

Real-time Encoding Frame Rate Control for H.263+ Video over the Internet

Hwangjun Song* Jongwon Kim * and C.-C. Jay Kuo *

August 31, 1998

Abstract

Most existing H.263+ rate control algorithms, e.g. the one adopted in the test model of the near-term (TMN8), focus on the macroblock layer rate control and low latency under the assumptions of with a constant frame rate and through a constant bit rate (CBR) channel. These algorithms do not accommodate the transmission bandwidth fluctuation efficiently, and the resulting video quality can be degraded. In this work, we propose a new H.263+ rate control scheme which supports the variable bit rate (VBR) channel through the adjustment of the encoding frame rate and quantization parameter. A fast algorithm for the encoding frame rate control based on the inherent motion information within a sliding window in the underlying video is developed to efficiently pursue a good tradeoff between spatial and temporal quality. The proposed rate control algorithm also takes the time-varying bandwidth characteristic of the Internet into account and is able to accommodate the change accordingly. Experimental results are provided to demonstrate the superior performance of the proposed scheme.

Keywords: H.263+, frame layer R-D model, rate control, frame rate control, VBR channel.

1 Introduction

Audiovisual communication over the Internet has gained more and more interest due to the fast development of networking and compression technologies recently. The digital video compression technique is one of the main components of an audiovisual communication system. International standards such as MPEG-1, MPEG-2, H.261 and H.263 have been developed to accommodate different application needs. New standards such as the advanced versions of H.263, MPEG-4 and MPEG-7 are also under development to achieve more functionalities. MPEG-1 and MPEG-2 are not suitable for low bit rate visual communications, e.g. video over the Internet, due to the associated large overhead. A near-term enhancement of H.263 known as H.263+ is an emerging video compression standard and the state-of-the-art for Internet video transmission [1]. To transmit compressed video efficiently over the Internet, we should consider both the complexity of the underlying video

*The authors are with the Integrated Multimedia Systems Center and the Department of Electrical Engineering-Systems, University of Southern California, E-mails: hwangjun@sipi.usc.edu, jongwon@sipi.usc.edu and cckuo@sipi.usc.edu

content and channel conditions, and develop an effective rate control scheme accordingly. Rate control of H.263+ video over the Internet is the main focus of this work.

It is well known that VBR (variable-bit-rate) video supports better quality than CBR (constant-bit-rate) video. Lakshman, Ortega and Reibman [2] classified VBR video into four classes: the unconstrained VBR, the shaped VBR, the constrained VBR and the feedback VBR. For the unconstrained VBR, if there are sufficient buffers at both the encoder and the decoder, rate control can be formulated as a optimization problem constrained to the bit budget only. Most video rate control algorithms have been developed under this scenario. ATM (Asynchronous Transfer Mode) is one of the most famous protocols, which support typically the constrained VBR. Rate control for video transmission over the ATM network has been studied by Chen and Lin [3], Hsu, Ortega and Reibman [4] and Reibman and Huskell [5]. Under the shaped VBR, the traffic patterns may be smoothed out at the cost of some additional delay while the content of the bit stream is unaffected. Under the feedback VBR, the network state information is accessible to the encoder, and the encoder can adjust its bit rate according to the change of the network traffic condition. It was argued in [2] that the feedback to the video encoder is one of the key characteristics of video transmission over packet networks. However, it is not a trivial task for the encoder to adapt to the state of the network quickly. Furthermore, it may not be efficient that the network state is sent back to the encoder via a feedback channel continuously due to the limited channel bandwidth. Thus, it requires a good coordination between the network feedback and the encoder to enhance the video quality and improve the efficiency of network usage. Generally speaking, rate control algorithms should be designed by considering channel characteristics as well as video characteristics.

In this work, we examine real-time H.263+ rate control for low-bit-rate VBR channels, specifically, unconstrained VBR and time-varying CBR. By time-varying CBR, we mean a VBR channel whose bandwidth is a piecewise constant function. It includes the feedback VBR and the renegotiated CBR, for which the encoding bit budget is allowed to change but only at isolated instants in time [2]. Furthermore, it can be the approximation of the time-varying channel such as the wireless communication channel if the time duration of CBR is relatively small compared to the bandwidth change speed. With the encoder/decoder buffers to absorb the short-term fluctuation, this approximation can be even more reasonable. If a rate control scheme works well for the time-varying CBR, this scheme can be extended to any other VBR channels.

One unique feature of our rate control scheme is to control spatial and temporal qualities simultaneously to improve the perceived quality for the low bit rate VBR channel. For the optimal spatial/temporal quality control, we need an integrated formulation. However, this is difficult to

perform. As a sub-optimal approach, we divide the temporal and spatial quality control into two separate tasks. The frame rate control considers the temporal quality while the frame layer rate control treats the spatial quality by adjusting the quantization parameter (QP). These two tasks have some implicit correlations in the proposed algorithm. Note that the proposed frame rate control is compatible with the H.263+ standard, since it allows the encoder to drop frames when needed. The frame drop information can be transmitted by TR (temporal reference) in the syntax bit of H.263+ or the customer picture clock frequency information.

For the unconstrained VBR channel, we need a basic unit to perform rate control. For example, the group of pictures (GOP) is generally used as a basic unit for the rate control in MPEG, which consists of one I-frame and several predictive frames (i.e. P- and B-frames) repeated periodically. Generally speaking, I-frames require a higher rate than predictive frames since motion estimation and compensation are not employed. Thus, for a very low bit rate environment, we have to reduce the number of I-frames, and the number of frames in a GOP can become larger. For H.263+, GOP is not suitable to serve as a basic rate control unit due to the long coding time-delay and the high computational complexity. Furthermore it is not efficient to consider I-frames and predictive frames simultaneously in H.263+. It is a general trend. Therefore, in this work, we define a GOP as a *group of predictive frames* without I-frame. The bit rate constraint is satisfied in each GOP. The proposed rate control use this new GOP as a basic rate control unit. Furthermore, we propose an effective encoding frame rate control algorithm with low-latency under time-varying CBR channel based on the R-D models. It can be more robust to the channel bandwidth fluctuation. The proposed frame rate control algorithm adopts a sliding window approach that does not impose additional encoding time-delay.

Even though several macroblock layer rate control algorithms were proposed before [6–8], there is little work about frame layer rate control for H.263+. Especially, since TMN8 rate control focuses on CBR channels and low-latency, frame layer rate control is not needed. However, frame layer rate control is required for efficient coding for VBR channels. In [9], we developed an efficient frame layer rate control algorithm with a moderate delay constraint. This work improves results in [9] by imposing the real-time implementation constraint. For real-time applications, the encoding time-delay and the computational complexity should be reduced. To achieve real-time frame rate control, frame layer rate-distortion (R-D) models will be derived.

This paper is organized as follows. The frame layer R-D model is studied in Section 2, which serves as the fundamental for the design of the two frame layer rate control algorithms in Sections 3 and 4. Rate control algorithms for unconstrained VBR channels and time-varying CBR channels

are examined in Sections 3 and 4, respectively. Several related technical issues are discussed in Section 5. Experimental results are presented in Section 6. Finally, concluding remarks are given in Section 7.

2 Rate and Distortion Models for Frame Layer

The encoding time delay and the computational complexity are critical to real-time video applications. A frame layer rate-distortion (R-D) model can be adopted to reduce them and will be derived in this section. The derived frame layer R-D model can be easily integrated with the R-D model used for existing macroblock layer rate control algorithms. Consequently, the proposed frame rate control is compatible with existing macroblock layer rate control. With our new framework, the encoding frame rate control is adopted as a main control mechanism with the macroblock layer rate control as an auxiliary control tool. The encoding frame rate control seeks a tradeoff between spatial and temporal quality to improve human perceived quality.

Generally speaking, there are two methods to achieve rate-distortion (R-D) modeling: statistical and experimental methods. One commonly used statistical model is to assume that the source signal has the generalized Gaussian distribution. For this model, one can get the closed-form R-D model [10], and other simplified models have also been proposed. Several models have been derived by using the experimental method, e.g. the quadratic rate model [11], the exponential model [12], the spline approximation model [13] and the normalized rate-distortion model [14], etc. On one hand, statistical models demand a lower computational complexity than experimental models. On the other hand, experimental models can provide a more accurate model through a data fitting process. TMN8 uses a statistical model for each macroblock. It is however too rough to model the statistics of a frame by estimating only the variance of the frame [13].

In this work, we propose a frame layer R-D modeling approach which constructs both the rate and distortion models with respect to the averaged quantization parameter (QP) of all macroblocks in each frame. It can be viewed as a hybrid statistical/experimental method. To be more specific, the quadratic rate model [11] and the affine distortion model are employed for the data fitting process while rate control results of the macroblock layer are used to determine the coefficients of the frame layer R-D model. Thus, the additional computational complexity required for the frame layer R-D modeling is negligible.

In terms of mathematics, the rate and distortion models can be written, respectively, as:

$$\begin{aligned}\hat{R}(\bar{q}_i) &= (a\bar{q}_i^{-1} + b\bar{q}_i^{-2})MAD(f_{ref}, f_{cur}), \\ \hat{D}(\bar{q}_i) &= a'\bar{q}_i + b',\end{aligned}$$

where a , b , a' and b' are model coefficients, f_{ref} is the reconstructed reference frame at the previous time instance, f_{cur} is the uncompressed image at the current time instance, \bar{q}_i is the average QP of all macroblocks in a frame, $\hat{R}(\bar{q}_i)$ and $\hat{D}(\bar{q}_i)$ are the rate and distortion models of a frame, respectively, and $MAD(f_{ref}, f_{cur})$ is the mean of absolute difference between f_{ref} and f_{cur} . Note that $MAD(f_{ref}, f_{cur})$ takes into account the dependency among frames. Coefficients a , b , a' and b' are determined by using the linear regression method. Conventionally, the R-D curve is computed based on integer QPs. In our case, \bar{q}_i can be a floating-point number since \bar{q}_i is the average QP of all macroblocks in a frame.

As done in MPEG-4 [15], we use an outlier removal process to improve the model accuracy. That is, if the difference between a data point and the derived model is greater than one standard deviation, the datum is removed. Based on filtered data, we can derive the rate and distortion models again.

We show the rate and distortion models in Figs. 2 (a) and (b), respectively, for the QCIF Salesman sequence, where the circle denotes the measure data points while the solid curve is the computed model. As shown in these two figures, the R-D modeling works reasonably well. The R-D modeling method in fact provides a very good approximation for all test sequences in our experiment.

3 Rate Control for Unconstrained VBR

In Section 3.1, short review of our previous work on the frame layer rate control without the real-time constraint is provided for the sake of completeness. Based on the rate and distortion models described in the previous section, we describe how to control the bit rates for each frame and adjust the Lagrange multiplier λ in Section refs:new-rc (It copes with the spatial quality control.), and the encoding frame rate to minimize the motion smoothness degradation in Section 3.3. (It treats the temporal quality control.) To reduce the encoding time-delay and complexity, we propose a sliding window approach. It is worthwhile to emphasize that the proposed rate control algorithm includes existing H.263+ macroblock layer rate control algorithms as one component. Thus, our work is compatible with existing rate control schemes [6–8].

3.1 Review of Previous Frame Layer Rate Control Work

The frame layer rate control to reduce quality fluctuation between adjacent frames was examined in our previous work [9]. If sufficient buffers at the encoder and the decoder are available, rate control under unconstrained VBR channels is the same as an optimization problem constrained by

the total bit budget. Therefore, we can formulate it as follows.

Determine $q_i, i = 1, 2, \dots, N_{gop}$, to minimize

$$\sum_{i=1}^{N_{gop}} d_i(q_1, q_2, \dots, q_i), \text{ subject to } \sum_{i=1}^{N_{gop}} r_i(q_1, q_2, \dots, q_i) \leq B_{gop}, \quad (1)$$

where N_{gop} is the frame number of a GOP, B_{gop} is the given bit budget for a GOP, $d_i(q_1, q_2, \dots, q_i)$ and $r_i(q_1, q_2, \dots, q_i)$ are the distortion measure and the allocated bit budget for the i th frame, respectively. Since it is very difficult to get the optimal solution in a GOP of H.263+ due to the long encoding delay and the high computational complexity, an effective sub-optimal approach was proposed in [9], where the global optimization problem is simplified as follows.

Determine $\vec{q}_m, m = 1, 2, \dots, M$, to minimize

$$\sum_{m=1}^M (D_m(\vec{q}_m) + \omega_q E_m(\vec{q}_m)), \text{ subject to } \sum_{m=1}^M r_m(\vec{q}_m) \leq B_{subgop} M, \quad (2)$$

where $\vec{q}_m = (q_{m,1}, q_{m,2}, \dots, q_{m,N_m})$ is the quantization parameter vector for the m th sub-GOP, N_m is the encoded frame number of the m th sub-GOP, $r_m(\vec{q}_m)$ is the assigned number of bits for the m th sub-GOP, M is the number of sub-GOPs in a GOP, N_{subgop} is the total frame number of a sub-GOP and N_{gop} is the total frame number of a GOP, ω_q is the weighting factor for the quality change and

$$D_m(\vec{q}_m) = \frac{1}{N_m} \sum_{i=1}^{N_m} d_i(q_1, q_2, \dots, q_i),$$

$$E_m(\vec{q}_m) = \frac{1}{N_m} \sum_{i=1}^{N_m} (d_i(q_1, q_2, \dots, q_i) - d_{i-1}(q_1, q_2, \dots, q_i, q_{i-1}))^2. \quad (3)$$

It is clear that we have the following relationship

$$\sum_{m=1}^M N_m = N_P, \quad B_{subgop} = \frac{N_{subgop}}{N_{gop}} \cdot B_{gop}.$$

By pursuing a reasonable sub-optimal solution to (2), the encoding time-delay and computational complexity can be significantly reduced. Details are referred to [9]. Despite this effort, the resulting algorithm is still not sufficient enough for real-time applications due to computational complexity and encoding time-delay proportional to the size of a sub-GOP. Thus, we have to change the problem formulation.

3.2 Spatial Quality Control: Frame Layer Rate Control with Adaptive Lagrange Multiplier

3.2.1 Low Complexity Formulation of Frame Layer Rate Control

Here, we consider a new and low complexity formulation of the rate control problem for real-time video applications based on R-D models in Section 2.

Determine q_i , $i = 1, 2, \dots, N_{gop}$, to minimize

$$\hat{D}_i(\bar{q}_i) + \omega \|\hat{D}_i(\bar{q}_i) - D_{i-1}\|, \quad (4)$$

subject to

$$\sum_{i=1}^{N_{gop}} R_i(\bar{q}_i) \leq B_{gop}, \quad (5)$$

where \hat{D}_i is the estimated distortion of the current frame, D_{i-1} is the actual distortion of the previous frame, and ω is the weighting factor for quality change between adjacent frames. The second term of (4) is included to reduce the flickering artifact caused by the abrupt quality change between adjacent frames.

By using the Lagrangian method, we can define a penalty function for the i th frame by combining the cost function and the constraint through a Lagrange multiplier. To satisfy the bit budget constraint smoothly, we use CBR as a reference.

$$P_i(\bar{q}_i) = \hat{D}_i(\bar{q}_i) + \omega \|\hat{D}_i(\bar{q}_i) - D_{i-1}\| + \lambda_i \cdot \max\{\hat{B}_i^{res}, 0\}, \quad (6)$$

$$\hat{B}_i^{res}(\bar{q}_i) = \sum_{j=1}^{i-1} R_j + \hat{R}_i(\bar{q}_i) - t_i \cdot C_{cbr}, \quad (7)$$

where $P_i(\bar{q}_i)$ is the cost function for i th frame, λ_i is the Lagrange multiplier for the i th frame, R_j is the used bit rate for the j th frame, t_i is the time instance of the i th encoded frame after the start of GOP and the average channel bandwidth

$$C_{cbr} = \frac{B_{gop}}{T_{gop}},$$

and where T_{gop} is the time duration of a GOP.

Based on the rate and distortion models in Section 2, we can determine the optimal QP to minimize the above penalty function. It was shown in [13] that $P_i(\bar{q}_i)$ is a convex function generally. Thus, we can get its optimal solution by using the gradient method.

$$\bar{q}_i^* = \arg \min_{\bar{q}_i} P_i(\bar{q}_i). \quad (8)$$

What we actually need is not \bar{q}_i^* , but $\hat{R}_i(\bar{q}_i^*)$ which is the target bit budget for the i th frame.

Let us consider one special case to the above optimization problem, i.e. $\hat{B}_i^{res} \leq 0$ and $\hat{R}_i(\bar{q}_i^*) \leq (t_{i+1} - t_i)C_{cbr}$. Then, it is not difficult to see that $\hat{R}_i(\bar{q}_i^*) = (t_{i+1} - t_i)C_{cbr}$. In this case, existing macroblock layer rate control algorithms effectively allocate the bit budget to each macroblock with the solution $R_i(\bar{q}_i^*)$. However, if $\hat{B}_i^{res} > 0$, we are able to provide a better solution than that given by existing macroblock layer rate control schemes.

3.2.2 Adaptive Adjustment of Lagrange Multiplier

The Lagrange multiplier method has been widely employed for bit rate allocation in video coding [16–18]. The Lagrange multiplier λ that satisfies the given bit budget (1) can guarantee the global optimality of the solution for both independent and dependent coding schemes [17, 18]. Furthermore, the optimal solution can be computed by the using gradient method under the convex hull assumption. In practice, an iteration process is however needed to determine the optimal Lagrange multiplier λ [16, 19]. Consequently, these approaches are not suitable for real-time video applications due to long time delay and a high computational cost.

In [20], Choi and Park attempted to adjust the Lagrange multiplier based on the buffer occupancy, and derived a discrete linear equation for buffer occupancy. Then, they proved the stability of the solution based on the Lyapunov theory. Lin and Chen [21] tried to control the Lagrange multiplier to avoid buffer underflow and overflow for the ATM network. Wiegand *et al.* [8] proposed a frame-to-frame update of the value of λ by using the least mean-square adaptation in the selection of an efficient macroblock coding mode.

The optimization of H.263+ coded video constrained by the bit budget is studied here. Since the Lagrange multiplier will be adjusted adaptively on the fly in real-time rate control, only a sub-optimal solution with a low computational complexity will be considered. The adaptive adjustment of λ is achieved by using the following rule

$$\lambda_{i+1} = \begin{cases} \lambda_i & \text{if } B_{-1} \leq B_i^{res} \leq B_1, \\ \lambda_i + \Delta_k & \text{if } B_k \leq B_i^{res} \leq B_{k+1}, \\ \lambda_i - \Delta_{-k} & \text{if } B_{-(k+1)} \leq B_i^{res} \leq B_{-k}, \end{cases} \quad (9)$$

where λ_{i+1} and λ_i are the Lagrange multipliers for the $i + 1$ th and the i th frames, Δ_k and Δ_{-k} are step sizes satisfying the monotonically increasing property

$$0 \leq \Delta_{|k|} \leq \Delta_{|m|}, \quad \text{if } |k| \leq |m|, \quad (10)$$

B_i , $i = \pm 1, \pm 2, \dots$, are threshold values as given in Fig. 1, and

$$B_i^{res} = \sum_{j=1}^i R_j - t_i \cdot C_{cbr} \quad (11)$$

is the accumulated residual bits up to frame i .

The rationale of the above adjustment rule is given below. As shown in (7), the Lagrange multiplier λ is expressed as a function of the accumulated residual bit B_i^{res} up to frame i . Note also that a smaller value of λ implies higher bit rates and better video quality. If B_i^{res} is located in region B_i with $i \geq 1$ as shown in Fig. 1, it means that too many bits have been used already so that we should reduce the bit rate for future frames by increasing the value λ_{i+1} of the Lagrange multiplier at time t_{i+1} . The Lagrange multiplier λ should be continuously increased until the residual bit is located in zone 0, i.e. in the interval (B_{-1}, B_1) . The opposite arguments are applicable to B_i with $i \leq -1$ as shown in Fig. 1. Since fewer bits are used so far, bit rates for the future frames can be increased by decreasing the Lagrange multiplier λ . Although our arguments for the above Lagrange multiplier control rule are heuristic, the rule is actually consistent with the conditions stated in [20] and [8].

As given in (9), we should determine the values of Δ_k and B_i . The following empirical values are chosen in our experiments. First, we choose the values of B_i and Δ_k to be symmetric, i.e.

$$B_i = -B_{-i}, \quad \text{and} \quad \Delta_k = \Delta_{-k}.$$

Then, B_i is chosen to be:

$$B_1 = \frac{F_{cur}^{int}}{30} \cdot C_{cbr}, \quad R_k = kR_1, \quad \text{for } k = 2, \dots, 7,$$

where F_{cur}^{int} is the current interval of encoded frames, 30 is the frame capturing rate. Δ_k is chosen to be:

$$\begin{aligned} \Delta_2 &= \frac{\epsilon}{2}, \quad \Delta_3 = \epsilon, \quad \Delta_4 = 2\epsilon \\ \Delta_5 &= 4\epsilon, \quad \Delta_6 = 8\epsilon, \quad \Delta_7 = 16\epsilon, \end{aligned}$$

where ϵ is a constant that can be slightly different for different test sequences. Generally speaking, the same value can be used for video with a similar amount of motion activities.

3.3 Temporal Quality Control: Motion-based Frame Rate Control with A Sliding Window

One objective of our rate control scheme is to keep the quality of P-frames nearly constant (or varying very slowly). Since each P frame is used as a reference frame for the following P frames, quality degradation propagates to later frames when a P frame is degraded severely. In fact, the R-D characteristics of predictive frames is greatly related to the motion in underlying video. The proposed rate control algorithm adjusts the frame rate adaptively based on the motion information

in the sliding window to reduce the image quality variation between adjacent frames. Since it is difficult to support both good spatial quality and temporal quality (in terms of motion smoothness) at very low bit rates, an encoding frame rate control is adopted for a tradeoff of spatial/temporal quality based on the motion information in video.

It is observed that human eyes are sensitive to the abrupt encoding frame rate (or interval) change. Our scheme aims at the reduction of temporal degradation in terms of motion jerkiness perceived by human beings. At the same time, no encoding time-delay is imposed for real-time processing. The next encoding frame position is estimated by motion information within a fixed length sliding window to avoid an abrupt frame rate change.

By adjusting the frame rate, we can avoid or reduce the sudden frame skipping in existing rate control algorithms, which degrades motion smoothness disastrously. Two problems have to be addressed for frame rate control. They are: when the frame rate should be changed, and how to change the encoding frame rate to preserve motion smoothness. They are considered in Sections 3.3.1 and 3.3.2, respectively.

Before a detailed presentation of our rate control algorithms, it helps to give an overview of the framework as shown in Fig. 3, where we consider an image sequence is captured under a constant frame rate, say, 30 frame per seconds (fps), as given in the top row of the figure. The rate control algorithm is able to select a sequence of frames for encoding and transmission as shown in the 2nd row of the figure. The number of frames skipped between two consecutive frames is called the encoded frame interval. In TMN8, the encoded frame interval is kept the same for the entire video. In our scheme, we allow the encoded frame interval to change depending on video and channel characteristics. To decide which captured frame to be encoded next (or the proper number of frames to skip), we concentrate on the statistics in a sliding window of length 12, which contains the current encoded frame and its previous 11 captured frames. To give an example, two sliding windows are labeled explicitly in the figure. Based on the information of the previous sliding window, the rate control algorithm decides to skip 3 frames and choose the 4th frame that follows as the frame for coding. Then, the window moves the position labeled with the current sliding window. Based on the characteristics of the new window, the rate control algorithm will select the next frame to encode.

3.3.1 Motion Change Detection in A Sliding Window

We need some measure to detect motion change in video. In this work, the histogram of difference (HOD) is adopted since HOD is very sensitive to local motion in video [22]. We define the distortion

measure between two frames f_n and f_m as:

$$D_h(f_n, f_m) = \frac{\sum_{i>|TH_{zero}|} hod(i)}{N_{pixel}}, \quad (12)$$

where $hod(i)$ is the histogram of the difference image, TH_{zero} is threshold value for detecting the closeness of the position to zero, and N_{pixel} is the total number of pixels. After the HOD values of consecutive frames in a sliding window are calculated, the estimated HOD value \hat{D}_h for the next frame can be calculated by

$$\hat{D}_h = D_h + \omega_h s_h, \quad (13)$$

where D_h is the HOD value between the two last encoded frames in the sliding window, ω_h is a weighting factor and s_h is the slope of approximating line which minimizes the mean square error of HODs in the sliding window. If the *MAD* measure of Section 2 is used, the computational complexity can be reduced. In fact, any measure that can detect the motion in video can be employed.

It is interesting to point out that s_h is related to motion change in video. The positive value means that the motion in video becomes faster while the negative value means that the motion in video becomes slower. Also, a larger value of $|s_h|$ implies a larger motion change.

3.3.2 Rule for Encoding Frame Interval Change

Based on the motion change information in the sliding window, we can determine the rule for the change of the encoding frame interval. Let \bar{D}_h denote the mean of all HOD values of frames in the sliding window. We can adjust the encoding frame rate based on the difference $\delta_{D_h} = \hat{D}_h - \bar{D}_h$ as follows.

- If $\delta_{D_h} \geq TH_\delta$, the encoding frame interval is increased by $\Delta F^{int}(F_{cur}^{int})$.
- If $\delta_{D_h} \leq -TH_\delta$, the encoding frame interval is decreased by $\Delta F^{int}(F_{cur}^{int})$.
- If $|\delta_{D_h}| < TH_\delta$, the encoding frame interval remains the same.

In above, the threshold value TH_δ is chosen to be the averaged *HOD* over the the sliding window, and F_{cur}^{int} is the current encoding frame interval, ΔF^{int} is a function of F_{cur}^{int} . In addition, we need the following rule to compensate for slow and steady motion change.

- If $B_i^{res} > TH_{B1}$, the encoding frame interval is increased by $\Delta F^{int}(F_{cur}^{int})$.
- If $B_i^{res} < TH_{B0}$ and the current encoding frame interval is greater than the frame capturing interval, the encoding frame interval is decreased by $\Delta F^{int}(F_{cur}^{int})$,

where TH_{B0} and TH_{B1} are threshold values.

Parameters TH_δ , TH_{B0} and TH_{B1} work as thresholding values for controlling the tradeoff between temporal and spatial quality. When the frame rate is changed, the frame rate is kept constant for a period of time as long as that of the sliding window to avoid the frequent occurrence of frame rate change. The first rule is for short-term motion change while the second rule is for the long-term motion. Furthermore, the following empirical rule to choose $\Delta F^{int}(F_{cur}^{int})$ is adopted in our experiment:

$$\Delta F^{int}(F_{cur}^{int}) = \lceil 0.3 \cdot F_{cur}^{int} \rceil,$$

where $\lceil x \rceil$ means the smallest integer greater than x .

4 Frame Rate Control for Time-varying CBR Channel

Time-varying CBR channels include the feedback VBR and the renegotiated CBR channel [2], where the available bandwidth is time-varying and can be modeled well with a piecewise constant function. This model will be valid if the bandwidth change speed is relatively low in comparison with the duration when the bandwidth is kept at a constant level. This model will be even more reasonable if there are buffers at the encoder and the decoder ends to offset the channel bandwidth fluctuation effect.

Based on the rate and distortion model equations (1) in Section 2, we can estimate the distortion of current frame. Since $\bar{q}_i > 0$, the estimated distortion \hat{D} can be expressed as

$$\hat{D} = a' \frac{aMAD(f_{ref}, f_{cur}) + \sqrt{(aMAD(f_{ref}, f_{cur}))^2 + 4bB(F_{cur}^{int})MAD(f_{ref}, f_{cur})}}{2B(F_{cur}^{int})} + b',$$

$$B(F_{cur}^{int}) = \frac{F_{cur}^{int}}{30} C_{tcb},$$

where C_{tcb} is the current channel bandwidth and F_{cur}^{int} is the current encoding frame interval under the assumption the camera captures frames at a rate of 30 fps. Note that \hat{D} increases when fast motion change occurs (with an increasing MAD) or when the channel bandwidth decreases (with a decreasing $B(F_{cur}^{int})$) suddenly.

Now, let us consider the rate control scheme. If the spatial quality is below a tolerable level due to fast motion change or sudden channel bandwidth decrease, we should reduce the temporal quality and improve the spatial quality in order to reduce the flickering artifact. At the same time, it is still desirable to control the temporal quality degradation. On the contrary, if the spatial quality is above a certain level, we should increase the temporal quality. Based on the discussion, the encoding frame rate control algorithm can be stated as follows:

- If $\hat{D} > TH_{D1}$, increase the encoding frame interval by $\Delta F^{int}(F_{cur}^{int})$.
- If $\hat{D} < TH_{D2}$ and the current frame interval is greater than the frame capturing interval, the encoding frame interval is decreased by $\Delta F^{int}(F_{cur}^{int})$.

TH_{D1} and TH_{D2} are two threshold values to be selected (see Section 6.2). By adopting this rate control scheme, we can avoid the abrupt change of the encoding frame rate and improve the spatial quality. This algorithm can be applied in real-time processing since the computational complexity is very low and low latency can be guaranteed.

5 Other Related Issues

It is worthwhile to comment on spatial and temporal artifacts for coded video. Blocking, ringing and texture deviation artifacts are often observed in low bit rate video as spatial quality degradations. As to temporal visual degradation, few research results are available. Flickering (or blinking) and motion jerkiness are the two major artifacts often observed. The flickering artifact is caused by the fluctuation of spatial image quality between adjacent frames while motion jerkiness occurs when there is an abrupt change of the coding frame rate or when the frame rate goes below a certain threshold required to generate smooth motion. In this research, the conventional PSNR (peak signal-to-noise ratio) quality measure is reported for the comparison purpose. The difference of PSNR values of adjacent frames is used to measure quality change (or flickering). Despite the fact that the change of PSNR does not correspond to flickering completely, we observe that the flickering effect can be reduced by keeping the image quality of each frame almost constant. Since the measure of the flickering effect is a very complicated and challenging problem, the subjective visual test is always needed to evaluate the performance of the proposed schemes. We attempt to comment on the subjective quality evaluation of the coded video whenever it is appropriate. In our experiment, macroblock layer rate control of TMN8 [7] is employed and the practical implementation is based on UBC H.263+ source codes [23].

At low bit rates, the frequent use of I frames causes disastrous unsmooth motion and time delay. Hence, the H.263+ standard recommends the macroblock-based update at least once every 132 times. Therefore, it is more common to insert an I frame only when the scene change is detected or the accumulated mismatch error cannot be recovered without the I frame. In the proposed rate control algorithm, we calculate HOD to estimate the motion change in video. By using HOD, we can detect the scene change and determine the I frame positions efficiently.

The frame interpolation technique [24] is required at the decoder to guarantee smooth display. Several methods have been proposed, including intra-frame filtering, motion adaptive filtering and

motion-compensated up-conversion. The intra-frame filtering method is the simplest way to increase the frame rate by repeating the current frame until a new frame is received. Motion-based interpolation leads to more smooth motion. However, its computational complexity is higher. In our current implementation, intra-frame filtering is adopted. As shown in the experimental section, intra-frame filtering does provide reasonable results.

6 Experimental Results

In our experiment, the macroblock layer rate control algorithm of TMN8 [7] is employed, and the implementation is based on the UBC H.263+ source code [23].

6.1 Unconstrained VBR Channel

First, experimental results are presented to demonstrate the performance of our rate control scheme for the unconstrained VBR channel in comparison with TMN8. The three test sequences are the QCIF “Salesman”, “Akiyo” and “Silent Voice”, and the target average bit rate is 24 kbps. In this experiment, we perform the rate control on a sequence consisting of 100 P frames corresponding to about 3.3 seconds (with a frame rate of 30 frames per second).

The performance comparison of three rate control algorithms for the three test sequences is shown in Table 1, where the average PSNR value and the standard deviation of PSNR are computed based on coded frames only. One can see from this table that even though TMN8 with 2-frame skipping has the best average PSNR, but its PSNR values have the largest standard deviation among the three rate control schemes. On the other hand, TMN8 with 1-frame skipping has the smallest standard deviation, but the largest average PSNR value. Our algorithm reduces the fluctuation of PSNR of TMN8 with 2-frame skipping and improves the average PSNR of TMN8 with 1-frame skipping.

The encoded frame position information for unconstrained VBR is listed in Table 2. With our algorithm, the encoding frame interval is changed from 1-frame skip to 2-frame skip from frame no. 54 to frame no. 72 for Salesman, from frame no. 54 to frame no. 100 for Akiyo and from frame no. 40 to frame no. 94 for Silent Voice. However, it is observed that human eyes cannot detect the encoded frame interval change effect due to the smooth encoding interval change as shown in Table 2.

The PSNR and rate plots as a function of the frame number are shown in Figures 4-6. For the PSNR plots, we see that the proposed rate control scheme follows the TMN8 with 2-frame skipping closely for most parts of the Salesman sequence. The TMN8 with 2-frame skipping performs

Table 1: Performance comparison with TMN8 under unconstrained VBR. Target average rate is 24kbps.

Method	Sequence	Avg PSNR	STD of PSNR	NO of Enc. frms
TMN8(2-frame skip)	Salesman	31.0904	0.9064	28
	Akiyo	35.5266	0.9188	29
	Silent Voice	30.7279	0.3822	29
TMN8(1-frame skip)	Salesman	30.6333	0.8295	42
	Akiyo	34.8258	0.7010	43
	Silent Voice	30.2198	0.4013	42
Proposed method	Salesman	30.9296	0.8076	32
	Akiyo	35.1607	0.8516	37
	Silent Voice	30.8802	0.4026	31

Table 2: The encoded frame position for the unconstrained VBR channel.

Sequence	Encoded frame positions
Salesman	20 22 24 26 28 30 32 34 36 38 40 42 44 46 48 50 52 54 57 60 63 66 69 72 74 76 78 80 82 85 88 91 94 97 100
Akiyo	18 20 22 24 26 28 30 32 34 36 38 40 42 44 46 48 50 52 54 57 60 63 66 68 70 72 74 76 79 82 85 88 91 94 97 100
Silent Voice	20 22 24 26 28 31 34 37 40 43 46 49 52 55 58 61 64 67 70 73 76 79 82 85 88 91 94 96 98 100

slightly better for the Akiyo sequence and worse for the Silent Voice sequence in comparison with our rate control scheme. This PSNR measure is however evaluated purely from the spatial domain. The proposed rate control scheme provides a higher temporal resolution than TMN8 with 2-frame skipping. Thus, in terms of subjective quality measure, the proposed variable frame rate scheme does not degrade motion smoothness severely and achieves a spatial/temporal quality trade off between TMN8 with 2-frame skipping and TMN8 with 1-frame skipping. As a result, it reduces the flickering effect, and the perceived quality is improved.

6.2 Time-varying CBR Channel

Next, we show the performance of frame rate control for time-varying CBR channels. The channel condition is modeled as follows. The channel bandwidth is a Gaussian-distributed random variable with a mean of 24 kbps and a standard deviation of 6 kbps for “Salesman” and “Silent Voice”, a mean of 48 kbps and a standard deviation of 12 kbps for “Foreman”, and a mean of 16 kbps and a standard deviation of 4 kbps for “Akiyo”. The reason for choosing different means and standard deviations is due to the nature of these test sequences. That is, larger mean and standard deviation values are needed for video with more complicated motion patterns. Salesman is widely used for slow motion video, Silent Voice is for moderate motion video, and Foreman is for relatively fast motion video. The duration time for a channel bandwidth to stay constant is another random variable with a uniform distribution between 10 and 40 frames. The channel bandwidth and duration time are generated by using the random number generator (in MATLAB) with the above statistical characteristics. Under these channel conditions, we seek a good trade-off between the spatial quality and the temporal quality to minimize the effect of time-varying channel bandwidth. Threshold values TH_{D1} and TH_{D2} in the rate control algorithm given in Section 4 are set to the MSE (mean squared error) of the I frame in the corresponding video.

Results for Salesman, Silent Voice, Foreman and Akiyo are shown in Figs. 7 - 10, respectively. A sample path for the time-varying CBR channel bandwidth is illustrated in part (a) for each figure. The rate control scheme is applied with respect to such a time-varying bandwidth condition. Under these channel conditions, the rate and PSNR plots as a function of the frame number are shown in Figs. 7 - 10 (b) and (c), respectively. We see from these figures that the proposed frame rate control can reduce the quality degradation while TMN8 does not work well. Especially, when the available channel bandwidth drops suddenly, the quality of TMN8 is degraded severely.

Statistical data of these curves are summarized in Tables 3 - 6. We see clearly that the proposed frame rate control algorithm under the time-varying CBR channel can improve the average PSNR by 0.3-0.9 dB and reduce the PSNR fluctuation by about 10 - 40% in comparison with TMN8.

Table 3: Performance comparison under time-varying CBR for the Salesman sequence.

Method	Avg PSNR	STD of PSNR	No of Enc. frms
TMN8 (CBR)	31.1162	0.9022	92
TMN8 (time-varying CBR)	31.2964	1.3144	92
Proposed method	31.8206	1.1448	77

Table 4: Performance comparison under time-varying CBR for the Silent Voice sequence.

Method	Avg PSNR	STD of PSNR	No of Enc. frms
TMN8 (CBR)	30.9601	0.4784	62
TMN8 (time-varying CBR)	30.8508	0.4918	62
Proposed method	31.0817	0.3908	56

Also, we can see the encoded frame position for time-varying CBR channels in Table 7. Note that the proposed frame rate control algorithm avoids the abrupt encoded frame interval change, which can degrade the perceived quality obviously. Our rate control scheme also provide a better visual quality than TMN8 in terms of subjective evaluation. Consequently, we can claim that the proposed frame rate control algorithm is more robust than TMN8 for the time-varying CBR channel.

7 Conclusion and Future Work

Rate control algorithms for low bit rate unconstrained VBR and time-varying CBR channels were proposed in this work. Time-varying CBR channel can be used to model various VBR channels such as the approximation of VBR, renegotiated CBR and feedback VBR. In our rate control scheme, we treat the encoding frame interval (or rate) as a control variable in order to pursue a good trade-off between spatial and temporal qualities. For the low-bit-rate unconstrained VBR channel, the proposed algorithm in Section 3 can improve human visual perceptual quality by providing a better trade-off between spatial and temporal qualities. For the time-varying CBR channel, the proposed algorithm in Section 4 can control the encoding frame interval to minimize the effect of the channel

Table 5: Performance comparison under time-varying CBR for the Foreman sequence.

Method	Avg PSNR	STD of PSNR	No of Enc. frms
TMN8(CBR)	30.5762	1.5377	65
TMN8(time-varying CBR)	30.3272	1.5518	65
Proposed method	31.2369	0.9250	54

Table 6: Performance comparison under time-varying CBR for the Akiyo sequence.

Method	Avg PSNR	STD of PSNR	NO of Enc. frms
TMN8(CBR)	34.0708	0.8025	89
TMN8(time-varying CBR)	34.0820	1.0844	89
Proposed method	34.3044	0.9504	82

Table 7: The encoded frame position for time-varying CBR channels.

Sequence	Encoded frame positions
Salesman	26 29 32 34 37 39 42 44 46 49 53 56 59 62 66 69 71 74 76 78 81 83 85 87 89 91 93 95 97 99 101 103 105 107 109 111 113 115 117 119 121 123 125 127 129 131 133 135 137 140 142 144 146 148 150 152 154 156 158 160 162 164 166 168 170 172 174 176 178 180 182 184 186 188 190 192 194 196 198 200
Silent Voice	18 22 25 28 31 34 37 40 43 46 49 52 55 58 61 64 67 70 74 78 82 87 92 96 100 103 106 109 112 115 118 121 124 127 130 133 136 139 142 145 148 151 154 157 160 164 168 172 176 180 184 188 192 196 200
Foreman	9 13 17 21 24 27 30 33 36 39 42 45 48 51 54 57 60 63 66 69 72 75 78 82 86 90 94 99 103 106 109 112 115 118 121 124 127 130 133 136 139 143 147 151 155 160 165 169 173 178 184 191 199

bandwidth change. Both algorithms can be employed for real-time video applications due to its negligible encoding time delay and computational overhead.

Even though the proposed rate control schemes give a satisfying performance, they have been developed based on observations and intuitive arguments. A more solid theoretical foundation is however needed. It is under our current investigation.

References

- [1] International Telecommunication Union, “Draft ITU-T H.263: Video coding for low bitrate communication,” draft international standard, ITU-T, July 1997.
- [2] T. V. Lakshman, A. Ortega, and A. R. Reibman, “VBR video: Tradeoffs and potentials,” *Proceeding of the IEEE*, Vol. 86, No. 5, May 1998, pp. 952–973.
- [3] J. J. Chen and D. W. Lin, “Optimal bit allocation for coding of video signals over ATM networks,” *IEEE Journal on Selected Areas in Communication*, Vol. 15, No. 6, August 1997, pp. 1002–1015.
- [4] C. Y. Hsu, A. Ortega, and A. R. Reibman, “Joint selection of source and channel rate for VBR video transmission under ATM policing constraints,” *IEEE Journal on Selected Areas in Communication*, Vol. 15, No. 6, August 1997.
- [5] A. R. Reibman and B. G. Haskell, “Constraints on variable bit-rate video for ATM networks,” *IEEE Trans. on Circuits and Systems for Video Technology*, Vol. 2, No. 4, December 1992, pp. 361–372.
- [6] D. Mukherjee and S. K. Mitra, “Combined mode selection and macroblock step adaptation for H.263 video encoder,” in: *Proc. of IEEE International Conference on Image Processing*, Vol. 2, October 1997, pp. 37–40.
- [7] J. Ribas-Corbera and S. Lei, “Rate control in DCT video coding for low-delay video communication,” *IEEE Trans. on Circuits and Systems for Video Technology*, submitted 1997.
- [8] T. Wiegand, M. Lightstone, D. Mukherjee, T. G. Campbell, and S. K. Mitra, “Rate-distortion optimized mode for very low bit rate video coding and emerging H.263 standard,” *IEEE Trans. on Circuits and Systems for Video Technology*, Vol. 6, No. 2, April 1996, pp. 182–190.
- [9] H. Song and C. C. J. Kuo, “Rate control for low bit rate video via variable frame rates and hybrid DCT/wavelet I-frame coding,” *IEEE Trans. on Circuits and Systems for Video Technology*, submitted 1998.
- [10] T. M. Cover and J. A. Thomas, *Elements of information theory*. John Wiley & Sons Inc., 1992.
- [11] T. Chiang and Y.-Q. Zhang, “A new rate control scheme using quadratic rate distortion model,” *IEEE Trans. on Circuits and Systems for Video Technology*, Vol. 7, No. 1, September 1997, pp. 246–250.
- [12] W. Ding and B. Liu, “Rate control of MPEG video coding and recording by rate-quantization modeling,” *IEEE Trans. on Circuits and Systems for Video Technology*, Vol. 6, February 1996, pp. 12–20.
- [13] L. J. Lin, A. Ortega, and C. C. J. Kuo, “Rate control using spline interpolated R-D characteristics,” in: *Proc. of SPIE Visual Communication and Image Processing*, April 1996, pp. 111–122.
- [14] K. H. Yang, A. Jacquin, and N. S. Jayant, “A normalized rate-distortion model for H.263-compatible codecs and its application to quantizer selection,” in: *Proc. of IEEE International Conference on Image Processing*, Vol. 2, October 1997, pp. 41–44.
- [15] Motion Picture Expert Group Video group, “MPEG-4 video verification model version 10.0,” draft in progress, ISO/IEC JTC1/SC29/WG11, February 1998.
- [16] A. Ortega, K. Ramchandran, and M. Vetterli, “Optimal trellis-based buffered compression and fast approximation,” *IEEE Trans. on Image Processing*, Vol. 3, January 1994, pp. 26–40.

- [17] K. Ramchandran, A. Ortega, and M. Vetterli, "Bit allocation for dependent quantization with application to multiresolution and MPEG video coder," *IEEE Trans. on Image Processing*, Vol. 3, No. 5, September 1994, pp. 533–545.
- [18] Y. Shoman and A. Gersho, "Efficient bit allocation for an arbitrary set of quantizer," *IEEE Trans. on Acoustic, Speech, and Signal Processing*, Vol. 36, September 1988, pp. 1445–1453.
- [19] W. Y. Lee and J. B. Ra, "A fast algorithm for optimal bit allocation," in: *Proc. of SPIE Visual Communication and Image Processing*, January 1997, pp. 167–175.
- [20] J. Choi and D. Park, "A stable feedback control of the buffer state using the controlled multiplier method," *IEEE Trans. on Image Processing*, Vol. 3, No. 5, September 1994, pp. 546–558.
- [21] D. W. Lin and J. J. Chen, "Efficient optimal rate-distortion coding of video sequences under multiple rate constraints," in: *Proc. of IEEE International Conference on Image Processing*, Vol. 2, October 1997, pp. 29–32.
- [22] J. Lee and B. W. Dickinson, "Temporally adaptive motion interpolation exploiting temporal masking in visual perception," *IEEE Trans. on Image Processing*, Vol. 3, No. 5, September 1994, pp. 513–526.
- [23] Image Processing Lab, UBC, "H.263+ encoder/decoder," tmn(h.263) codec, Feb 1998.
- [24] A. M. Tekalp, *Digital Video Processing*. Prentice-hall, 1995.

List of Tables

1	Performance comparison with TMN8 under unconstrained VBR. Target average rate is 24kbps.	15
2	The encoded frame position for the unconstrained VBR channel.	15
3	Performance comparison under time-varying CBR for the Salesman sequence.	17
4	Performance comparison under time-varying CBR for the Silent Voice sequence.	17
5	Performance comparison under time-varying CBR for the Foreman sequence.	17
6	Performance comparison under time-varying CBR for the Akiyo sequence.	18
7	The encoded frame position for time-varying CBR channels.	18

List of Figures

1	Illustration of the region partitioning for the adaptive control of the Lagrange multiplier.	23
2	Frame layer R-D modeling for the QCIF Salesman sequence: (a) the rate model and (b) the distortion model as a function of the average QP of macroblocks.	23
3	Overview of the framework of proposed rate control algorithms.	24
4	Performance comparison for the QCIF Salesman with a target average rate at 24 kbps: (a) the rate plot and (b) the PSNR plot as a function of the frame number.	24
5	Performance comparison for the QCIF Akiyo with a target average rate at 24 kbps: (a) the rate plot and (b) the PSNR plot as a function of the frame number.	25
6	Performance comparison for the QCIF Silent Voice with a target average rate at 24 kbps: (a) the rate plot and (b) the PSNR plot as a function of the frame number.	25
7	Performance comparison for the QCIF Salesman with a time-varying CBR channel: (a) the bandwidth variation plot, (b) the rate plot and (c) the PSNR plot as a function of the frame number.	26
8	Performance comparison for the QCIF Silent Voice with a time-varying CBR channel: (a) the bandwidth variation plot, (b) the rate plot and (c) the PSNR plot as a function of the frame number.	27
9	Performance comparison for the QCIF Foreman with a time-varying CBR channel: (a) the bandwidth variation plot, (b) the rate plot and (c) the PSNR plot as a function of the frame number.	28
10	Performance comparison for the QCIF Akiyo with a time-varying CBR channel: (a) the bandwidth variation plot, (b) the rate plot and (c) the PSNR plot as a function of the frame number.	29

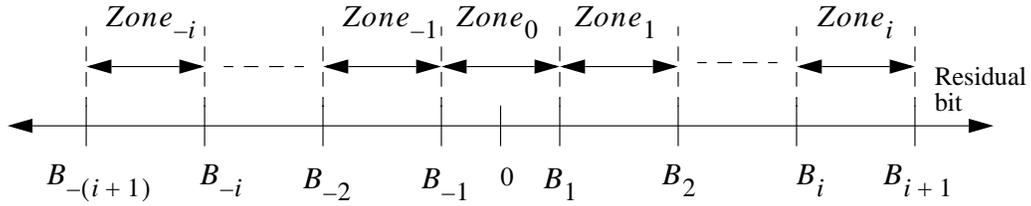


Figure 1: Illustration of the region partitioning for the adaptive control of the Lagrange multiplier.

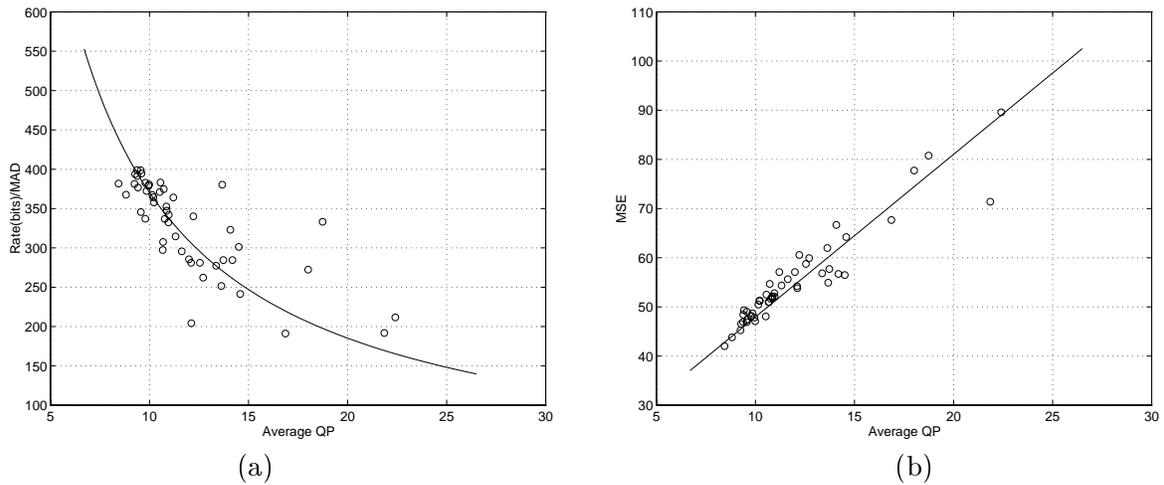
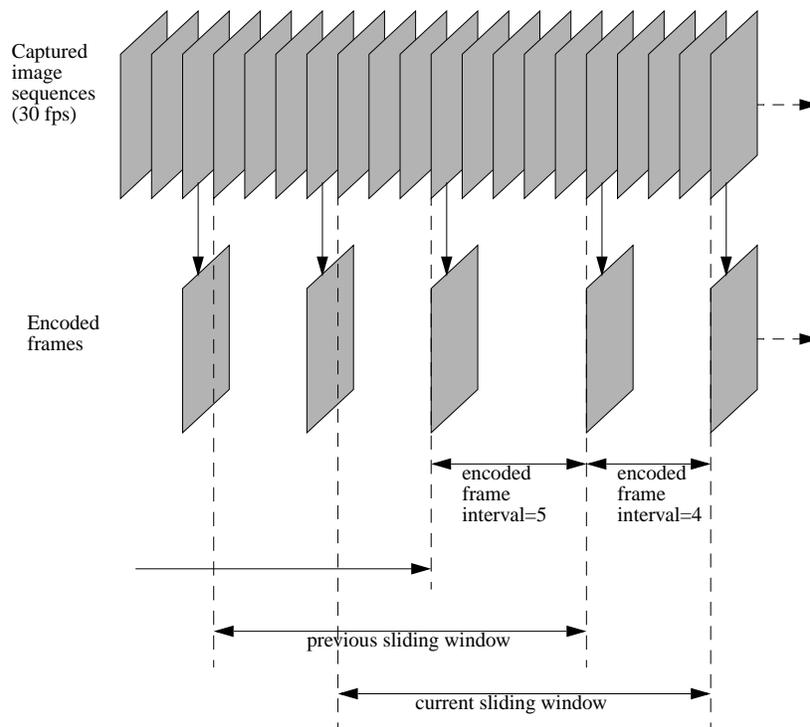


Figure 2: Frame layer R-D modeling for the QCIF Salesman sequence: (a) the rate model and (b) the distortion model as a function of the average QP of macroblocks.



The length of sliding window is 12 frames in the captured image sequences.

Figure 3: Overview of the framework of proposed rate control algorithms.

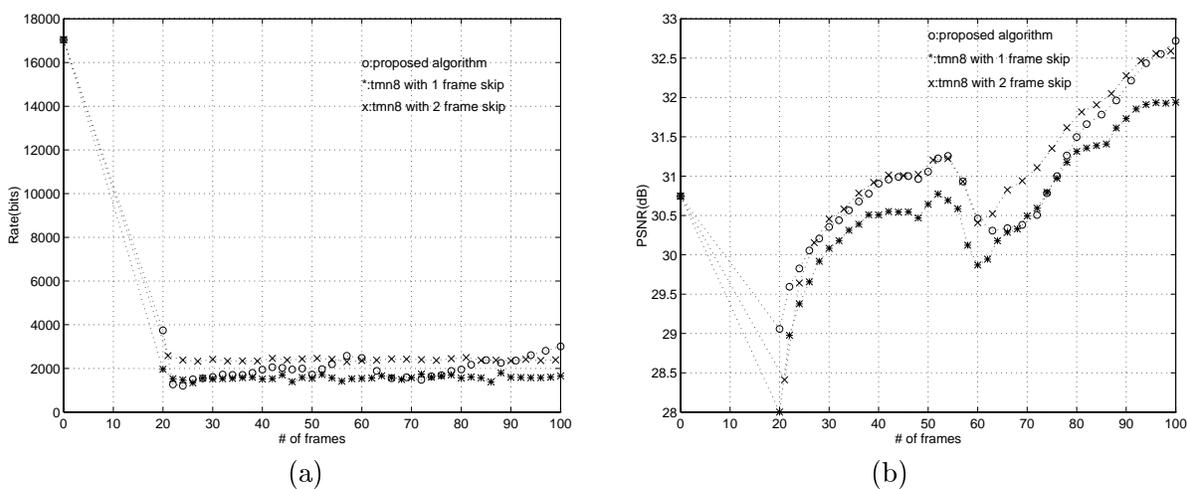
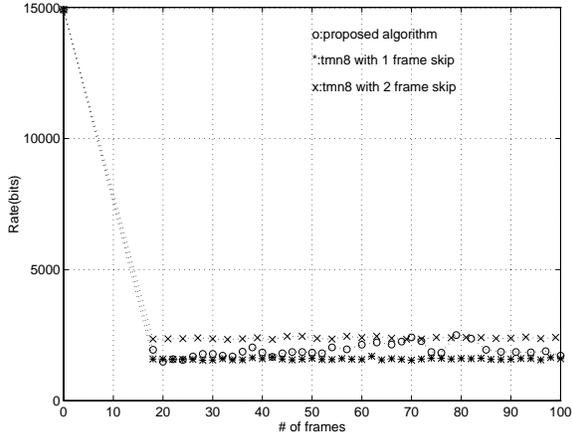
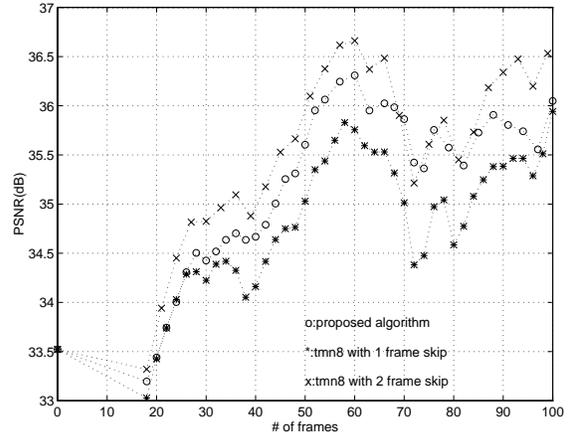


Figure 4: Performance comparison for the QCIF Salesman with a target average rate at 24 kbps: (a) the rate plot and (b) the PSNR plot as a function of the frame number.

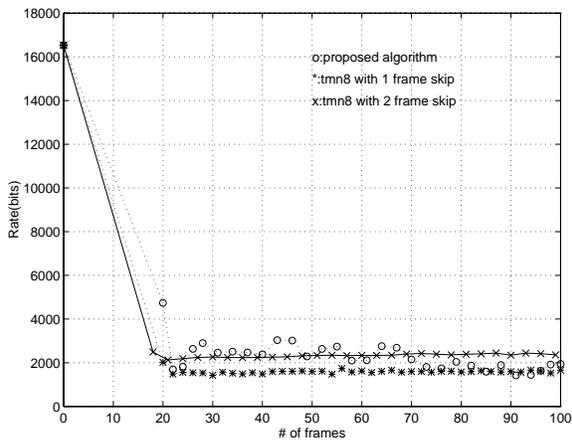


(a)

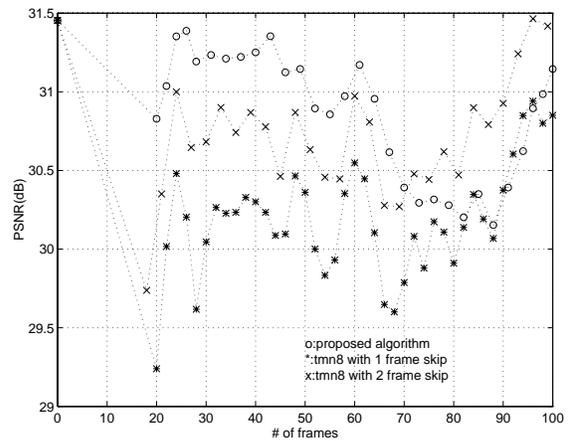


(b)

Figure 5: Performance comparison for the QCIF Akiyo with a target average rate at 24 kbps: (a) the rate plot and (b) the PSNR plot as a function of the frame number.

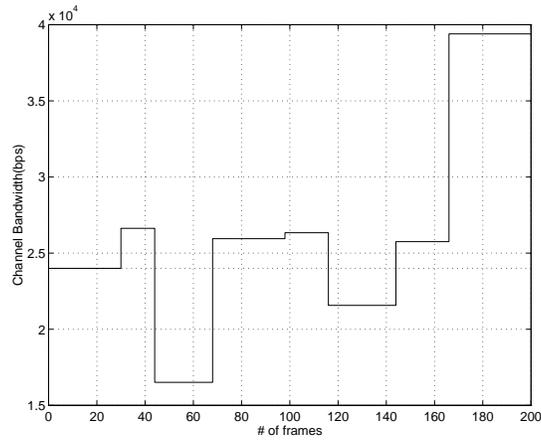


(a)

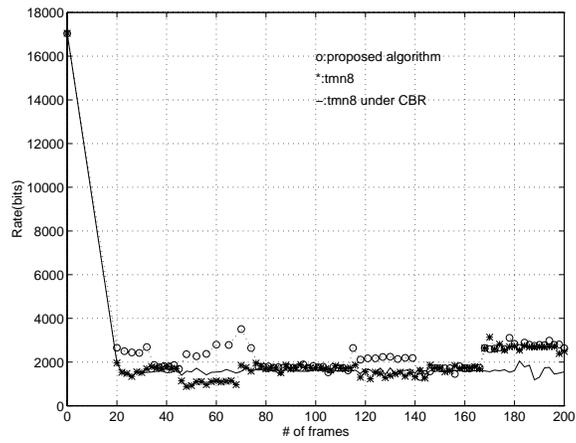


(b)

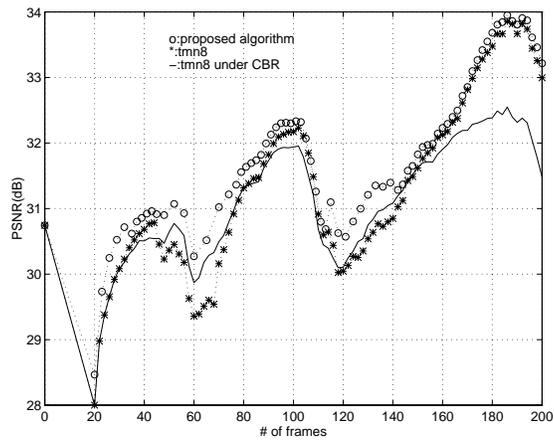
Figure 6: Performance comparison for the QCIF Silent Voice with a target average rate at 24 kbps: (a) the rate plot and (b) the PSNR plot as a function of the frame number.



(a)

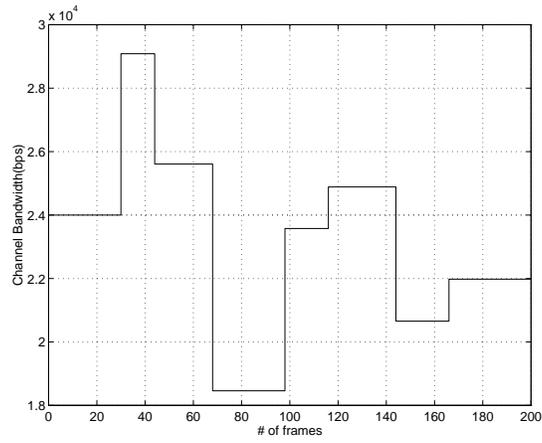


(b)

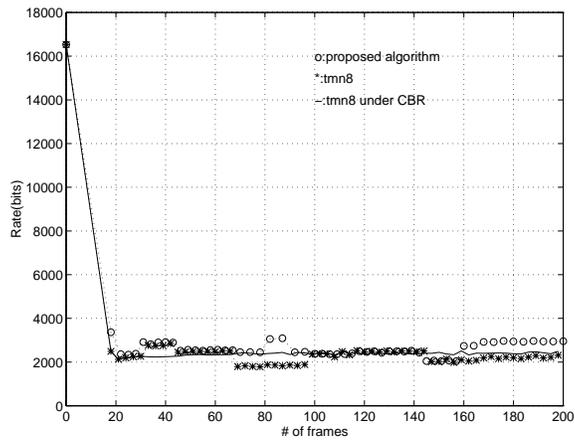


(c)

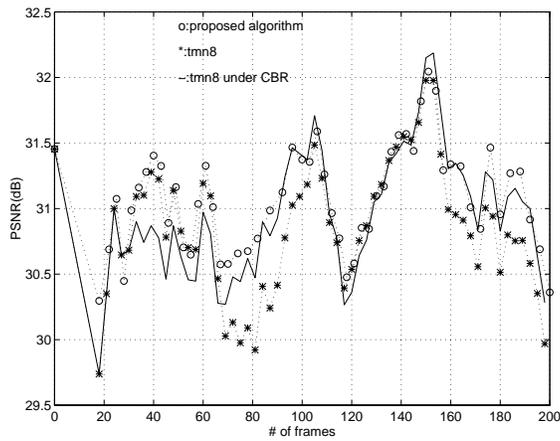
Figure 7: Performance comparison for the QCIF Salesman with a time-varying CBR channel: (a) the bandwidth variation plot, (b) the rate plot and (c) the PSNR plot as a function of the frame number.



(a)

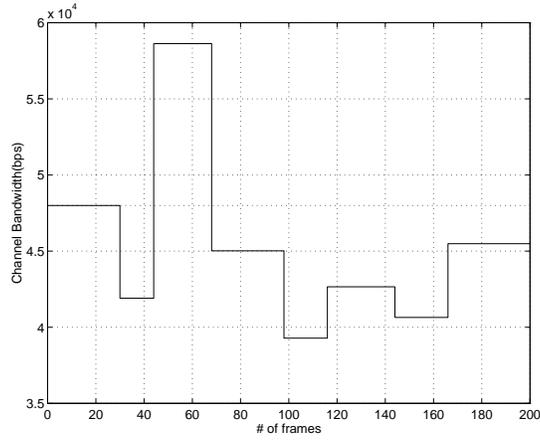


(b)

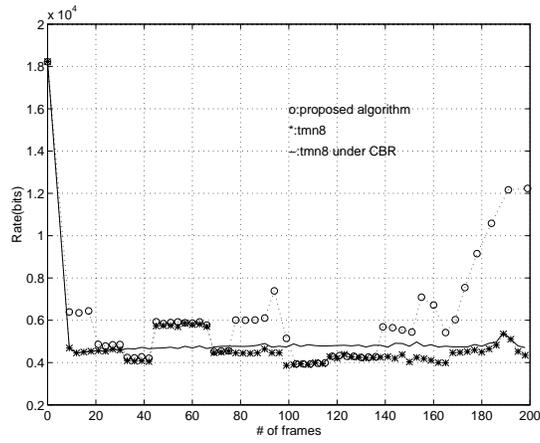


(c)

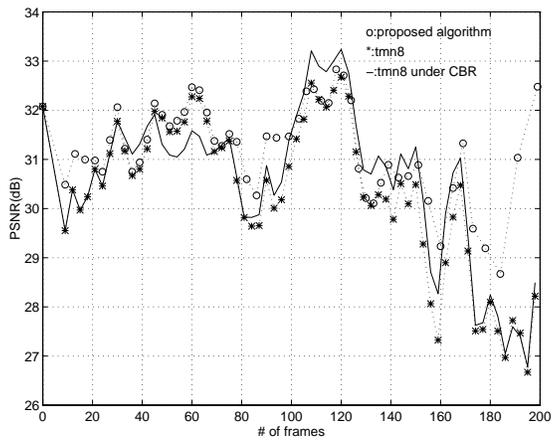
Figure 8: Performance comparison for the QCIF Silent Voice with a time-varying CBR channel: (a) the bandwidth variation plot, (b) the rate plot and (c) the PSNR plot as a function of the frame number.



(a)

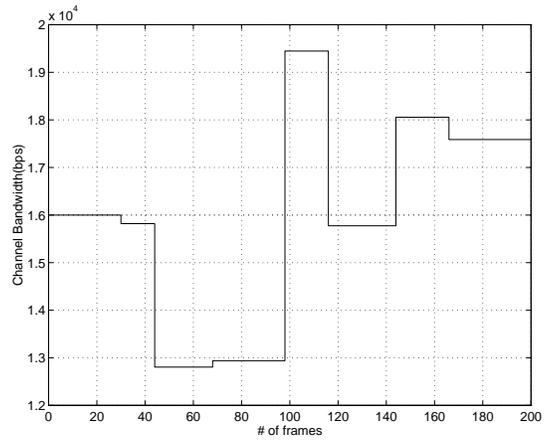


(b)

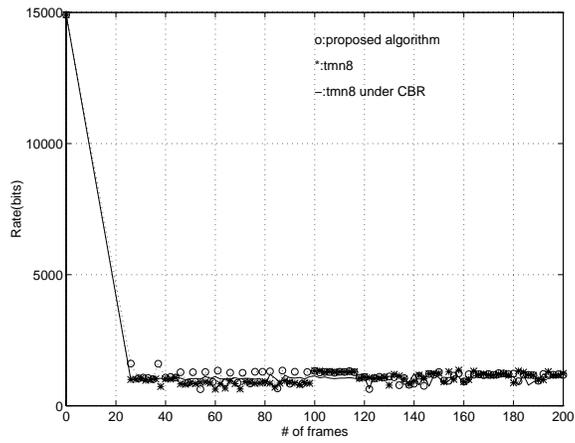


(c)

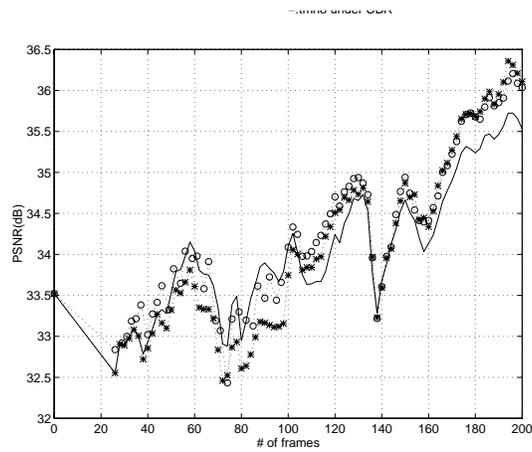
Figure 9: Performance comparison for the QCIF Foreman with a time-varying CBR channel: (a) the bandwidth variation plot, (b) the rate plot and (c) the PSNR plot as a function of the frame number.



(a)



(b)



(c)

Figure 10: Performance comparison for the QCIF Akiyo with a time-varying CBR channel: (a) the bandwidth variation plot, (b) the rate plot and (c) the PSNR plot as a function of the frame number.