao@pku.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/TCSVT.2011.2168150

was given in [6]. In [7], an efficient H.264/AVC block partitioning prediction algorithm for MPEG-2 to H.264/AVC transcoding was proposed to speedup transcoding process. In [8], transcoding was used to intelligently derive scalable bitstreams from the existing single-layer bitstream of H.264/AVC. In [9], the transcoding task was distributed among peers and completed collaboratively for peer-to-peer Internet protocol television networks. In [10], a machine learning tool was used to exploit the correlation of mode selection between MPEG-2 and H.264/AVC.

Transcoding plays a key role in many video streaming applications, such as in video-on-demand (VOD). High-quality video stream is recorded and stored at the server, and it needs to be adapted to the output bitrates for different users with different channel bandwidths or terminal devices. It is the general requirement to ease the consumption of multimedia content for universal multimedia access (UMA). Rate control is a key technology to meet this huge demand for UMA. It is desired to achieve both of the accurate target bitrate and the consistent/smooth visual quality at the same time.

In the past decade, a number of transcoding rate control algorithms were proposed to maintain high visual quality with low computational complexity. In [11], a novel R-D model was analytically formulated for bitrate adaption transcoding, and a new rate control algorithm was proposed accordingly. Pantoja *et al.* [12], [13] proposed two rate control algorithms for transcoding VC-1 to H.264/AVC. In [14], Xiu *et al.* proposed a rate control algorithm that considered proportional integral derivative controller to achieve smooth visual quality for transcoding. In [15], an efficient rate control for MPEG-2 to H.264/AVC transcoding was developed by introducing quantization parameter (QP) interpolation. In [16], Wang *et al.* proposed a frame layer adaptive rate control algorithm suitable for the video bitrate transcoding from MPEG-2 to H.264/AVC. In [17], a rate control algorithm for real-time transcoding from MPEG-2 to H.264/AVC was proposed, where a minimum QP was considered for achieving the maximum usage of available bit resources. However, most of the aforementioned algorithms did not fully exploit the information from the encoded bitstream, and the smooth visual quality of transcoding is not guaranteed. In addition, most transcoding-based applications require low computational complexity and low buffer storage. At such a situation, the open-loop or close-loop structure instead of cascade structure is usually preferred for transcoding. Thus, most of the state-of-the-art rate control algorithms could not be used in transcoding for real-time video streaming. For example, in JVT-H017r3 [18], [19], the mean absolute difference (MAD) required for rate control is not yet available during open-loop or close-loop transcoding because there are no reconstructed video frames. In this paper, a novel R-D model is first established theoretically for better approximating the practical R-D feature of transcoding. Second, a window-level rate control algorithm which operates at the window basic unit is proposed for the consistent visual quality and compliant buffer constraint. The window basic unit consists of several frames or group of pictures (GoPs), which is processed together in the rate control. The consistent visual quality is achieved by utilizing the information of QP and bits consumption of the encoded bitstream in the manner of the classical two-pass model [20] at window basic unit. The sliding window buffer control strategy [21] is applied to address the buffer constraint problem. Finally, based on the established R-D model and the proposed window-level rate control algorithm, a universal rate control scheme is developed for various transcodings. Since only the information of QP and bits consumption of the encoded bitstream is required by the proposed scheme and it can be easily obtained by parsing the encoded bitstream for transcoding, the proposed scheme could be universally used to any transcoding purpose and with low computational complexity.

The remainder of this paper is organized as follows. Section II develops a novel R-D model for transcoding. A window-level rate control algorithm is described in Section III. The proposed universal rate control scheme for transcoding is detailed in Section IV. The experimental results are presented in Section V, and some concluding remarks are given in the last section.

II. R-D Model for Transcoding

In the existing transcoding rate control algorithms, the R-D models are the same as those for general encoding of raw video data, and the statistical information of raw video data is used for rate control during transcoding. However, transcoding is performed on the decoded/degraded data instead of raw video data. Thus, the statistical information might not be accurate enough for transcoding. There usually exists significant bit control error in transcoding rate control if the conventional R-D models are used. Since the transcoding process is performed on the encoded bitstream generated from the encoding of raw video data, which indicates that the raw video data is actually quantized twice. Thus, in the proposed R-D model, we also include the quantization procedure of encoding into the R-D model for transcoding, which is not done in the existing transcoding R-D models to the best of our knowledge. Therefore, both the quantizations in encoding and transcoding are considered in the proposed R-D models.

To investigate the quantization effect of encoding on transcoding, a set of experiments of encoding are first implemented on a raw video sequence *Foreman* (CIF) using constant QPs ranging from 16 to 44 $\{Q_1 = 16, 18, 20, 22, 24, 28, 36, 44\}$. Eight bitrate values $\{R_1(Q_1)\}$ and eight distortion values $\{D_1(Q_1)\}$ are recorded for raw video encoding respectively. Then, transcoding is further performed on the encoded bitstreams using $Q_2\{Q_2 = 20, 24, 28, 36\}$. Eight bitrate values $\{R_2(Q_1, Q_2)\}$ and eight distortion values $\{D_2(Q_1, Q_2)\}$ are recorded for each Q_2 respectively. The performance of an H.264/AVC cascade transcoder for bitrate adaption compared to that of a standard H.264/AVC encoder is drawn in Fig. 1.

A. Rate Model

In Fig. 1(a), $\{R_2(Q_1, Q_2)\}$ is drawn along $\{R_1(Q_1)\}$ for each Q_2, where the horizontal axis is the bitrate of raw video encoding, represented by $R_1(Q_1)$, and the vertical axis is the bitrate of transcoding, represented by $R_2(Q_1, Q_2)$. From Fig. 1(a), $R_2(Q_1, Q_2)$ remains almost the same when

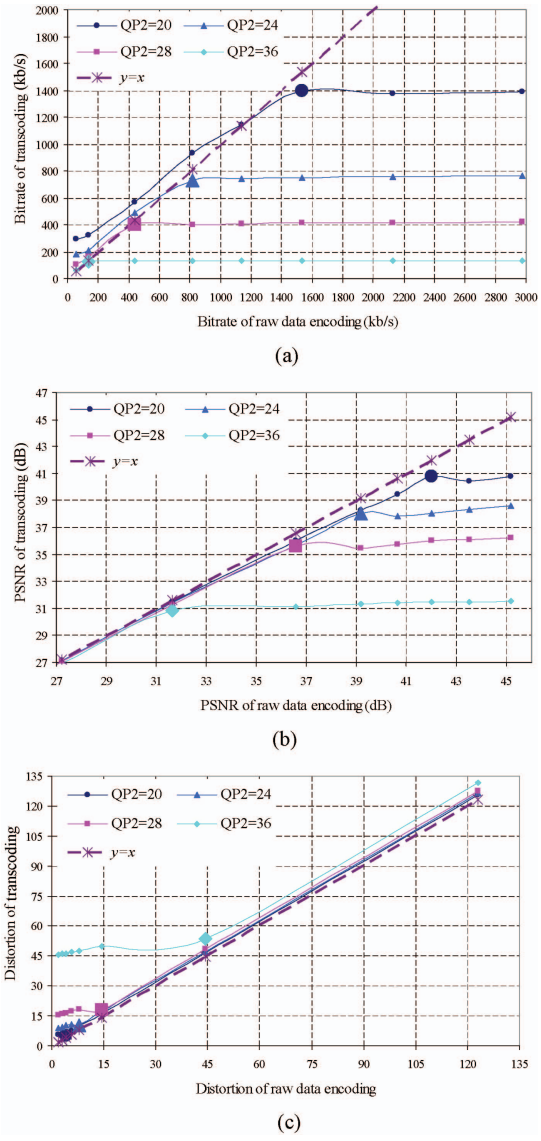

Fig. 1. Bitrate, PSNR, and distortion of transcoding versus those of raw video encoding on *Foreman* (CIF). (a) Bitrate modeling results. (b) PSNR modeling results. (c) Distortion modeling results.

Q_2 is larger than Q_1, and it reduces dramatically when Q_1 is larger than Q_2. As Q_1 becomes larger than Q_2, $R_2(Q_1, Q_2)$ is larger than $R_1(Q_2)$. From the aspect of quantization, $R_2(Q_1, Q_2)$ should equal $R_1(Q_1)$ when Q_2 equals Q_1, which is quantized twice with the same QP. However, $R_2(Q_1, Q_2)$ is a little less than $R_1(Q_2)$ when Q_1 equals Q_2, because macroblocks may select different MVs from the previous ones during transcoding. In Fig. 1(a), the curve of "$y = x$" is plotted with a slope of 1. Thus, $R_2(Q_1, Q_2)$ above "$y = x$" indicates that the bitrate of transcoding is larger than that of raw video data encoding, and vice versa.

Based on the observation of Fig. 1(a), the idealized rate characteristics are drawn in Fig. 2, where the curves of $R_2(Q_1, Q_2)$ are divided by the curve of "$y = sx$" in dashed. Thus, the rate model of transcoding can be approximated as

$$R_2(Q_1, Q_2) = \begin{cases} s \times R_1(Q_2), & Q_2 \geq Q_1 \\ r \times R_1(Q_1) + (s - r) \times R_1(Q_2), & Q_2 < Q_1 \end{cases} \quad (1)$$

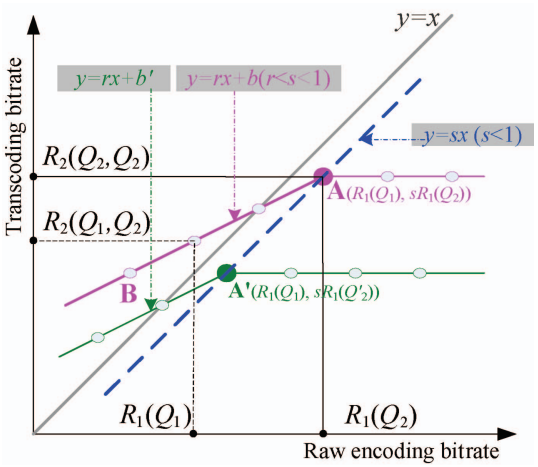

Fig. 2. Proposed bitrate model for transcoding with respect to bitrate/distortion of raw video encoding.

where $R_1(Q_1) = a \times Q_1^{-\alpha}$ is under the assumption of Cauchy distribution of quantized DCT coefficients [26]. The derivation of the bottom formula in (1) is as follows. Given the slope r and the coordinate of $\mathbf{A}(x_A, y_A)$, the straight line across **A** and **B** can be described as

$$y(A, B) = r(x - x_A) + y_A. \quad (2)$$

The coordinate of **A** is $(R_1(Q_2), s \times R_1(Q_2))$ in Fig. 2, so (2) can be also written as

$$\begin{aligned} y(A, B) &= r \times x + (y_A - r x_A) \\ &= r \times x + (s \times R_1(Q_2) - r \times R_1(Q_2)) \\ &= r \times x + (s - r) \times R_1(Q_2). \end{aligned} \quad (3)$$

The parameters s and r are the slopes of the two straight lines in dashed and red color, respectively. The bits cost of encoding should be theoretically determined by the larger one of Q_1 and Q_2 in transcoding. The encoding bits would approach 0 if Q_1 becomes infinite, i.e., "$y = sx$" in Fig. 2 crosses (0, 0), so the slope s can be calculated by $R_2(Q_2, Q_2)/R_1(Q_2)$. Usually, s is less than 1 because $R_2(Q_1, Q_2) < R_1(Q_1)$ when $Q_2 = Q_1$ as stated above. The parameter s is somewhat different for different transcoding, which can be given by offline experiments and updated dynamically during transcoding using statistics of the actual bits usage. In addition, we also notice that r is less than s from Fig. 2. Thus, we can conclude that s and r are subject to $r < s < 1$ in (1) for bitrate adaption of H.264/AVC transcoding. Actually, such a conclusion can be drawn for any other video coding standard with respect to transcoding for bitrate adaption by the same way of the experiments and analysis mentioned previously.

B. Distortion Model

Fig. 1(b) and (c) shows the peek signal-to-noise ratio (PSNR) and distortion (measured by mean squared error) of transcoding with respect to those of encoding. In Fig. 1(c), $\{D_2(Q_1, Q_2)\}$ is drawn along $\{D_1(Q_1)\}$ for each Q_2. From Fig. 1(c), all curves of $\{D_2(Q_1, Q_2)\}$ are above "$y = x$," which indicates that the distortion of transcoding is always larger than

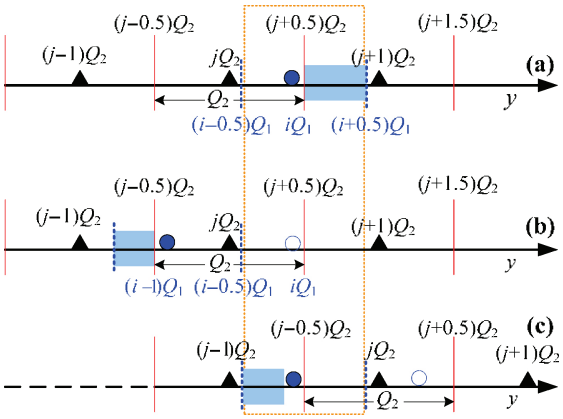

Fig. 3. Difference between transcoding quantization and general one-pass quantization.

that of encoding. This fact can be proved by the following discussions about Fig. 3.

In Fig. 3, the DCT coefficients are arranged along the horizontal axis according to their amplitudes. The shaded DCT coefficients in Fig. 3(a) are in $[(j+0.5)\ Q_2,\ (j+1.5)\ Q_2]$, so they are quantized to $(j+1)\ Q_2$ by one-pass quantization of general encoding. However, they are quantized to jQ_2 during transcoding, because they are first quantized to iQ_1 by Q_1 during encoding, and then are further quantized to jQ_2 ($iQ_1[(j-0.5)Q_2,\ (j+0.5)Q_2]$) by Q_2 during transcoding. The same situation also occurs for Fig. 3(b). It will result in Fig. 3(c) by transiting Fig. 3(b) horizontally. The combination of Fig. 3(a) and (c) would form a new shaded region where the DCT coefficients are quantized to two different values by one-pass quantization and transcoding quantization. Assume that the QP of encoding on raw video source is Q_1 and the QP of transcoding on decompressed video source is Q_2, the distortion of encoding by Q_1 is calculated by

$$D(Q_1) = \sum_{j=-\infty}^{+\infty} \int_{(j-0.5)Q_1}^{(j+0.5)Q_1} (y - jQ_1)^2 p(y)dy \tag{4}$$

where y is the DCT coefficient and $p(y)$ represents the probability distribution function of y. If the quantization with Q_2 is further performed on the results of quantization with Q_1, the distortion from Q_1 followed by Q_2 can be deduced by

$$D_{Q_1}(Q_2) = \sum_{j=-\infty}^{+\infty} \sum_{(j-0.5)Q_2 \le iQ_1}^{iQ_1<(j+0.5)Q_2} \int_{(i-0.5)Q_1}^{(i+0.5)Q_1} (y - jQ_2)^2 p(y)dy. \tag{5}$$

With the help of Fig. 3, the difference between these two quantizations can be derived analytically as (6), shown at the bottom of the page, where $D(Q_2)$ denotes the distortion of one-pass quantization by Q_2 for general encoding on raw video source. It is in the form of (4) by replacing Q_1 with Q_2. By replacing y in $e_1(y)$ and $e_2(y)$ with $(j+0.5)Q_2$ and $(j-0.5)Q_2$, respectively, both $e_1\ (y)$ and $e_2(y)$ can be proved positive by (7), and (8), shown at the bottom of the page. Thus, we can finally conclude that the transcoding distortion is always larger than that of the one-pass quantization on raw video data, i.e., $D_{Q1}(Q_2) - D(Q_2) > 0$.

Based on the observation of Fig. 1(c) and the conclusion that the distortion of transcoding is always larger than that of encoding, the distortion model of transcoding is deduced in a similar manner to the rate model mentioned above. We idealize the characteristics of transcoding distortion with respect to that of raw video data encoding in Fig. 4, where $D_1(Q_1)$ represents the distortion of raw video data encoding with QP of Q_1, and $D_2(Q_1,\ Q_2)$ represents the distortion of transcoding with QP of Q_2 preceded by encoding with QP of Q_1. According to (6)–(8), $D_2(Q_1,\ Q_2)$ is always larger than $D_1(Q_1)$ for bitrate adaption transcoding, so the curve of $D_2(Q_1,\ Q_2)$ is always above that of $D_1(Q_1)$ in Fig. 4. In Fig. 4, $D_2(Q_1,\ Q_2)$ parallels the horizontal axis when Q_1 is less than Q_2 which means transcoding distortion is determined by Q_2 as $Q_1 < Q_2$, and $D_2(Q_1,\ Q_2)$ starts to increase as Q_1 becomes larger than Q_2 which indicates that transcoding distortion is related to Q_1 as $Q_1 > Q_2$. Therefore, the distortion model of transcoding is

$$\begin{aligned} D_{Q_1}(Q_2) - D(Q_2) &= \sum_{j=-\infty}^{+\infty} \sum_{(j-0.5)Q_2 \le iQ_1}^{iQ_1<(j+0.5)Q_2} \int_{(i-0.5)Q_1}^{(i+0.5)Q_1} (y - jQ_2)^2 p(y)dy - \sum_{j=-\infty}^{+\infty} \int_{(j-0.5)Q_2}^{(j+0.5)Q_2} (y - jQ_2)^2 p(y)dy \\ &= \sum_{j=-\infty}^{+\infty} \left\{ \int_{(j+0.5)Q_2}^{(i+0.5)Q_1} \left[(y - jQ_2)^2 - (y - (j+1)Q_2)^2\right] p(y)dy + \int_{(i-0.5)Q_1}^{(j-0.5)Q_2} \left[(y - jQ_2)^2 - (y - (j-1)Q_2)^2\right] p(y)dy \right\} \end{aligned} \tag{6}$$

$$\begin{aligned} &= \sum_{j=-\infty}^{+\infty} \left\{ \int_{(j+0.5)Q_2}^{(i+0.5)Q_1} \left[(2Q_2 y - (2j+1)Q_2^2\right] p(y)dy + \int_{(i-0.5)Q_1}^{(j-0.5)Q_2} \left[(2j-1)Q_2^2 - 2Q_2 y\right] p(y)dy \right\} = e_1(y) + e_2(y) \\ e_1(y) &= \sum_{j=-\infty}^{+\infty} \int_{(j+0.5)Q_2}^{(i+0.5)Q_1} \left[(2Q_2 y - (2j+1)Q_2^2\right] p(y)dy > \sum_{j=-\infty}^{+\infty} \int_{(j+0.5)Q_2}^{(i+0.5)Q_1} \left[(2Q_2(j+0.5)Q_2 - (2j+1)Q_2^2\right] p(y)dy \\ &= \sum_{j=-\infty}^{+\infty} \int_{(j+0.5)Q_2}^{(i+0.5)Q_1} \left[(2jQ_2^2 + Q_2^2 - 2jQ_2^2 - Q_2^2\right] p(y)dy = 0 \end{aligned} \tag{7}$$

$$\begin{aligned} e_2(y) &= \sum_{j=-\infty}^{+\infty} \int_{(i-0.5)Q_1}^{(j-0.5)Q_2} \left[(2j-1)Q_2^2 - 2Q_2 y\right] p(y)dy > \sum_{j=-\infty}^{+\infty} \int_{(i-0.5)Q_1}^{(j-0.5)Q_2} \left[(2j-1)Q_2^2 - 2Q_2^2(j-0.5)\right] p(y)dy \\ &= \sum_{j=-\infty}^{+\infty} \int_{(i-0.5)Q_1}^{(j-0.5)Q_2} \left[(2jQ_2^2 - Q_2^2) - 2jQ_2^2 + Q_2^2\right] p(y)dy = 0\ (y \in [(i-0.5)Q_1, (j-0.5)Q_2)). \end{aligned} \tag{8}$$

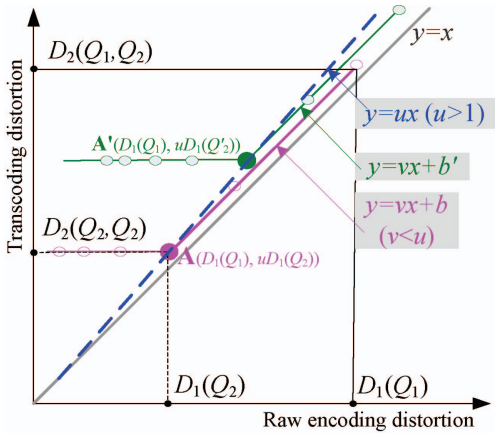

Fig. 4. Proposed distortion model for transcoding with respect to bitrate/distortion of raw video encoding.

approximated as

$$D_2(Q_1, Q_2) = \begin{cases} u \times D_1(Q_2), & Q_2 \geq Q_1 \\ v \times D_1(Q_1) + (u - v) \times D_1(Q_2), & Q_2 < Q_1 \end{cases} \tag{9}$$

where $D_1(Q_1) = b \times Q_1^{\beta}$ $(b, \beta > 0)$ is with the assumption of Cauchy distribution of quantized DCT coefficients [26]. The parameters u and v in (9) are subject to $u > 1$ and $v < u$ from the analysis of experimental statistics drawn in Fig. 1(c).

From (1) and (9), the output bitrate and distortion of transcoding are decided by both QPs of encoding and transcoding. The models of (1) and (9) are derived from the transcoding of bitrate adaption of H.264/AVC, and the following analyses and conclusions are mainly under such a situation. Other types of transcoding, such as format conversion among H.264/AVC, MPEG-2, and AVS-P2, are similar to this situation, i.e., their R-D models are all in a form of two-parameter formula.

III. Window-Level Rate Control

Most existing rate control algorithms for transcoding are based on the conventional rate control algorithms [18], [19] for general one-pass encoding. Although the bits consumption of pre-encoding is used to bit allocation for transcoding, such a usage does not make full use of the pre-encoding information for smooth visual quality. The two-pass rate control algorithm in [20] is regarded as being most efficient for smooth visual quality. In addition, it can be easily adapted to perform at a window basic unit (a group of successive frames or GoPs). For compliant buffer constraint, our previous proposed sliding window buffer check (SWBC) strategy [21] can be employed. Thus, a window-level rate control algorithm can be then proposed for both smooth visual quality and compliant buffer constraint as follows.

A. Traditional Two-Pass Rate Control Algorithm

In the two-pass rate control [20], the first-pass encoding yields the statistics of the entire sequence such as bits usage profile, scene change, and QPs. The second-pass encoding reallocates target bits among different scenes to achieve smooth visual quality by $R_{2,n} = k(Q_{1,n})^p$ [20], where $R_{2,n}$ is the amount of reallocated bits of scene n for the second-pass encoding, $Q_{1,n}$ is the quantization step size of the first-pass encoding, k is the model parameter, and p is the scene-dependent factor. For simplicity, the traditional two-pass rate control model at sequence level is named as sequence-level rate control (SRC). The fundamental of the two-pass rate control lies in the fact that there is a high correlation between bitrate profile of constant bitrate (CBR) encoding and QP profile of variable bitrate (VBR) encoding [20]. The first-pass encoding employs CBR rate control, such as JVT-H017r3 for H.264/AVC [18], [19]. The second-pass encoding computes a new set of QPs based on the bits and QP profiles of the first-pass encoding. $R_{2,n}$ can be rewritten as

$$b_{2,i} = kb_{1,i}(Q_{1,i})^p \tag{10}$$

TABLE I
Symbols Used in the Proposed Algorithm

Symbol	Meaning
N	Window size
R, F_r	Target bitrate and frame rate for transcoding
Y	Total number of bits of a window in the input bitstream
Z	Target bits quota of a window for transcoding
i, j	Frame index and iteration index($i = 0, \ldots, N-1$, $j \geq 0$)
$Q_{1,i}$, $b_{1,i}$	QP and frame bits of the input bitstream
$Q_{2,i}^j$, $b_{2,i}^j$	QP and frame bits of transcoding for the jth iteration
L, U, L_1, U_1	The iterative bounds of the algorithm in Table II
μ, η, k, d	Temporal variables for the iterative algorithm of Table II
r, s, u, v	Transcoding R-D model parameters in (1) and (9)
α, β	Parameters of Cauchy-based rate and distortion models
B, T, T_e	Buffer size, buffer delay, and initial encoder buffer delay
$Q_{2,i}$,	Final QPs of iterative algorithm
N	Window size
R, F_r	Target bitrate and frame rate for transcoding

TABLE II
Proposed Window-Level Rate Control for Transcoding

Step 1: *Initialization*: window size $N = T_e \times F_r$; target bits of a window $Z=N\times R/F_r$; $T_e = 0.8*T$ (referring to JM software of H.264/AVC); $s = 1$, $r = 1$ in (1) and (9).

Step 2: Read a segment of encoded bitstream for transcoding, and parse the frame bits usage $\{b_{1,i}\}$ and QP usage $\{Q_{1,i}\}$ $(i = 0, 1, \ldots, N-1)$.

Step 3: Compute QPs $\{Q_{2,i}\}$ for transcoding at window basic unit by using the iterative algorithm in Table III.

Step 4: In the current window, the buffer is checked with SWBC, and then the target bits quota $\{b_{2,i}\}$ and QP $\{Q_{2,i}\}$ of each frame are regulated accordingly if conflicting with buffer constraints.

Step 5: Perform transcoding of current window using $\{Q_{2,i}\}$ computed from Step 4.

Step 6: Update the model parameters of (1) and (9) with the actual coding bits $\{b(i)\}$ and QP $\{Q(i)\}$ of transcoding. In (1), if $Z < \sum_{i=0}^{N-1} b_{1,i}$, $s = \sum_{i=0}^{N-1} \left(b(i)/b_{2,i}\right)$, otherwise r and s are calculated from a liner regression function as

$$\begin{cases} r = \begin{vmatrix} \sum b(i) & N \\ \sum b(i) \times b_{2,i} & \sum b_{2,i} \end{vmatrix} \Big/ \begin{vmatrix} \sum b_{2,i} & N \\ \sum \left(b_{2,i}\right)^2 & \sum b_{2,i} \end{vmatrix} \\ s = \begin{vmatrix} \sum b_{2,i} & \sum b(i) \\ \sum \left(b_{2,i}\right)^2 & \sum b(i) \times b_{2,i} \end{vmatrix} \Big/ \begin{vmatrix} \sum b_{2,i} & N \\ \sum \left(b_{2,i}\right)^2 & \sum b_{2,i} \end{vmatrix}. \end{cases}$$

The parameters of (9) are updated using the same way as that of (1).

TABLE III
BITS REALLOCATION OF TWO-PASS ALGORITHM

Initialization:
$Y = \sum_{i=0}^{N-1} b_{1,i}$, $Z = N \times R/F_r$; $\mu = Z \Big/ \sum_{i=0}^{N-1} b_{1,i} \left(Q_{1,i}\right)^p$,
$\eta = Y \Big/ \sum_{i=0}^{N-1} b_{1,i} \left(Q_{1,i}\right)^p$;
L= 1E-7$\times\mu$, U= 1E+4$\times\mu$, L_1= 1E-7$\times\eta$, U_1= 1E+4$\times\eta$;
$b_{2,i}^0 = b_{1,i}$ and $Q_{2,i}^0 = Q_{1,i}$;
$k = 0$, $d = 0$, $j = 0$;
for ($d = U$ to L, $d *= 0.5$; j++)
$k += d$;
Compute $\left\{b_{2,i}^j\right\}$ as $b_{2,i}^j = k b_{2,i}^{j-1} (Q_{2,i}^{j-1})^p$ (j>1) based on (10);
Scale $\left\{b_{2,i}^j\right\}$ as $b_{2,i}^j = s \times b_{2,i}^j$;
Compute $\left\{Q_{2,i}^j\right\}$ from $\left\{b_{2,i}^j\right\}$ as $Q_{2,i}^j = b_{2,i}^{j-1} \times Q_{2,i}^{j-1} \Big/ b_{2,i}^j$;
Scale $\left\{Q_{2,i}^j\right\}$ as $Q_{2,i}^j = \frac{Q_{2,i}^j}{\sqrt[\beta]{u}}$;
if $\sum b_{2,i}^j > Z$
$k -= d$;
end if
end for
if ($Y > T$)
for ($d = U_1$ to L_1, $d *= 0.5$; j++)
$k += d$;
Compute both $\left\{b_{2,i}^j(Q_1)\right\}$ as $b_{2,i}^j(Q_1) = k \times b_{2,i}^{j-1}(Q_1) \times (Q_{2,i}^{j-1})^p$ (j >1) according to (10);
($b_{2,i}^j(Q_1)$ refers to the bits reallocation at the bitrate of encoding but not target bitrate)
Compute $\left\{b_{2,i}^j\right\}$ again as $b_{2,i}^j = r \times b_{2,i}^j(Q_1) + (s - r) \times b_{2,i}^j$;
Compute $\left\{Q_{2,i}^j\right\}$ from $\left\{b_{2,i}^j\right\}$ as $Q_{2,i}^j = b_{2,i}^{j-1} \times Q_{2,i}^{j-1} \Big/ b_{2,i}^j$;
Clip $\left\{Q_{2,i}^j\right\}$ as $Q_{2,i}^j = \max\{Q_{1,i} \frac{Q_{2,i}^j(Q_1)}{\sqrt[\beta]{u}}\}$;
if $\sum b_{2,i}^j > Y$
$k -= d$;
end if
end for
end if
$Q_{2,i} = Q_{2,i}^j$.

for frame i, where $b_{1,i}$ and $b_{2,i}$ are the number of bits of the first-pass and second-pass, respectively. $\{b_{1,i}\}$ and $\{Q_{1,i}\}$ ($i = 0, 1, \ldots, N-1$) are given by the first-pass encoding. p is derived experimentally, which is usually less than 1. In addition, the bits usage of the second-pass encoding $\{b_{2,i}\}$ ($i = 0, 1, \ldots, N-1$) is subject to $B_{\text{tot}} = \sum_{i=0}^{N-1} b_{2,i}$, where B_{tot} is the total bits budget for all frames. Thus, by summing both sides of (10) for all frames, k can be deduced as

$$k = \frac{B_{\text{tot}}}{\sum_{i=0}^{N-1} b_{1,i}(Q_{1,i})^p}.$$

B. Window-Level Rate Control Algorithm

Real-time video coding applications are very popular nowadays, such as video streaming over Internet and live TV program. To this aim, a CBR rate control algorithm [19] is preferred for the video distribution over networks under given bandwidths, and real-time encoding for video streaming instead of downloading of video over network is desired. Usually, there is a significant picture quality fluctuation for CBR algorithm, the reason of which lies in the unavailability of the global information of the sequence. To obtain smooth visual quality, the VBR algorithms such as [20] are preferred, which are designed for offline encoding without real-time and buffer constraints.

To obtain smooth visual quality in real-time encoding with compliant buffer constraint, a window-level rate control (WRC) algorithm is proposed. WRC takes the advantages of the two-pass rate control model for smooth visual quality and our previous proposed SWBC [21] for buffer control. In WRC, the traditional two-pass rate control algorithm [20] is adapted to be operated at a window basic unit instead of the entire sequence. WRC is actually a tradeoff between real-time encoding and visual quality smoothness.

As mentioned above, WRC is performed at the window basic unit. The window size is deduced from buffer constraint. In H.264/AVC, the hypothetical reference decoder (HRD) [22], [23] is used to address the buffer constraint. In HRD theory, the end-to-end delay consists of initial decoder delay (T_d) and initial encoder delay (T_e). T_d is the least time that the decoder should wait for the incoming bitstream before it starts to decode. The decoder may underflow without T_d because it needs a certain period to receive the encoded bits over a limited bandwidth channel. T_e is the time-delay of the encoded bits entering into communication channel after they are generated. In HRD, the frames are assumed to be encoded at a uniform time stamp [23], which indicates that the ith frame is encoded at $T_e + i\times(1/F_r)$ given the initial encoder delay T_e and frame rate F_r. Assuming that a window consists of $T_e \times F_r$ frames, the last frame (indexed by $T_e \times F_r - 1$) of the window would be encoded at $(T_e \times F_r - 1)\times(1/F_r) = T_e - 1(1/F_r)$ which is before the deadline of the first frame of the window to enter the communication channel, i.e., T_e. Thus, a window size of $N = T_e \times F_r$ frames can be processed in the manner of two-pass rate control algorithm to yield smooth visual quality. At the same time, the demand of real-time is not compromised since there is no additional time delay introduced by utilizing the inherent delay T_e of a streaming system. Certainly, the window size should be also concerned with computer power in real-time encoding because the bits and QP usages may not be ready for a window before its deadline. Fortunately, for transcoding, the frame bits and QP usages can be easily obtained for the entire sequence at the cost of negligible computational complexity by parsing input bitstream, so the window size is only decided by buffer constraint in transcoding.

In addition, WRC also needs a buffer control for its applications in real-time video encoding and transcoding. The traditional buffer control strategy [19] checks the buffer status at frame level as

$$B_e(j-1) + \left(b(j) - \frac{R}{f}\right) < B \tag{11}$$

where $B_e(j)$ is the buffer fullness of the jth frame, $b(j)$ represents the number of bits of the jth frame, and R and f are the channel bandwidth and frame rate, respectively. Once the buffer violates, QP will have to be updated immediately in the next frame. There would be visual quality fluctuation if scene

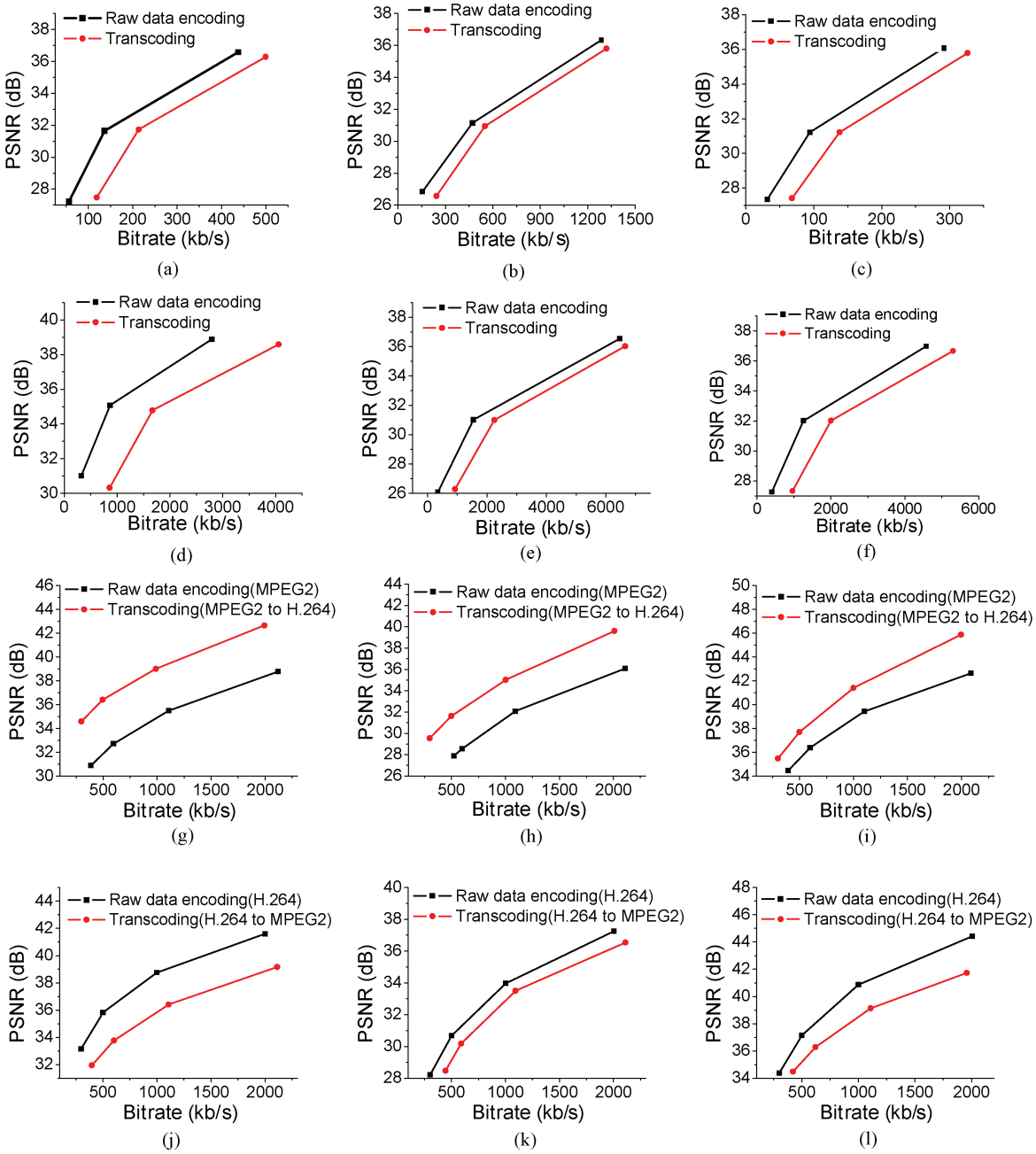

Fig. 5. R-D characteristics of raw video sequence encoding and transcoding. (a)–(f) Raw video sequence encoding and bitrate adaption transcoding of H.264/AVC. (g)–(i) Raw video sequence encoding using MPEG-2 and transcoding from MPEG-2 to H.264/AVC. (j)–(l) Raw video sequence encoding using H.264/AVC and transcoding from H.264/AVC to MPEG-2.

changes. Since the bits usage of the entire sequence is available in transcoding, a better buffer control strategy at window basic unit can be expected. It could promise smooth visual quality under buffer constraint when scene changes. SWBC is used in WRC for smooth visual quality by checking buffer status at window basic unit as

$$\sum_{i=j-N}^{j-1} \max\left(0, b(i) - \frac{R}{f}\right) + b(j) < B \tag{12}$$

where "max" is due to the constraint arrival time [23] of HRD, N is the window size, and B is the given buffer size. In SWBC, the QPs of a window of frames are regulated if the buffer overflows. Meanwhile, the window slides forward one frame each time as the buffer status is examined, so that SWBC is compatible with the traditional buffer measurement at frame level. This process is performed after gathering the frame bits usage of the encoded bitstream and followed by transcoding process. Moreover, the buffer status is updated with the actual number of coding bits during transcoding. Thus, the QPs of a window of frames are obtained as smooth as possible and the output bitstream can be well controlled under buffer constraint.

In [24], a theoretical window model was proposed to get better tradeoff between visual quality smoothness and buffer compliance. The theoretical analyses and experiments demonstrated that the smoother visual quality can be obtained given a

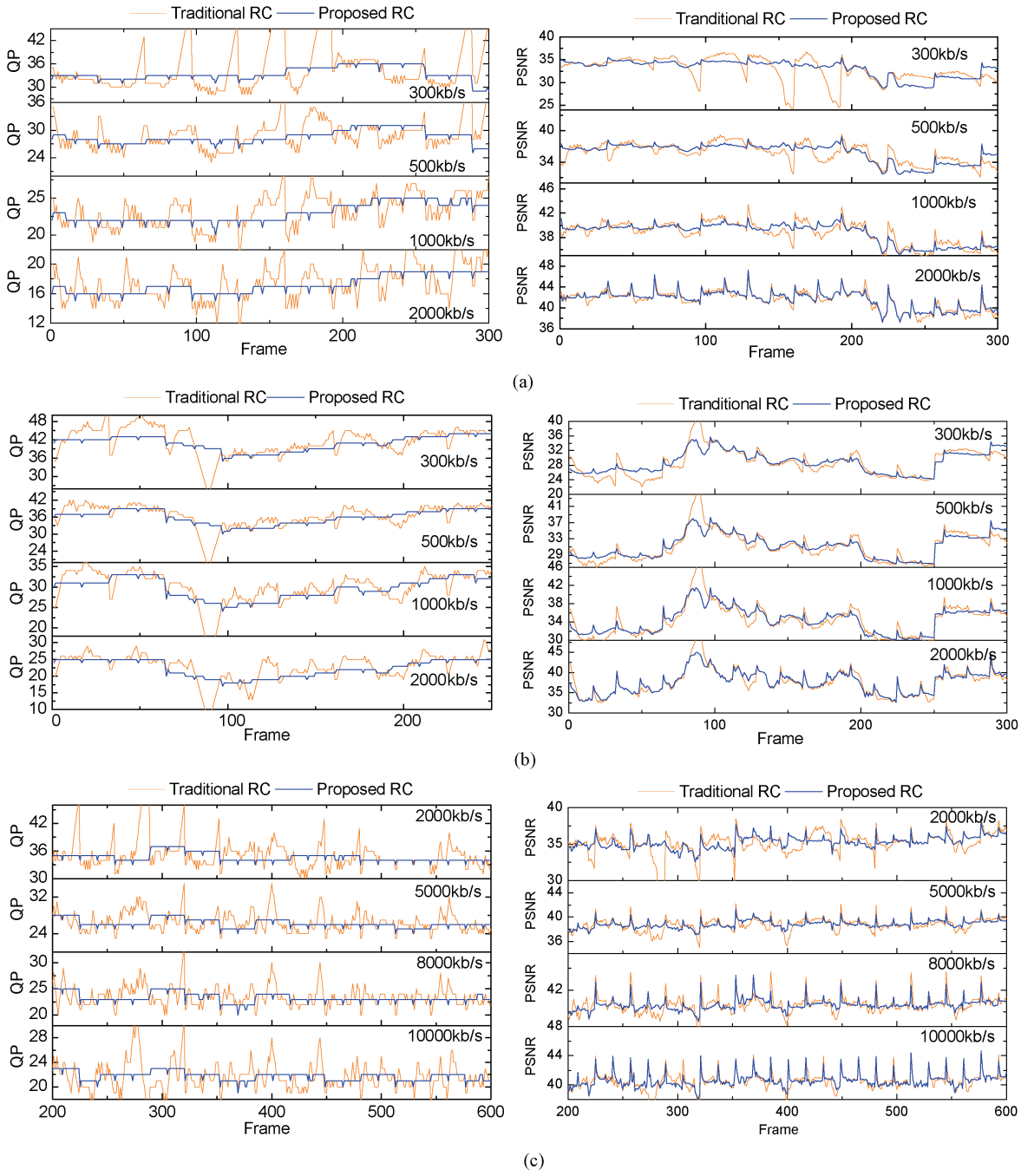

Fig. 6. Frame QP/PSNR curves for JVT-H017r3 and the proposed scheme. (a) *Foreman*. (b) *Football*. (c) *Crew*.

larger window size. The visual quality smoothness and buffer constraint was handled by window model in [24]. However, they are processed separately in this proposed window-level rate control algorithm, where the smooth visual quality benefits from the usage of two-pass model [20] at window basic unit, and the buffer constraint is managed by the proposed SWBC.

IV. Universal Rate Control Scheme for Transcoding

As discussed above, the proposed WRC can produce consistent visual quality with buffer compliance for real-time requirement. For general video encoding, WRC would consume many computational powers to gather the bits and QP usages from the first-pass encoding. For transcoding, it can easily obtain the bits and QP usages by parsing the encoded bitstream. Therefore, a real-time VBR transcoder can be achieved by applying WRC to a standard transcoder. However, significant bit control errors may be caused in transcoding if the traditional R-D models are applied. At such a situation, the newly proposed R-D models in (1) and (9) become necessary.

By combining the proposed R-D models and the proposed WRC, a universal rate control scheme for real-time transcoding is developed accordingly. In the proposed scheme, the window size is first determined by the given buffer constraint. Then, the bits and QP usages of a window are obtained by

TABLE IV
COMPARISON OF CODING EFFICIENCY OF ALL THE TEST ALGORITHMS

Resolution	Sequence	Target Bitrate (kb/s)	SRC			WRC			JVT-H017r3			Proposed		
			Bitrate (kb/s)	PSNR	Error (%)	Bitrate (kb/s)	PSNR	Error (%)	Bitrate (kb/s)	PSNR	Error (%)	Bitrate (kb/s)	PSNR	Error (%)
CIF	*Foreman*	2000	1744.50	41.01	−12.78	1982.18	41.55	−0.89	2000.02	41.46	0.00	1999.24	41.60	−0.04
		1000	820.55	37.97	−17.95	994.57	38.72	−0.54	1001.76	38.76	0.18	999.10	38.76	−0.09
		500	394.79	34.70	−21.04	492.1	35.73	−1.58	503.35	35.76	0.67	499.98	35.82	0.00
		300	256.56	32.37	−14.48	293.57	33.08	−2.14	303.17	32.92	1.06	297.41	33.15	−0.86
	Football	2000	1827.14	36.65	−8.64	1972.63	37.20	−1.37	2001.62	37.22	0.08	2003.41	37.25	0.17
		1000	999.68	33.77	−0.03	978.61	33.77	−2.14	1000.37	33.90	0.04	1001.61	33.97	0.16
		500	501.34	30.67	0.27	489.34	30.52	−2.13	500.18	30.61	0.04	501.34	30.67	0.27
		300	302.46	28.22	0.82	309.46	28.13	3.15	300.48	28.03	0.16	302.46	28.22	0.82
	Mobile	2000	1819.64	33.57	−9.02	1986.41	33.96	−0.68	2001.24	33.89	0.06	2003.40	34.13	0.17
		1000	902.67	29.98	−9.73	979.96	30.29	−2.00	1001.35	30.59	0.14	999.96	30.45	0.00
		500	439.03	26.39	−12.19	481.89	26.99	−3.62	503.24	27.10	0.65	501.89	27.07	0.38
		300	271.47	23.42	−9.51	308.2	24.01	2.73	306.33	24.08	2.11	302.20	24.08	0.73
	Silent	2000	1804.27	43.99	−9.79	1993.4	44.34	−0.33	2001.92	44.17	0.10	2007.02	44.43	0.35
		1000	907.73	40.40	−9.23	1012.17	40.66	1.22	1000.68	40.75	0.07	1000.17	40.87	0.02
		500	450.70	36.53	−9.86	487.31	37.05	−2.54	500.80	36.71	0.16	500.31	37.15	0.06
		300	289.71	34.13	−3.43	307.24	34.26	2.41	300.93	34.13	0.31	300.24	34.37	0.08
Average				33.99	−9.16		34.39	−0.65		34.38	+0.36		34.50	+0.14
					9.30			1.93			0.36			0.26
720P	*Night*	10 000	9029.31	38.00	−10.75	9810.02	38.19	−1.90	9999.57	38.14	0.17	9998.06	38.32	−0.39
		8000	6711.38	36.81	−19.20	7850.23	37.38	−1.87	8000.63	37.53	0.21	8001.94	37.49	−0.06
		5000	4091.97	35.02	−22.19	4851	35.83	−2.98	5000.99	35.89	0.30	4992.85	35.97	−0.14
		2000	1958.27	30.86	−2.13	1878.37	31.81	−6.08	2008.10	32.08	0.40	1995.85	31.98	−0.04
	Crew	10 000	9072.94	40.47	−10.22	9850.83	40.59	−1.49	10009.72	40.53	0.18	10047.53	40.71	−0.78
		8000	6385.32	39.60	−25.29	7846.59	40.01	−1.92	8006.67	40.04	0.23	8002.12	40.12	−0.17
		5000	3699.61	38.21	−35.15	4815.57	38.89	−3.69	5003.31	38.86	0.27	4949.55	39.01	−0.41
		2000	1580.51	34.65	−26.54	1871.15	35.56	−6.44	2001.92	35.55	0.32	1985.60	35.68	−0.22
	Harbor	10 000	8041.79	36.00	−24.35	9900.52	36.60	−1.91	10003.45	36.69	0.19	9981.57	36.84	−0.03
		8000	6152.47	35.01	−30.03	7920.00	35.80	−1.98	8002.15	35.96	0.34	8000.81	35.99	0.28
		5000	3546.69	32.98	−40.98	4857.51	34.00	−2.88	5006.06	34.20	0.40	4969.48	34.28	0.02
		2000	1381.36	28.68	−44.78	1866.03	30.10	−6.83	2010.78	30.50	0.62	1985.09	30.36	0.02
Average				35.52	−24.3		36.23	−3.16		36.33	+0.30		36.40	−0.16
					24.3			3.16			0.30			0.21

parsing the input bitstream. Next, the reallocation of bits for transcoding is performed by using (10) to achieve a VBR visual quality. In order to realize a two-pass rate control, (10) is implemented iteratively. In each iteration, a set of new QPs and bits usage of frames are derived from the previous one. Finally, the bits quota of each frame is further modulated by (1). A new QP is subsequently calculated by $Q_{2,i} = b_{1,i}Q_{1,i}/b_{2,i}$ for each frame after getting its bits quota $b_{2,i}$. The result is further modulated by (9). The flow of the proposed universal rate control scheme for transcoding is summarized as follows. The symbols used in the following algorithms are summarized in Table I.

V. EXPERIMENTAL RESULTS AND DISCUSSIONS

The proposed rate control scheme is implemented on JM14.2 of H.264/AVC under the following conditions: profile: 100, number of reference frames: 5, full search, search range: 16, GoP length: 32, rate-distortion optimization: on, and context-adaptive binary arithmetic coding. The presentation of experiments consists of two parts: one is for exhibiting the bit control accuracy and coding performance of the proposed algorithm, and the other is for showing the consistent visual quality of the proposed algorithm.

A. Coding Performance Comparison

In general, rate control algorithms for transcoding are application-oriented. As mentioned in Section II, each of [11]–[17] proposed a specific rate control algorithm for its own application. Regarding the proposed scheme, it could adapt to a universal transcoding purpose, including bitrate adaption, temporal/spatial resolution adaption, and video format conversion, because only the R-D relationship between encoding and transcoding as formulated in (1) and (9) is required, which could be obtained offline, or easily built from the R-D statistics of transcoding several frames.

The proposed scheme can be used to any transcoder, such as cascade, open-loop, and close-loop transcoders. For simplicity of presentation, only a cascade transcoder is chosen to be the testing platform. SRC [20], WRC, and JVT-017r3 [18], [19] are applied as benchmarks. JVT-H017r3 is performed at frame level for a fair comparison, since SRC, WRC, and the proposed universal rate control scheme all use the frame QP. The comparative experiments are made with a number of benchmark sequences, including *Foreman*, *Football*, *Mobile*, and *Silent* in CIF, *Night*, *Crew*, and *Harbor* in 720P (1280×720) high-definition (HD) resolution. The encoded bitstreams for transcoding are generated with the CBR which is the usual situation of video encoding over bandwidth-limited communication channels. The experimental results are summarized in the following.

First, we show briefly why it is necessary to propose the specific R-D model for transcoding by a number of experiments. The experiments are performed for three cases: bitrate adaption of H.264/AVC, format conversion between MPEG-2 and H.264/AVC, and format conversion between AVS-P2 and H.264/AVC. The experimental results are shown in Fig. 5. It can be seen that there exist large distances between the R-D curves of raw video sequence encoding and transcoding, which indicates that the R-D characteristics of encoding is

TABLE V
BD-PSNR COMPARISON BETWEEN THE PROPOSED SCHEME AND BENCHMARKS

Resolution	Sequence	SRC		WRC		JVT-H017r3	
		BD-PSNR (dB)	BD-Bitrate (%)	BD-PSNR (dB)	BD-Bitrate (%)	BD-PSNR (dB)	BD-Bitrate (%)
CIF	*Foreman*	−0.02	0.34	0.02	−0.37	0.09	−1.97
	Football	0.10	−2.14	0.07	−1.45	0.06	−1.34
	Mobile	−0.03	0.64	0.03	−0.39	−0.03	0.48
	Silent	−0.02	0.29	0.14	−2.64	0.25	−4.77
Average		0.01	−0.22	0.06	−1.21	0.10	−1.90
720P	*Night*	0.31	−5.86	0.00	−0.36	0.13	−6.34
	Crew	−0.05	3.08	0.00	−0.56	0.20	−9.90
	Harbor	−0.06	2.23	0.17	−4.61	0.11	−3.68
Average		0.07	−0.18	0.06	−1.84	0.15	−6.64

TABLE VI
TRANSCODING BETWEEN MPEG-2 AND H.264/AVC

Resolution	Sequence	Target Bitrate (kb/s)	Raw Video Sequence Encoding (MPEG-2; TM5)			Transcoding (MPEG-2 to H.264/AVC; the Proposed Algorithm)			Raw Video Sequence Encoding (H.264/AVC; JVT-H017r3)			Transcoding (H.264/AVC to Transcoding (MPEG-2; the Proposed Algorithm)		
			Bitrate (kb/s)	PSNR	Error (%)	Bitrate (kb/s)	PSNR	Error (%)	Bitrate (kb/s)	PSNR	Error (%)	Bitrate (kb/s)	PSNR	Error (%)
CIF	*Foreman*	2000	2118.30	38.77	5.92	1994.72	38.93	−0.26	2000.12	43.05	0.01	1998.10	38.91	−0.10
		1000	1109.10	35.50	10.91	991.65	37.65	−0.84	1002.41	39.94	0.24	1001.10	36.02	0.11
		500	595.40	32.73	19.08	497.03	35.86	−0.59	502.96	36.82	0.59	497.30	32.97	−0.54
		300	387.60	30.90	29.20	299.07	34.21	−0.31	302.74	34.68	0.91	296.70	31.35	−1.10
	Football	2000	2108.70	36.08	5.43	2008.95	36.07	0.45	2001.05	39.01	0.05	2000.70	36.25	0.04
		1000	1090.90	32.08	9.09	1001.24	34.03	0.12	1000.24	34.91	0.02	1002.30	33.01	0.23
		500	598.90	28.56	19.78	499.01	31.28	−0.20	500.17	31.52	0.03	498.80	29.18	−0.24
		300	–	–	–	296.86	29.34	−1.05	300.16	29.38	0.05	–	–	–
Average				33.52	+14.20		34.67	−0.34		35.18	+0.27		33.96	−0.23
					14.20			0.48			0.27			0.34
720P	*Night*	10 000	10061.9	37.62	0.62	10014.99	37.84	0.15	10017.36	39.44	0.17	10006.25	34.03	0.06
		8000	8061.7	36.66	0.77	7998.53	37.12	−0.02	8016.98	38.64	0.21	8004.21	33.72	0.05
		5000	5062.4	34.75	1.25	4997.37	36.10	−0.05	5014.88	37.02	0.30	5001.87	32.88	0.04
		2000	2068.9	30.85	3.45	1996.98	32.47	−0.15	2007.9	33.63	0.40	2000.14	30.40	0.01
	Harbor	10 000	10037.6	36.45	0.38	10004.72	35.72	0.05	10019.32	38.08	0.19	9998.61	34.92	−0.01
		8000	8034.4	35.42	0.43	8014.22	35.66	0.18	8027.41	37.18	0.34	8001.52	34.34	0.02
		5000	5034.6	33.23	0.69	4996.65	34.16	−0.07	5020.13	35.3	0.40	4998.22.	32.76	−0.04
		2000	2043.0	29.17	2.15	2000.09	31.45	0.00	2012.36	31.81	0.62	2005.15	30.07	0.26
Average				34.27	+1.22		35.07	+0.01		36.39	+0.33		32.89	+0.06
					1.22			0.08			0.33			0.05

much different from that of transcoding. Second, the proposed scheme and the benchmarks are compared with respect to bit control error and R-D performance. The comparative results on bitrate adaption of H.264/AVC are listed in Table IV, where up to 9.3% and 24.3% bit control errors are observed for SRC on CIF and HD sequences, respectively. Although the bit control error for WRC is less than 1.93% and 3.16% on CIF and HD sequences, respectively, the instant bit control error at window-level may be significant, which would result in buffer violation over constant bandwidth channels. Such cases are very common in video format transcoding between two video coding standards. From our experiments, the larger the difference of compression efficiency is, the more the bit control error is. Even for the transcoding within a video coding standard, the bit control error is still significant if the encoding parameters of transcoding are different from those of encoding. The bit control error can be handled well by the proposed scheme for transcoding as shown in Table IV. It can be observed that the bit control error of proposed scheme is less than 0.14% and 0.16% for CIF and HD sequences, respectively, while achieves 0.1 dB–0.3 dB PSNR improvement in average against the benchmarks. In addition, the Bjøntegaard delta (BD) PSNR and BD bitrate [27] performances are computed between the proposed scheme and other benchmark algorithms as tabulated in Table V. There are averagely 0.1 dB and 0.15 dB BD-PSNR improvements for the proposed scheme against the traditional algorithm JVT-H017r3 for CIF and HD sequences, respectively. We also list the experimental results of transcoding between MPEG-2 and H.264/AVC in Table VI for demonstrating the advantage of the proposed scheme at the situations of video format conversion. In Table VI, both of the MPEG-2 and H.264/AVC encoding use their default rate control algorithms, i.e., TM5 for MPEG-2 and JVT-H017r3 for H.264/AVC, and the transcoding employs the proposed algorithm. Due to the low coding efficiency of MPEG-2, the lowest bitrate cannot reach the target bitrate of 500 kb/s and 300 kb/s for *Football* and *Mobile*, which are labeled by "−" in Table VI. As shown in Table VI, a small bit control error below 0.05% can be observed for the proposed scheme. In Table VII, the experimental results of transcoding between H.264/AVC and AVS-P2 are tabulated, where the conclusion is similar to that for Table VI.

TABLE VII

TRANSCODING BETWEEN H.264/AVC AND AVS-P2 (AVS-P2 USES THE SAME RATE CONTROL SCHEME TO H.264/AVC)

Resolution	Sequence	Target Bitrate (kb/s)	Raw Video Sequence Encoding (AVS-P2)			Transcoding (AVS-P2 to H.264/AVC; the Proposed Algorithm)			Raw Video Sequence Encoding (H.264/AVC; JVT-H017r3)			Transcoding (H.264/AVC to AVS-P2; the Proposed Algorithm)		
			Bitrate (kb/s)	PSNR	Error (%)	Bitrate (kb/s)	PSNR	Error (%)	Bitrate (kb/s)	PSNR		Bitrate (kb/s)	PSNR	Error (%)
CIF	*Foreman*	2000	2002.43	41.53	0.12	1994.72	38.93	−0.26	2000.12	43.05	0.01	2001.55	40.14	0.08
		1000	1002.78	38.40	0.28	991.65	37.65	−0.84	1002.41	39.94	0.24	999.78	37.86	−0.02
		500	500.27	35.49	0.05	497.03	35.86	−0.59	502.96	36.82	0.59	500.66	35.25	0.13
		300	299.19	33.36	−0.27	299.07	34.21	−0.31	302.74	34.68	0.91	299.60	33.32	−0.13
	Football	2000	1999.65	38.58	−0.02	2008.95	36.07	0.45	2001.05	39.01	0.05	1999.40	36.61	−0.03
		1000	1000.35	34.42	0.04	1001.24	34.03	0.12	1000.24	34.91	0.02	1000.37	33.73	0.04
		500	500.41	30.84	0.08	499.01	31.28	−0.20	500.17	31.52	0.03	500.40	30.61	0.08
		300	300.40	28.64	0.13	296.86	29.34	−1.05	300.16	29.38	0.05	300.18	28.5	0.06
Average				34.38	+0.05		34.67	−0.34		35.18	+0.27		34.50	+0.03
					0.12			0.48			0.27			0.07
720P	*Night*	10 000	10009.06	38.88	0.09	10015.96	37.94	0.16	10017.36	39.44	0.17	9985.58	37.78	−0.14
		8000	8000.16	38.08	0.00	7983.94	37.51	−0.20	8016.98	38.64	0.21	7989.70	37.22	−0.13
		5000	5005.16	36.37	0.10	4993.78	36.34	−0.12	5014.88	37.02	0.30	5003.26	35.9	0.07
		2000	2021.21	32.79	1.06	1997.88	33.45	−0.11	2007.9	33.63	0.40	2001.93	32.57	0.10
	Harbor	10 000	9997.39	37.56	−0.03	10014.86	36.44	0.15	10019.32	38.08	0.19	9993.31	36.25	−0.07
		8000	7999.20	36.60	−0.01	8014.28	35.92	0.18	8027.41	37.18	0.34	7996.86	35.58	−0.04
		5000	4999.59	34.66	−0.01	4990.32	34.60	−0.19	5020.13	35.3	0.40	4999.68	34.09	−0.01
		2000	2005.22	30.94	0.26	2000.25	31.61	0.01	2012.36	31.81	0.62	2002.25	30.75	0.11
Average				36.33	+0.18		35.48	−0.02		36.39	+0.33		35.02	−0.01
					0.20			0.14			0.33			0.08

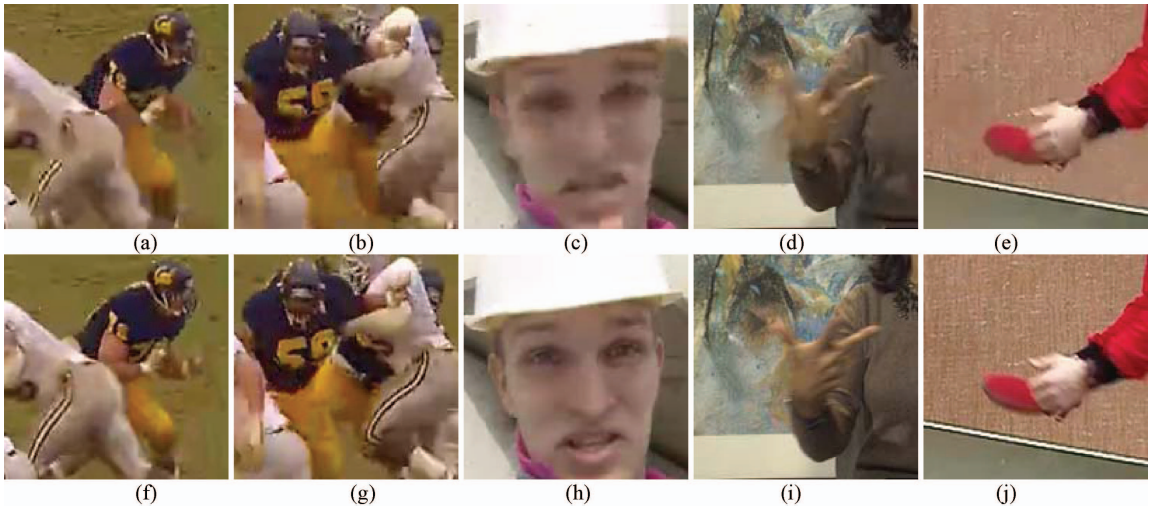

Fig. 7. Perceptual visual quality comparison of the proposed scheme and JVT-H017r3. (a)–(e) Coded by JVT-H017r3. (f)–(j) Coded by the proposed scheme at 500 kb/s (30 Hz). (a) and (f) are the 14th frame of *Football*, (b) and (g) are the 30th frame of *Football*, (c) and (h) are the 143th frame of *Foreman*, (d) and (i) are the 175th frame of *Silent*, and (e) and (j) are the 58th frame of *Tennis*.

B. Smooth Visual Quality Comparison

Although the existing rate control algorithms [11]–[17] utilized some of statistical information of pre-encoded bitstream for transcoding, e.g., MAD was predicted from both pre-encoding and previously coded frames in [16], and bit allocation was performed with the help of pre-encoded frame's complexity [17]. They all managed the information of pre-encoding in the manner of one-pass algorithm as JVT-H017r3 or TM5. The information of future frames was supposed to be utilized in bit allocation to achieve smooth visual quality. In Fig. 6, the frame QP/PSNR curves are compared between the proposed scheme and traditional one-pass algorithm of JVT-H017r3. The comparison is under the same buffer constraint, and the buffer constraint is obtained for the proposed scheme by employing the proposed SWBC. From Fig. 6, much more smooth visual quality can be observed for the proposed scheme. For the subjective visual quality comparison, two GoPs coded by the traditional and proposed algorithms are compared in Fig. 7, where the top-row pictures are coded by JVT-H017r3, and others are coded by the proposed algorithm. From Fig. 7, the perceptual visual quality of the proposed scheme is much better than that of JVT-H017r3 from the aspects of blurring and blocking artifacts. Note that JVT-H017r3 is directly implemented on the frames decoded with a cascade transcoder with MAD and complexity of the current frame being predicted from those of previous frames. Such a process is not applicable for other types of transcoders, such as open-loop and close-loop ones, since there are no decoded frames in these transcoders. However, the proposed scheme is suitable for any transcoding purposes since only the information of QP and bits consumption of encoding is used.

The two-pass rate control algorithm cannot be used in real-time encoding because its complexity is twice to a standard encoder. However, the QP and bits usages needed by the two-pass R-D model can be easily obtained for transcoding by parsing the encoded bitstream. Applying the proposed scheme, only the operations of parsing the syntax of picture header and locating picture header in a bitstream are introduced into a standard encoder. The computational complexity of the proposed scheme is comparable to that of the reference rate control [19], which all account for less 1% of the total encoding time of a H.264/AVC encoder with fast motion estimation. Therefore, the real-time transcoding is possible when the proposed scheme is used.

VI. Conclusion

In this paper, a novel R-D model was first established for transcoding. Second, a window-level rate control algorithm was developed for smooth visual quality and compliant buffer constraint. Then, a universal rate control scheme was proposed for various transcoding purposes based on the proposed R-D model and window-level rate control algorithm. Compared to the state-of-the-art algorithms, the proposed scheme achieved more bit control accuracy and much more consistent visual quality, while with low computational complexity. The proposed universal rate control scheme benefited the real-time video streaming applications, such as cable TV program, VOD, and live TV over Internet.

References

[1] A. Vetro, C. Christopoulos, and H. Sun, "Video transcoding architectures and techniques: An overview," *IEEE Signal Process. Mag.*, vol. 20, no. 2, pp. 18–29, Mar. 2003.

[2] H. Sun, W. Kwok, and J. W. Zdepski, "Architectures for MPEG compressed bitstream scaling," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 6, no. 2, pp. 191–199, Apr. 1996.

[3] Y. Nakajima, H. Hori, and T. Kanoh, "Rate conversion of MPEG coded video by re-quantization process," in *Proc. IEEE ICIP*, vol. 3. Oct. 1995, pp. 408–411.

[4] P. Yin, A. Vetro, B. Liu, and H. Sun, "Drift compensation for reduced spatial resolution transcoding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 12, no. 11, pp. 1009–1020, Nov. 2002.

[5] P. A. A. Assuncao and M. Ghanbari, "A frequency-domain video transcoder for dynamic bit-rate reduction of MPEG-2 bit streams," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 8, no. 8, pp. 953–967, Dec. 1998.

[6] G. J. Shen, Y. He, W. Cao, and S. Li, "MPEG-2 to WMV transcoder with adaptive error compensation and dynamic switches," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 16, no. 12, pp. 1460–1476, Nov. 2006.

[7] Q. Tang and P. Nasiopoulos, "Efficient motion re-estimation with rate-distortion optimization for MPEG-2 to H.264/AVC transcoding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 20, no. 2, pp. 262–274, Feb. 2010.

[8] J. D. Cock, S. Notebaert, P. Lambert, and R. Van de Walle, "Architectures for fast transcoding of H.264/AVC to quality-scalable SVC streams," *IEEE Trans. Multimedia*, vol. 11, no. 7, pp. 1209–1224, Nov. 2009.

[9] J.-C. Wu, P. Huang, J. J. Yao, and H. H. Chen, "A collaborative transcoding strategy for live broadcasting over peer-to-peer IPTV networks," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 21, no. 2, pp. 220–224, Feb. 2011.

[10] G. F. Escribano, H. Kalva, J. L. Martínez, P. Cuenca, L. Orozco-Barbosa, and A. Garrido, "An MPEG-2 to H.264 video transcoder in the baseline profile," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 20, no. 5, pp. 763–768, May 2010.

[11] K.-D. Seo, S.-H. Lee, J.-K. Kim, and J.-S. Koh, "Rate control algorithm for fast bit-rate conversion transcoding," *IEEE Trans. Consumer Electron.*, vol. 46, no. 4, pp. 1128–1136, Nov. 2000.

[12] M. Pantoja and N. Ling, "A two-level rate control approach for video transcoding," in *Proc. IEEE ICIP*, Nov. 2009, pp. 3701–3704.

[13] M. Pantoja and N. Ling, "Low complexity rate control for VC-1 to H.264/AVC transcoding," in *Proc. IEEE ISCAS*, May 2009, pp. 888–891.

[14] X. Y. Xiu, L. Zhuo, and L. S. Shen, "A H.264 bit rate transcoding scheme on PID control," in *Proc. ISCIT*, Oct. 2005, pp. 1039–1042.

[15] Y.-N. Xiao, H. Lu, X. Y. Xue, V.-A. Nguyen, and Y.-P. Tan, "Efficient rate control for MPEG-2 to H.264/AVC transcoding," in *Proc. IEEE ISCAS*, May 2005, pp. 1238–1241.

[16] X. Y. Wang, Y. Zhang, H. L. Li, and W. L. Zhu, "Adaptive rate control for dynamic bandwidth in video transcoding," in *Proc. IEEE ICCS*, Nov. 2008, pp. 773–777.

[17] J. Y. Yang, Q. H. Dai, W. L. Xu, and R. Ding, "A rate control algorithm for MPEG-2 to H.264 real-time transcoding," in *Proc. IEEE ICIP*, Sep. 2005, pp. 1995–2003.

[18] 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.

[19] S. W. Ma, Z. G. Li, and F. Wu, "Proposed draft adaptive rate control," document JVT-H017r3, Joint Video Team (JVT) of ISO/IEC MPEG and ITU-T VCEG, 8th Meeting, Geneva, Switzerland, May 2003.

[20] 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.

[21] L. Xu, J. Li, and N. Liao, *Method and Apparatus for Two-Pass Video Signal Encoding Using a Sliding Window of Pictures*, European Patent EP2200320, Jun. 23, 2010.

[22] J. Ribas-Corbera, P. A. Chou, and S. L. Regunathan, "A generalized hypothetical reference decoder for H.264/AVC," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, no. 7, pp. 674–687, Jul. 2003.

[23] *Time-Shift Causality Constraint on the CAT-LB HRD*, document JVT-E133, Joint Video Team (JVT) of ISO/IEC MPEG and ITU-T VCEG, 5th Meeting, Geneva, Switzerland, Oct. 2002.

[24] L. Xu, D. B. Zhao, X. Y. Ji, L. Deng, 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.

[25] *Information Technology Advanced Coding of Audio and Video Part2: Video*, AVS-P2 Standard Draft, Mar. 2005.

[26] 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.

[27] G. Bjøntegaard, "Calculation of average PSNR differences between RD-Curves," document VCEG-M33, ITU-T SG16 Q.6, Austin, TX, Apr. 2001.

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

He was a Post-Doctoral Researcher with the Department of Computer Science, City University of Hong Kong, Kowloon, Hong Kong, from August 2009 to July 2011. Currently, he is with the Department of Electronic Engineering, Chinese University of Hong Kong, as a Post-Doctoral Researcher. His current 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, in 1983, the University of Waterloo, Waterloo, ON, Canada, in 1985, and the Ph.D. degree from the University of Hagen, Hagen, Germany, in 1996.

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

Hanli Wang (M'08) received the B.S. and M.S. degrees in electrical engineering from Zhejiang University, Hangzhou, China, in 2001 and 2004, respectively, and the Ph.D. degree in computer science from the City University of Hong Kong, Kowloon, Hong Kong, in 2007.

From 2007 to 2008, he was a Research Fellow with the Department of Computer Science, City University of Hong Kong, and also a Visiting Scholar with Stanford University, Palo Alto, CA, invited by Prof. C. K. Chui. From 2008 to 2009, he was a Research Engineer with Precoad, Inc., Menlo Park, CA. From 2009 to 2010, he was an Alexander von Humboldt Research Fellow with the University of Hagen, Hagen, Germany. In 2010, he joined the Department of Computer Science and Technology, Tongji University, Shanghai, China, as a Professor. His current research interests include digital video coding, image processing, pattern recognition, and video content analysis.

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 the Institute of Computing Technology, Chinese Academy of Sciences (CAS), Beijing, China, in 2010.

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

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

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

Wen Gao (F'08) received the M.S. degree in computer science from the Harbin Institute of Technology, Harbin, China, in 1985, and the Ph.D. degree in electronics engineering from the University of Tokyo, Tokyo, Japan, in 1991.

He was a Professor of computer science with the Harbin Institute of Technology from 1991 to 1995 and with the Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China, from 1996 to 2005. He is currently a Professor with the School of Electronics Engineering and Computer Science, Peking University, Beijing, China. His current research interests include developing systems and technologies for video coding, face recognition, sign language recognition and synthesis, and multimedia retrieval. He has published four books and over 500 technical articles in refereed journals and proceedings in the areas of signal processing, image and video communication, computer vision, multimodal interface, pattern recognition, and bioinformatics.

Dr. Gao received several awards including five national awards for his research achievements and activities. He was the General Co-Chair of the IEEE ICME07, and the Head of the Chinese Delegation to the Moving Picture Expert Group of International Standard Organization. Since 1997, he has been the Chairman of the working group responsible for setting a national Audio Video Coding Standard for China.