MITSUBISHI ELECTRIC RESEARCH LABORATORIES
http://www.merl.com

Bit Allocation for MPEG-4 Video Coding with Spatio-Temporal Trade-offs

Lee, J.; Vetro, A.; Wang, Y.; Ho, Y.

TR2003-75 July 02, 2003

Abstract

This paper describes rate control algorithms that consider the trade-off between coded quality and temporal rate. We target improved coding efficiency for both frame-based and object-based video coding. We propose models that estimate the rate distortion characteristics for coded frames and objects, as well as skipped frames and objects. Based on the proposed models, we propose three types of rate control algorithms. The first is for frame-based coding, in which the distortion of coded frames is balanced with the distortion incurred by frame skipping. The second algorithm applies to object-based coding, where the temporal rate of all objects is constrained to be the same, but the bit allocation is performed at the object-level. The third algorithm also targets object-based coding, but in contrast to the second algorithm, the temporal rates of each object may vary. The algorithm also takes into account the composition problem, which may cause holes in the reconstructed scene when objects are encoded at different temporal rates. We propose a solution to this problem that is based on first detecting changes in the shape boundaries over time at the encoder, then employing a hole detection and recovery algorithm at the decoder. Overall, the proposed algorithms are able to achieve the target bit rate, effectively code frames and objects with different temporal rates, and maintain a stable buffer level.

Bit Allocation for MPEG-4 Video Coding with Spatio-Temporal Trade-offs

Jeong-Woo Lee, Anthony Vetro, *Member, IEEE*, Yao Wang, *Senior Member, IEEE* and Yo-Sung Ho, *Member, IEEE*

***Abstract*— This paper describes rate control algorithms that consider the trade-off between coded quality and temporal rate. We target improved coding efficiency for both frame-based and object-based video coding. We propose models that estimate the rate-distortion characteristics for coded frames and objects, as well as skipped frames and objects. Based on the proposed models, we propose three types of rate control algorithms. The first is for frame-based coding, in which the distortion of coded frames is balanced with the distortion incurred by frame skipping. The second algorithm applies to object-based coding, where the temporal rate of all objects is constrained to be the same, but the bit allocation is performed at the object-level. The third algorithm also targets object-based coding, but in contrast to the second algorithm, the temporal rates of each object may vary. The algorithm also takes into account the composition problem, which may cause holes in the reconstructed scene when objects are encoded at different temporal rates. We propose a solution to this problem that is based on first detecting changes in the shape boundaries over time at the encoder, then employing a hole detection and recovery algorithm at the decoder. Overall, the proposed algorithms are able to achieve the target bit rate, effectively code frames and objects with different temporal rates, and maintain a stable buffer level.**

***Index Terms*— Object-based Coding, MPEG-4, Rate Allocation, Rate Control, Rate-Distortion, Composition Problem**

I. Introduction

DURING the past decade, a number of video coding standards have been developed for communicating video data. These standards include MPEG-1 for CD-ROM storage [1], MPEG-2 for DVD and DTV application [2], H.261/H.263 for video conferencing [3], and MPEG-4 for object-based application [4]. In contrast to MPEG-2 [5], H.263 [6] and MPEG-4 [10] allow us to encode a video sequence with variable *frameskip*, where *frameskip* is the number frames that have been skipped. With this policy, the encoder may choose to skip frames to either satisfy buffer constraints or optimize the video coding process.

For the most part, frame skipping has only been employed to satisfy buffer constraints. In this case, the encoder is forced to drop frames since limitations on the bandwidth do not allow the buffer to drain fast enough. Consequently, bits that would be used to encode the next frame cannot be added to the buffer because they would cause the channel buffer to overflow. We should note that skipping frames could lead to poor reconstruction of the video since frames are skipped according to buffer occupancy and not according to content characteristics.

If frame skipping is allowed, the problem for the coding efficiency can be stated as follows. Given a video sequence, should the encoder choose to code more frames with lower spatial quality or fewer frames with higher spatial quality? It should be emphasized that the overall distortion of the coded sequence must also include in its calculations for the distortion of the skipped frames. This is a point that is often overlooked in other papers that report data with skipped frames.

The majority of the literature on rate-distortion (R-D) optimization does not touch on temporal aspect [6], [7], [8]. These papers consider mode decisions for motion and block coding [6], optimizations on the quantization parameter [7] and frame-type selection [8]. In these papers, however, it is assumed that the frame rate is fixed. Although the trade-off between spatial and temporal quality has been studied by Martins [9], the trade-off was achieved with a user selectable parameter.

In this paper, we consider three types of coding scenarios. The first is frame-based coding, in which the distortion of coded frames is balanced with the distortion incurred by frame skipping. The second applies to object-based coding, where the temporal rate of all objects is constrained to be the same, but the bit allocation is performed at the object-level. The third scenario also targets object-based coding, but in contrast to the second scenario, the temporal rates of each object may vary. In order to enable a trade-off between spatial and temporal quality to be made, we propose models that estimate the R-D characteristics for coded and skipped frames/objects. This is one of the primary contributions of this paper.

Given the proposed R-D models, we propose rate control algorithms to improve the coding efficiency and maintain stable buffer levels for the three types of coding scenarios described above. This is the second contribution of this paper. Improved coding efficiency is achieved through bit allocation to the frames or objects, and also by balancing the coded frame distortion with the distortion incurred by frames that have been skipped. It should be emphasized that object-based rate control algorithm have a great deal of flexibility since each object may be encoded at a different frame rate, and that past work in this area did not address this problem [10], [11].

One of the key difficulties in coding objects with different temporal rates is due to the composition problem, in which holes may appear in the reconstructed scene when objects

Manuscript received August 19, 2002; revised March 18, 2003. This paper was recommended by Associate Editor Huifang Sun.

Jeong-Woo Lee and Yo-Sung Ho are with the Department of Information and Communications, Kwangju Institute of Science and Technology, Kwangju, KOREA. E-mail:{jeongwoo,hoyo}@kjist.ac.kr

Anthony Vetro is with the Mitsubishi Electric Research Laboratories, Murray Hill, NJ 07974 USA.E-mail:avetro@merl.com

Yao Wang is with the Department of Electric Engineering, Polytechnic University, Brooklyn, NY 11201 USA. E-mail:yao@vision.poly.edu

Publisher Item Identifier

are encoded at different temporal rates [10]. This can easily be avoided if all the objects in the scene are constrained to the same temporal rate, i.e., a *fixed* temporal rate. However, to fully explore the potential coding gains that object-based coding offers, different temporal rates for each object, i.e., *variable* temporal rates, must be allowed. The third contribution of this paper is a proposed solution to this problem, in which changes in the shape boundaries over time are detected at the encoder, then hole detection and recovery algorithms are employed at the decoder. Although the idea of detecting the change in shape boundaries over time was presented in [17], rate control algorithms that effectively utilize this information for object-based coding were not described.

The rest of this paper is organized as follows. In Section II, we discuss the models that we use to estimate rate and distortion for coded and skipped frames/objects. In Section III, we propose a frame-based rate control algorithm that considers a variable temporal rate. In Section IV, we propose a object-based rate control algorithm with a constrained temporal rate that considers trade-off in the spatial and temporal quality for object-based coding. In Section V, we presents the general framework for object-based coding with variable frameskip, including rate and buffer constraints that apply for arbitrary frameskip among objects and propose an object-based rate control algorithm for this new framework. Also in this section, we briefly discuss the composition problem and propose a solution to overcome it. Simulation results to evaluate the proposed algorithms are presented in Section VI, and some conclusions are drawn in Section VII.

II. Rate-Distortion Models

The purpose of this section is to introduce models that estimate rate and distortion characteristics of video for coders that use variable frameskip. In order to model the distortion of skipped frames, we assume that the skipped frames are interpolated from previous coded frame. While most of the formulation is provided for video frames, it should be noted that these models can also be applied to object data as well.

A. Rate Modeling

The relationship between rate and distortion for texture coding has been given a considerable amount of attention for rate control applications. For example, in [19], a model is derived from classic R-D theory and then modified to match the encoding process of practical encoders. In [20], a generic rate-quantizer model was proposed which can be adapted according to changes in picture activity. In [15], a rate control scheme using a quadratic rate-quantizer (R-Q) model was presented. Most recently, He and Mitra have proposed a ρ-domain model that estimates the rate based on the percentage of zeros among quantized coefficients [21]. All of these models can provide a relationship between the rate and quantizer for a given set of data. In the results presented later on, the quadratic R-Q model described in [15] is used. Since this model has been extensively tested and used in other related works, and is not very central to this paper, it is not reviewed here. On the other hand, the skipping of frames is a focal point of this paper, so it is worthwhile to formally introduce the frameskip parameter and briefly discuss its impact on the rate.

Given the R-Q relationship for a single frame, the average bit-rate over time, $\overline{R}$, can be expressed as,

$$\overline{R} = \sum_{k=i}^{i+\overline{F}} R(t_k) \cong \overline{F} \cdot \overline{R}(t_k) \tag{1}$$

where t_k is the time index, $\overline{F}$ is the average number of coded frames per second and $\overline{R}(t_k)$ is the average number of bits per coded frame. In our experiments, we have confirmed that $\overline{R}(t_k)$ increases as $\overline{F}$ decreases.

The parameter that will tie the formulations for the rate and distortion together is the frameskip parameter, f_s, which represents the distance between two coded frames at a given time instant, i.e., $f_s - 1$ represents the number of frames that have been skipped. This parameter may change at each coding instant, therefore the relation between this parameter and the average coded frame rate, $\overline{F}$, is defined by the average frameskip parameter, $\overline{f}_s$, and is given by,

$$\overline{f}_s = \frac{F_{src}}{\overline{F}} \tag{2}$$

where F_{src} is the source frame rate. To be clear, f_s is a parameter that will be used to quantify the distortion due to frame skipping. In turn, this affects the values of $\overline{f}_s$ and $\overline{F}$ and ultimately ties back to the average bit-rate, $\overline{R}$.

B. Distortion Modeling

We consider a general formulation for distortion that accounts for skipped frames, $\bar{D}(Q, f_s)$, where Q represents the quantization parameter used for coding and f_s represents the frameskip parameter. Furthermore, we denote the average distortion for coded frames by $\bar{D}_c(Q)$ and the average distortion for skipped frames by $\bar{D}_s(Q, f_s)$. The coded distortion is dependent on the quantizer, while the temporal distortion depends on both the quantizer and the amount of frameskip. Although the frameskip factor does not directly influence the coded distortion of a particular frame, it does indeed impact this part of the distortion indirectly. For example, the amount of frameskip will influence the residual component, and secondly, it will also have an impact on the quantizer that is chosen. Therefore, the quantization parameter should be represented in terms of the previously coded time and f_s. It is important to note that the distortion for skipped frames has a direct dependency on the quantization parameter in the coded frames. The reason is that the skipped frames are interpolated from the coded frames, thereby carrying the same spatial quality, in addition to the temporal distortion.

Given the above, we consider the average distortion over the specific time interval $\lceil t_i, t_{i+f_s}]$, which is given by,

$$\bar{D}_{\lceil t_i, t_{i+f_s}]}(Q_{i+f_s}, f_s) = \frac{1}{f_s}\left[D_c(Q_{i+f_s}) + \sum_{k=i+1}^{i+f_s-1} D_s(Q_i, k)\right] \tag{3}$$

In the above, the distortion over the specified time interval is due to the spatial distortion of 1 coded frame at $t = t_{i+f_s}$

plus the temporal distortion of $f_s - 1$ skipped frames, which is dependent on the quantizer for the previously coded frame at $t = t_i$.

1) Distortion for Coded Frames: From classic rate-distortion modeling [13], it is well-known that the variance of the quantization error is given by,

$$\sigma_q^2 = a \cdot 2^{-2R} \cdot \sigma_z^2 \tag{4}$$

where σ_z^2 is the input signal variance, R is the average rate per sample and a is a constant that is dependent on the *pdf* of the input signal and quantizer characteristics. We use the above equation to model the spatial distortion for coded frame at $t = t_i$,

$$D_c(Q_i) = a \cdot 2^{-2R(t_i)} \cdot \sigma_{z_i}^2 \tag{5}$$

where i is the coded time index, Q_i is the quantization parameter for coded frame at $t = t_i$, and $R(t_i)$ is the average rate per sample at $t = t_i$. The above model is valid for a wide array of quantizers and signal characteristics.

As stated earlier, we have found that the average bits/frame increases for larger values of f_s, which implies that the amount of frameskip impacts the statistics of the residual. However, contrary to our intuition, we have found that the variance remains almost the same regardless of the motion characteristics within a scene. This indicates that the variance is not capable of reflecting small differences in the residual that impact the actual relation between rate and distortion. The explanation for this phenomenon may be explained by the the design of the particular coding scheme and is believed to happen because of the presence of high-frequency coefficients. Actually, we believe that it is not only their presence, but also the position of such coefficients. If certain run-lengths are not present in the VLC table, less efficient escape coding techniques must be used. This probably means that f_s affects the pdf of the residual, i.e., the value of a, while not changing $\sigma_{z_i}^2$ much. We feel that the ρ-domain analysis in [21] provides a good framework to study this further.

2) Distortion for Skipped Frame: To model the temporal distortion due to skipped frames, we assume, without loss of generality, that a temporal interpolator simply repeats the previously coded frame. Other interpolators that average past and future reference frames, or make predictions based on motion, may still be considered in this framework.

The distortion due to skipped frames can be broken into two parts: one due to the coding of the reference and another due to the interpolation error. This can be derived as follows. Let $\hat{\psi_k}$ denote the estimated frame at $t = t_k$, $\breve{\psi}_i$ denote the last coded frame at $t_i < t_k$, and $\hat{\psi_k} = \breve{\psi}_i$. Then, the estimation error at $t = t_k$ is,

$$e_k = \psi_k - \hat{\psi_k} = \psi_k - \breve{\psi}_i = \underbrace{\psi_k - \psi_i}_{\Delta z_{i,k}} + \underbrace{\psi_i - \breve{\psi}_i}_{\Delta c_i} \tag{6}$$

where ψ_i and ψ_k represent the original frame at $t = t_i$ and $t = t_k$, respectively. $\Delta z_{i,k}$ and Δc_i represent the frame interpolation error and coding error, respectively. Assuming these quantities are independent, the MSE is given by,

$$E\{e_k^2\} = E\{\Delta^2 c_i\} + E\{\Delta^2 z_{i,k}\} \tag{7}$$

which is equivalently expressed as,

$$D_s(Q_i, k) = D_c(Q_i) + E\{\Delta^2 z_{i,k}\} \tag{8}$$

In order to derive the expected MSE due to frame interpolation, we first assume that the frame at $t = t_k$ is related to the frame at $t = t_i$ with motion vectors $(\Delta x(x,y), \Delta y(x,y))$,

$$\psi_k(x,y) = \psi_i(x + \Delta x(x,y), y + \Delta y(x,y)) \tag{9}$$

In the above, it is assumed that every pixel (x,y) has a motion vector associated with it. In practice, we use block motion vectors to approximate the motion at every pixel inside the block. By expanding (9) in a Taylor series, we obtain the following equation.

$$\begin{aligned}\psi_i(x + \Delta x(x,y), y + \Delta y(x,y)) &= \\ \psi_i(x,y) + \frac{\partial \psi_i}{\partial x}\Delta x_{i,k} &+ \frac{\partial \psi_i}{\partial y}\Delta y_{i,k} + \cdots\end{aligned} \tag{10}$$

where $(\frac{\partial \psi_i}{\partial x}, \frac{\partial \psi_i}{\partial y})$ represent the spatial gradients in the x and y directions. Then,

$$\Delta z_{i,k} = \frac{\partial \psi_i}{\partial x}\Delta x_{i,k} + \frac{\partial \psi_i}{\partial y}\Delta y_{i,k} \tag{11}$$

It should be noted that the above has been expanded using a first-order Taylor expansion and is valid for small $(\Delta x, \Delta y)$. This is equivalent to the optical flow equation, where the same condition on motion is also true. As a result, (11) is less accurate for sequences with large motion. However, for these sequences, the accuracy of the estimation is not so critical since an optimized encoder would not choose to lower the frame rate for such sequences anyway.

Treating the spatial gradients and motion vectors as random variables and assuming the motion vectors and spatial gradients are independent and zero-mean, we have,

$$E\{\Delta^2 z_{i,k}\} = \sigma_{x_i}^2 \sigma_{\Delta x_{i,k}}^2 + \sigma_{y_i}^2 \sigma_{\Delta y_{i,k}}^2 \tag{12}$$

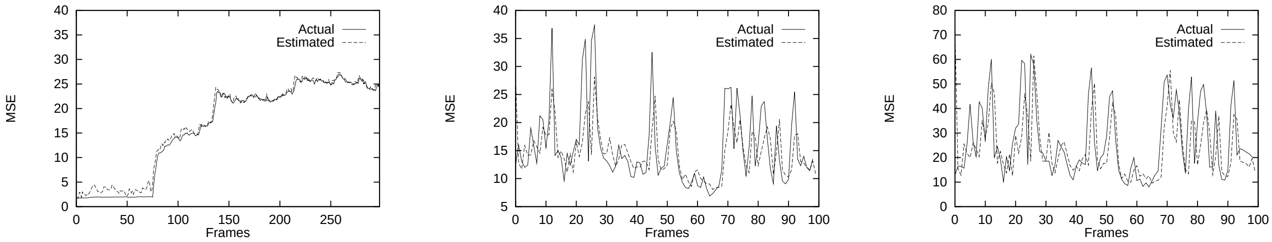
where $(\sigma_{x_i}^2, \sigma_{y_i}^2)$ represent the variances for the x and y spatial gradients in frame at $t = t_i$, and $(\sigma_{\Delta x_{i,k}}^2, \sigma_{\Delta y_{i,k}}^2)$ represent the variances for the motion vectors in the x and y directions, respectively.

3) Practical Considerations: The main practical aspect to consider is how the equations for the distortion of skipped frames are evaluated based on current and past data [14]. For instance, in its current form, (12) assumes that the motion between i, the current time instant, and k, a future time instant is known. However, this would imply that motion estimation is performed for each candidate frame, k. Since such computations are not practical, it is reasonable to assume linear motion between frames and approximate the variance of motion vectors by,

$$\sigma_{\Delta_{i,k}}^2 \approx \sigma_{\Delta_{i-f_l,i}}^2 \cdot \left(\frac{k-i}{f_l}\right)^2 \tag{13}$$

where f_l denotes the amount of frameskip between the last coded frame and its reference.

Similarly, estimates of the distortion for the next candidate frame to be coded (i.e., calculation of (4)) requires knowledge of a and $\sigma_{z_i}^2$, which depends on f_s. Since we do not want



(a)

(b)

(c)

Fig. 1. Comparison of Actual vs. Estimated Distortion for AKIYO. (a) Distortion for Coded Frames (b) Distortion for Skipped Frames : First Skipped Frames Only (c) Distortion for Skipped Frames : Second Skipped Frames Only.

to perform motion estimation for the entire set of candidate frames, the actual residuals are not available either. To overcome this, the residual for the set of candidate frames at a given time instant is predicted based on the residual of the current frame at $t = t_i$. Since we have observed that changes in the variance for different frameskip are very small, this approximation is expected to have a minimal impact on the estimated distortion. In this way, changes in D_c are only affected by the bit budget for candidate frameskip factors.

4) Accuracy of Distortion Models: To confirm the accuracy of the distortion models, we devise two sets of experiments. A first experiment to test the accuracy of the estimated distortion for coded frames and a second experiment to test the accuracy of the estimated distortion for skipped frames. Results are provided for the AKIYO sequence, as this is one such sequence in which the assumptions for estimating the distortion for skipped frames hold.

In the first experiment, the sequence is coded at a full frame-rate of 30fps. Three fixed quantizers are used to code the sequence, $Q = 2$ for the first 100 frames, $Q = 15$ for the next 100 frames and $Q = 30$ for the last 100 frames. Fig. 1(a) shows the estimated distortion for the coded frames, which are calculated according to (5) with $a = 1$. From this plot, we note that the estimated distortion tracks the actual distortion quite accurately. Similar results have been obtained for sequences with higher motion [23].

In the second experiment, the sequence is coded at a fixed rate of 10fps. Fig. 1(b) and Fig. 1(c) show a comparison of the actual and estimated distortion for the skipped frames. Fig. 1(b) shows the first skipped frames, while Fig. 1(c) shows the second skipped frames. The plots indicate that the estimated distortion of skipped frames is quite accurate. In case the motion in the sequence is large, the distortion due to frame skipping would be a significant factor in the overall distortion. In this case, we do not expect the estimation to be accurate due to the assumptions made in the derivation of this model. However, it is safe to say that an optimized coder will never choose to skip frames for such a sequence - we only need to know that the distortion is high. Rather, it will resort to using a coarser quantizer until buffer constraints force the encoded sequence to a lower frame-rate. As a result, the accuracy for estimating the distortion of the skipped frames is much more critical for sequences with low to moderate motion.

III. Frame-based Rate Control

Based on the R-D models developed in the previous section, we describe a frame-based rate control algorithm that accounts for frameskip.

A. Algorithm Overview

Our objective is to minimize the average distortion given by (3) subject to constraints on the overall bit rate and buffer occupancy. Stated formally,

$$\arg\min \bar{D}_{[t_i, t_{i+f_s}]}(Q_{i+f_s}, f_s) \tag{14}$$
$$\text{subject to} \begin{cases} \bar{R} \leq R \\ B_i + R(t_{i+f_s}) < B_{\max} \\ B_i + R(t_{i+f_s}) - f_s \cdot R_{drain} > 0 \end{cases}$$

where $\bar{R}$ is the number of generated bit rate, R is the target bit rate, $B_{\max}$ is the maximum buffer size in bits, B_i is the current buffer level, also in bits, and R_{drain} is the rate at which the buffer drains per frame.

In order to solve the above problem, we use the rate control algorithm outlined in Fig. 2. The parameter f_l denotes the amount of frameskip between the last coded frame and its reference. It should be noted that the parameter δ is used to limit the change in f_s from one coded frame to another, similar to the usual bounding of the quantization parameter.

B. Bit Allocation and Buffer Control

Given a candidate value of f_s, target bits for the frame are dependent on this value of f_s and the buffer occupancy, B_i.

```
1  Set $f_s = \max\{1, f_l - \delta\}$, $D_{min} = \infty$.
2  Calculate the target bits for the frame, which is mainly depen-
   dent on the current value of $f_s$ and $B_i$.
3  Determine the quantizer value, $Q_{i+f_s}$ using R-Q model.
4  Estimate the distortion using (3).
5  Check if the quantizer and the rate that it expects to produce
   still satisfies rate and buffer constraints. If not, skip to last
   step since the current value of $f_s$ is no longer valid.
   Otherwise, continue.
6  If the current distortion is less than $D_{min}$, then replace $D_{min}$
   with the current distortion and record encoding parameters.
7  Repeat from step 2 with $f_s = f_s + 1$ while new value of $f_s$,
   $f_s \leq \min\{f_l + \delta, f_{max}\}$
```

Fig. 2. Rate Control Algorithm for Frame-based Coding.

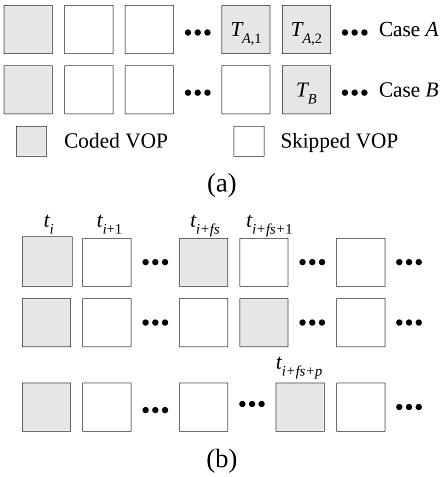


Fig. 3. Coding Modes. (a) Model I (b) Model II

As in [10], initial target bits, T_V, are determined according to the number of remaining bits, the number of frames left in the sequence and the number of bits spent during the last frame. The only difference with the initial calculation is that the remaining number of frames are divided by the candidate value of f_s. In this way, a proportionately higher number of bits will be assigned when the frameskip is higher. After the initial target bits, T_v, has been determined, it is scaled by,

$$T_B = T_V \cdot \frac{\tilde{B}_i + 2(B_{\max} - \tilde{B}_i)}{2\tilde{B}_i + (B_{\max} - \tilde{B}_i)} \tag{15}$$

where the modified buffer fullness, $\tilde{B}_i$, accounts for the current value of frameskip and is expressed as,

$$\tilde{B}_i = B_i - (f_s - 1) \cdot R_{drain} \tag{16}$$

This modification to the traditional buffer fullness must be made to simulate the lower occupancy level as a result of frame skipping. Otherwise, the scaling operation in (15) would force the target too low. If the target is too low for higher f_s values, the resulting quantizer would not be able to differentiate itself from the quantizers that were computed at lower f_s values. In this case, it would be difficult for the trade-off between coded and temporal distortion in (3) to ever be in favor of skipping frames.

IV. Object-Based Rate Control: Constrained Temporal Rates

In this section, we propose an object-based rate control algorithm that supports the coding of multiple objects in a scene, while improving the coding efficiency. The temporal rate of objects is allowed to vary, but we impose the constraint that the temporal rates of all objects must be equal. In the following, we first introduce the different coding modes that are considered, then provide an overview of the algorithm, followed by a discussion of bit allocation and buffer issues.

A. R-D Coding Modes

Assuming that a frame at $t = t_i$ is the last coded frame, the R-D coding modes determine the manner in which the following frames are coded. Given a target number of bits, Fig. 3 shows the possible coding modes for considering the trade-off between spatial and temporal quality, where Fig. 3(a) considers trade-off without regard to the current quantizer value and Fig. 3(b) considers trade-off when the current quantizer is too large.

Instead of coding frames with a fixed frameskip rate, we consider coding them with a variable frameskip over a specific time interval $[t_i, t_{i+f_s+1}]$. The value of f_s is to be determined and indicates that $(f_s - 1)$ frames should be skipped due to a lack of target bits to code the texture in the next frame. The number of bits used to code the shape, motion and header information of the previous frame is used to determine if there are sufficient number of bits or not.

Given rate and buffer constraints, we present the encoder with two options as part of Model I:

1) The encoder codes the two successive frames. [Case A]
2) The encoder skips the current frame and codes the next frame with the higher target bits allocated by skipping the current frame. [Case B]

The above choices are made continuously on a frame-by-frame basis and are based on the R-D characteristics of the frames as described further below.

Given the proposed R-D models, we consider the distortion over the specific time interval $[t_i, t_{i+f_s+1}]$ for each case. For Case A, the distortion of the jth object is defined as,

$$\bar{D}^j_{I,A}(Q_j, f_s) = D_c(Q_{j,i+f_s}) + D_c(Q_{j,i+f_s+1}) \tag{17}$$

For Case B,

$$\bar{D}^j_{I,B}(Q_j, f_s) = D_s(Q_{j,i}, i + f_s) + D_c(Q_{j,i+f_s+1}) \tag{18}$$

In the above, we ignore the temporal distortion of $(f_s - 1)$ skipped objects. If the target bits are sufficient for the current frame, as is the case for high bit-rates, this distortion will disappear. Although both cases have the same expression for the coded distortion (third term), the coded distortion in (18) will usually be smaller that the coded distortion in (17) since more target bits are allocated to the latter case. In the event that the target bits, $T_{A,2}$, for the second frame is smaller than the shape, motion and header information, $T_{hdr}(t = t_i)$, of the previously coded frame, the frame at $t = t_{i+f_s+1}$ is skipped in Case A. As a result, the distortion is given by,

$$\bar{D}^j_{I,A}(Q_j, f_s) = D_c(Q_{j,i+f_s}) + D_s(Q_{j,i+f_s}, i + f_s + 1) \tag{19}$$

As stated earlier, since every object in the current time is either coded or all objects are skipped, the distortion over all object are defined by,

$$d_{I,A}(Q, f_s) = \sum_{j=1}^{M} \bar{D}^j_{I,A}(Q_j, f_s) \tag{20}$$

$$d_{I,B}(Q, f_s + 1) = \sum_{j=1}^{M} \bar{D}^j_{I,B}(Q_j, f_s) \tag{21}$$

where M denotes the total number of video objects.

With regard to coding efficiency, we consider the case in which the current quantizer for the allocated target bits is too large. In this case, the encoder should decide whether to code the current frame with this quantizer or skip several frames and assign a lower quantizer to the next coded frame as shown in Fig. 3(b). In typical coding simulations, a large quantizer will be assigned to a frame at time index $(i + f_s)$ due to a high buffer level leading to an insufficient number of available bits. Although the buffer occupancy is decreased with frames up to time index $(i + f_s)$ skipped, it should be noted that the number of available bits is not sufficient to allocate a low quantizer to a coded frame because f_s is only determined by the number of bits for the header information of the previous frame. As a result, the coded frame with low quality is likely to affect the distortion of several frames to follow. Let $f_{\max}$ be the last coded time index that the coded frame affects and p the number of skipped frames for assigning a lower quantizer. Given the above, we consider the distortion over the specific time interval $[t_i, t_{i+f_s+p}]$. If we ignore the distortion occurred by a lack of the target bits, we can quantify the distortion of Model II as follows:

$$\bar{D}^j_{II}(Q_{j,i+f_s+p}, f_s+p) = \sum_{k=i+f_s}^{i+f_s+p-1} D_s(Q_{j,i}, k) + D_c(Q_{j,i+f_s+p}) + \sum_{k=i+f_s+p+1}^{i+f_{\max}} D_s(Q_{j,i+f_s+p}, k) \quad (22)$$

Finally, the distortion over all objects for this case is defined as,

$$d_{II}(Q, f_s+p) = \sum_{j=1}^{M} \bar{D}^j_{II}(Q_{j,i+f_s+p}, f_s+p) \quad (23)$$

B. Algorithm Overview

Given the above coding modes, we now consider a rate control algorithm for object-coding that maximizes the performance subject to constraints on the overall bit rate and buffer occupancy.

For Model I, the coding mode is determined by,

$$\min\{d_{I,A}(Q, f_s), d_{I,B}(Q, f_s+1)\} \quad (24)$$

where the constraint on the rate is written as,

$$\bar{R} \leq R \quad (25)$$

The constraints on buffer are given by,

$$\begin{aligned}
&\mathbf{C}_{\mathbf{A},\mathbf{1}}\equiv\begin{cases} B_i + R(t_{i+f_s}) < B_{\max} \\ B_i + R(t_{i+f_s}) - f_s \cdot R_{\text{drain}} > 0 \end{cases}\\
&\mathbf{C}_{\mathbf{A},\mathbf{2}}\equiv\begin{cases} B_{i+f_s} + R(t_{i+f_s+1}) < B_{\max} \\ B_{i+f_s} + R(t_{i+f_s+1}) - (f_s+1) \cdot R_{\text{drain}} > 0 \end{cases}\\
&\mathbf{C}_{\mathbf{B},\mathbf{1}}\equiv\begin{cases} B_i < B_{\max} \\ B_i - f_s \cdot R_{\text{drain}} > 0 \end{cases}\\
&\mathbf{C}_{\mathbf{B},\mathbf{2}}\equiv\begin{cases} B_i + R(t_{i+f_s+1}) < B_{\max} \\ B_i + R(t_{i+f_s+1}) - (f_s+1) \cdot R_{\text{drain}} > 0 \end{cases}
\end{aligned} \quad (26)$$

By considering the constraints on the rate and buffer, we select Case *A* or Case *B* with a minimum distortion as a coding mode.

For Model II, the coding mode is determined by,

$$\arg\min d_{II}(Q, f_s+p) \quad (27)$$

where the constraint on the rate follows (25). The constraints on buffer are given by,

$$\begin{aligned}
&\mathbf{C}_{\mathbf{II},\mathbf{p-1}}\equiv\begin{cases} B_i < B_{\max} \\ B_i - (f_s+p-1) \cdot R_{\text{drain}} > 0 \end{cases}\\
&\mathbf{C}_{\mathbf{II},\mathbf{p}}\equiv\begin{cases} B_i + R(t_{i+f_s+p}) < B_{\max} \\ B_i + R(t_{i+f_s+p}) - (f_s+p) \cdot R_{\text{drain}} > 0 \end{cases}
\end{aligned} \quad (28)$$

With the constraints on the rate and buffer, we determine the frameskip value which is the sum of f_s and p.

In order to control the rate and select the proper coding mode, we use the algorithm shown in Fig. 4. It should be noted that the process for Model II is performed when the current quantizer for the allocated target bits is too large.

C. Bit Allocation and Buffer Control

With regards to bit allocation, given the remaining number of frames, N'_r, the calculation of the target bits for the frame follows the procedures discussed in Section III-B. The remaining number of frames, N'_r, and the buffer fullness, $\tilde{B}_i$, are determined according to the current mode.

For Case A in Model I, they are changed as follows:

$$\begin{aligned} N'_r &= N_r - f_s \\ \tilde{B}_i &= B_i + R(t_{i+f_s}) - f_s \cdot R_{drain} \end{aligned} \quad (29)$$

For Case B in Model I,

$$\begin{aligned} N'_r &= (N_r - f_s + 1)/2 \\ \tilde{B}_i &= B_i - f_s \cdot R_{drain} \end{aligned} \quad (30)$$

For Model II,

$$\begin{aligned} N'_r &= (N_r - f_s + 1)/p \\ \tilde{B}_i &= B_i - (f_s+p-1) \cdot R_{drain} \end{aligned} \quad (31)$$

where N_r and B_i are the remaining number of frames and the buffer fullness at time t_i, respectively.

In order to find appropriate quantizer values for every object in the scene, we need to distribute the total target bits for

```
1  Set $f_s = 1$.
2  Calculate the target bits for the frame. See Sec. III-B.
3  Determine whether the current frame should be skipped or not.
   If the target bits may not be enough even to code the motion,
   shape and header information, increase the value of $f_s$ until
   the target is larger than information used in the previous
   frame. If the frame is skipped, repeat from Step 2.
4  Distribute the target bits according to (32) for each object.
5  Determine the current quantizer and the target bits for each
   object subject to constraints on the previous quantizer.
6  Estimate the distortion using (20) and (21). To estimate the
   distortion, we need to calculate the target bits for the next
   coded VOP as well.
7  Check if the selected quantizer and the rate still satisfy the
   rate and buffer constraints. If not, repeat from Step 2 with
   $N_r = N_r - f_s$. Otherwise, continue to Step 8.
8  Select the proper coding mode for Model I. See Sec. IV-A
9  Select the proper coding mode for Model II using (27) and
   (28). It is important to note that we should have a standard
   of judgments for determining a threshold quantizer. However,
   it is very difficult to choose the exact value of quantizer
   to be considered. In our simulations, we choose 26 as a
   threshold value.
10 Encode the next frame.
```

Fig. 4. Constrained Object-Based Rate Control Algorithm.

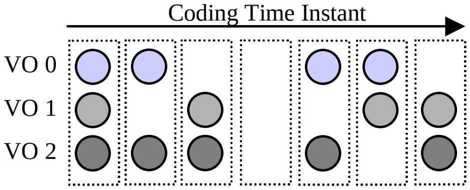


Fig. 5. Object-based Coding with Arbitrary Frameskip.

a frame among the multiple objects. In previous work [10], [11], [16], the distribution was a function of the size, motion and mean-absolute difference (MAD) of each object. Once the target bits for each object were determined, the available bits for texture were calculated by subtracting the bits used for motion, shape, and other side information. The main drawback of this approach is that there may be insufficient bits to code the texture for very low bit-rate cases, even if T_B is sufficiently larger than the header bits for the previous frame. To overcome this problem, we define the following distribution of the total target, T_B, which guarantees that the target bits for the jth object, T_j, is always larger than the sum of bits used for motion, shape, and header information, which is denoted as T_{hdr}:

$$T_j = (T_B - T_{hdr}) \cdot (w_m \mathrm{MOT}_j + w_v \mathrm{VAR}_j) + T_{hdr,j} \quad (32)$$

In the above, $\mathrm{MOT_j}$ and $\mathrm{VAR_j}$ denote the motion and MAD^2 of the jth object as defined in [10]. The weights should be $w_m, w_v \in [0,1]$ and satisfy: $w_m + w_v = 1$.

V. Object-based Rate Control: Unconstrained Temporal Rates

In this section, we consider an object-based rate control algorithm, which in contrast to the previous section, the temporal rates for each object are unconstrained. In other words, we have a freedom to choose different frameskip factors and corresponding quantization parameters for each object. While this problem shares a number of common issues with the algorithms presented above, it is unique in that we now have the opportunity to exploit the varying properties of each video object, however we must also consider the impact of rate control decisions on a shared buffer and also address the composition problem. This section introduces the general object-based coding framework, presents a set of constraints on both the rate and the buffer, and presents a solution to solving the composition problem.

A. Unconstrained Framework

To illustrate the problem being addressed, Fig. 5 shows an example of object-based coding in which each object has an arbitrary frameskip. Let M denote the set of object id's, and L denote the complete set of time indices. $M(l)$ denotes the set of coded objects at time index l $(t = t_l)$. Also, given $l \in L$, let l_0 equal the previous value of l, except when l is the first element in L; in that case $l_0 = 0$. For example, in Fig. 5, $M = 0,1,2$, $L = l_0, l_1, l_2, l_3, l_4, l_5, l_6$, $M(l_0) = 0,1,2$, $M(l_1) = 0,2$, $M(l_2) = 1,2$, and so on. Then, the constraint on the rate can be written as

$$\sum_{l\in L} \sum_{j\in M(l)} R_j(t = t_l) \le R_{\mathrm{budget}} \quad (33)$$

where $R_j(t = t_l)$ is the number of bits used for the jth object at $t = t_l$. (33) essentially says that the sum of rates for all objects at all time instants within the specified time interval must be less than the calculated bit rate budget over that time interval.

In order to ensure that buffer overflow and underflow are avoided at every coded time instant, we have a set of buffer constraints which are given by

$$B_{i+l_0} + \sum_{j\in M(l)} R_j(t = t_{l_0} : t_l) < B_{\max};\ \forall l \in L \quad (34)$$

$$B_{i+l_0} + \sum_{j\in M(l)} R_j(t = t_{l_0} : t_l) - (l - l_0) R_{drain} > 0;\ \forall l \in L \quad (35)$$

where i is the current time index, B_{i+l_0} is the current buffer level in bits, $B_{\max}$ is the maximum buffer size, and R_{drain} is the rate at which the buffer drains per time instant. $R_j(t = t_{l_0} : t_l)$ is the number of bits used for the jth VO at time index from l_0 to l.

With the above constraints, it is possible to formulate a problem that aims to minimize the overall coding distortion. Such a problem could be solved by searching over all valid combinations of frameskip factors and quantization parameters within a specified period of time. However, the complexity of such a approach is very high. Not only are there many combinations of frameskip factors and quantization for each object, but we must also track the individual time instant that each object is coded. In the following, a low-complexity object-based coding framework is considered that aims to improve overall coding efficiency.

B. Algorithm Overview

In this section, we consider the rate control algorithm for a restricted object-based framework. We refer to this framework as being restricted since a decision on the frameskip factor of a particular object is made locally, i.e., without considering the various combinations of frameskip factors. With this framework, the proposed algorithm first determines the set of objects to be coded at the next coding time, and then considers the bit allocation for each object to be coded.

The purpose of the rate control algorithm is to maximize the coding performance subject to constraints on the overall bit rate and buffer occupancy. The problem is formulated as follows:

$$\min_{M(i+f_s)\subset M} \left| d_{M(i+f_s)}(Q, f_s) \right| \quad (36)$$

$$\text{subject to} \begin{cases} \bar{R} \le R \\ B_i + \sum_{j\in M(i+f_s)} R_j(t = t_{i+f_s}) < B_{\max} \\ B_i + \sum_{j\in M(i+f_s)} R_j(t = t_{i+f_s}) - f_s \cdot R_{drain} > 0 \end{cases}$$

where $M(i + f_s)$ denotes the set of coded object. $d_{M(i+f_s)}(Q, f_s)$ represents the total distortion of objects belonging to $M(i + f_s)$ at time index $(i + f_s)$.

In order to control the rate and select the optimal coded object set, we use the rate control algorithm, shown in Fig. 6. With the rate control algorithms presented earlier, if the target bits are less than the header bits which are used to code the motion, shape and header information, then the encoder is forced to skip all objects. In the rate control presented here, however, the algorithm allows the encoder to code a subset of objects.

Similar to the frame-based rate control discussed in the previous section, we distribute the target bits to each object using (32). In order to support uniform picture quality from frame to frame, we restrict the current quantizer to the previous quantizer of the same object [10].

C. Selecting Possible Coded Sets of Objects

We first determine the number of frames to be skipped based on the current buffer fullness. In case any object cannot be coded at the current time index, the frameskip factor, f_s, is increased. However, it should be noted that the frameskip rate for the proposed object-based algorithm is smaller than that for the frame-based algorithm. This is because a subset of all objects can be coded within the unconstrained object-based coding framework.

Based on the buffer constraints given by (34) and (35) with time index $l = (i + f_s)$, we determine the possible set of objects to be coded at time index l. For each subset of the set of object id's, the subset belongs to the possible coded set, M_L, if it satisfies the constraints on the buffer.

D. Selecting an Optimal Set of Objects

The set of objects to be coded at a particular time instant is selected based on the total distortion associated with each frame, which includes both coded distortion due to the quantization error and skipped distortion due to the skipped objects.

```
1  Set $f_s = 1$, $D_{min} = \infty$
2  Calculate the initial target bits for the current time index, and
   scale this target based on the current buffer level and the buffer
   size. See Sec. III-B.
3  Compare the target bits with the motion, shape and header
   bits of the previous coded objects, and determine the coded
   object set, $M_L$, based on the target bits. The set $M_L$
   includes all possible subsets of $M$. In the case that every object
   should be skipped, $M_L$ is defined as an empty set, and we
   repeat from step 2 with $f_s = f_s + 1$. See Section V.B.2.
4  Distribute the target bits according to (32) for each object
   included in the subset $M(i + f_s)$ belonging to $M_L$.
5  Calculate the quantization parameter and the target bits for
   each object.
6  Estimate the distortion using (38) and check the buffer condi-
   tion using (36).
7  If the current distortion is less than $D_{min}$, then replace $D_{min}$
   with the current distortion and record encoding parameters.
8  Repeat from step 4 with the next subset $M(i + f_s)$ belonged
   to $M_L$.
9  Determine the optimal set of coded objects, $M^*(i + f_s)$, with
   minimum distortion according to (37) and encode the objects
   belonging to this set.
10 Update the next coding time index using $f_s$ and $M^*(i + f_s)$,
   and repeat from step 1.
```

Fig. 6. Unconstrained Object-Based Rate Control Algorithm.

The optimal set of objects to be coded, $M^*(i + f_s)$, are those that satisfy

$$d_{M^*(i+f_s)}(Q, f_s) = \min_{M(i+f_s) \subset M_L} \left| d_{M(i+f_s)}(Q, f_s) \right| \quad (37)$$

where,

$$d_{M(i+f_s)}(Q, f_s) = \sum_{j \in M(i+f_s)} D_c(Q_{j,i+f_s}) + \sum_{j \notin M(i+f_s)} D_s(Q_{j,i}, f_s) \quad (38)$$

E. Target Bit Allocation and Distribution

The initial target bits, T_V, for the current time index is allocated based on the initial assumption that every object is coded at the current time index. Similar to the frame-based bit allocation, the initial target for each object is usually calculated based on the remaining bits, T_r, the number of bits used for coding the previous jth object, $\tilde{T}_{p,j}$, the current value of f_s, the remaining number of frames, N'_r and the number of objects. The remaining number of frames, N'_r, are subtracted the value of f_s from the remaining number of frames, N_r, at $t = t_i$. If the jth object is not coded at the previous coded time index, $\tilde{T}_{p,j}$ is assigned the coded bits determined from the actually coded time index. After the initial target has been determined, the target bits are scaled according to (15).

Based on the scaled target, we now consider distributing the available bits to the objects to be coded. Let T_{hdr} be the number of bits used for the shape, motion and header information of the previous objects belonging to the subset $M(i + f_s)$, i.e.,

$$T_{hdr} = \sum_{j \in M(i+f_s)} T_{j,hdr} \quad (39)$$

In order to guarantee that the target for the jth object is always larger than $T_{j,hdr}$, we use the distribution given by eqn. (32). Given the target bits for each object, the quantizer for each object is calculated using a given rate-quantizer model.

F. Discussion on Composition Problem

This section discusses a solution to the composition problem, which is a problem associated with the coding of multiple video objects at varying temporal rates. This problem was described in [18], and in [17], shape hints that were based on a Hamming distance metric were proposed to measure the amount of change in an object boundary between frames. By using this shape hint, the possibility for the composition problem to occur could be detected in the encoder. Assuming that some tolerable amount of holes are allowed in the encoded scene, hole detection and recovery algorithms should be employed to cover up the undefined pixels. In the following, encoder and decoder side operations are presented.

To detect and avoid the composition problems in the encoder, a simple algorithm can be used [18]. First, it is important to determine the existence of any still objects, which are either full-rectangular objects that consume the entire frame or arbitrarily shaped objects that do not move.

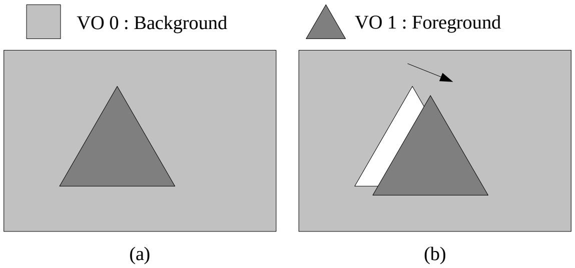


Fig. 7. Recovery of frame with Full-rectangular Object. (a) Previous frame (b) Current frame.

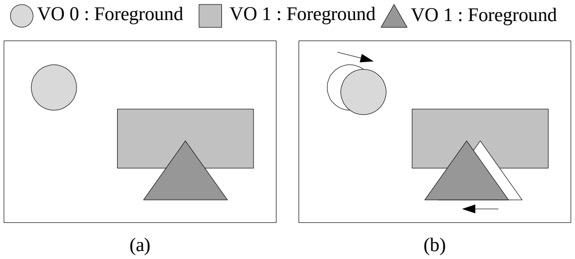


Fig. 8. Recovery of frame without Full-rectangular Object. (a) Previous frame(b) Current frame.

Then, we need to check the possibility of the composition problem for the moving objects. Since it takes at least two moving objects to cause the composition problems, we can immediately apply the proposed object-based rate allocation to the encoding process if the number of still objects is more than the total number of objects minus two. If this condition is not satisfied, we must then determine if the movement of the non-still objects is tolerable or not. To do so, each shape hint is compared to an empirically determined threshold. If the shape hint for the object is smaller than the threshold, the movement of the objects may be tolerable in the reconstructed image. If the threshold is exceeded by any one hint, then we may conclude that too much distortion in the composed scene will result. However, we should note that due to the sensitivity of the human visual system, holes may easily be detected even though they do not impact the average PSNR. In order to overcome this problem, we consider reducing this effect by recovering any undefined pixels in the decoder.

In order to reduce the effect of holes in the scene, we first check for the existence of holes, then restore the detected hole regions. Fig. 7 shows how to detect the holes in the reconstructed frame with background object which is a full-rectangular object. Let $A_{j,c}$ and $A_{j,p}$ be sets that contain pixel positions of the segmentation map of a jth object at the current and the previous coded time instant, respectively, and let $A_{j,d}$ be the difference between these two sets. Then, the set H_i which represents hole regions of the reconstructed frame at $t = t_i$ is calculated by,

$$H_i = \bigcup_{j=0}^{M-1} [A_{j,d} - \bigcup_{k=0,k\neq j}^{M-1} A_{k,c}] \tag{40}$$

If the the jth object is skipped, $A_{j,d}$ includes no elements.

In case the reconstructed frame consists of arbitrarily-shaped objects without a background object as shown in Fig. 8, the proposed algorithm detects the hole region depicted by the white region in the Fig. 9(a). In order to detect the exact hole region as shown in Fig. 9(b), we impose a restriction on the hole detection algorithm. For the set H_i, an element belongs to H_i is discarded if there is no elements of skipped objects within a search range around pixel positions of the element. Because the effect of the composition problem is already minimized at the encoder, it is not necessary to select a high value for the search range. The empirically determined value of the search range used in our experiments is 8. We should note that a search range is not needed if an object with a background exists.

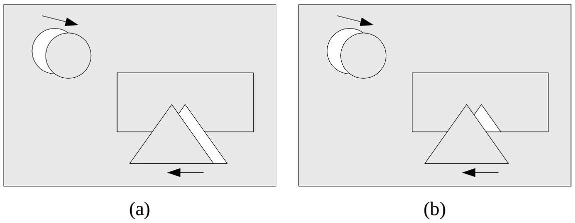


Fig. 9. Detection of Hole Regions. (a) Initial Hole Region (b) Exact Hole Region.

After detecting the holes in the reconstructed frame, we now recover the pixels within holes. We assume that one object is coded and the other is skipped in the frame. Because the coded object has the original shape of the object, changes of the coded object result in the degradation by the sensitivity of the human visual system. Therefore, the pixels within holes must be recovered from the region of the skipped objects. In order to recover holes, we use the Euclidean distance between pixels within holes and pixels in the segmentation plane of the skipped objects. Each pixel within holes is replaced by he pixel in the shape boundary of the skipped object that has a minimum distance between the pixel in the holes and the pixel in the skipped object.

VI. Simulation Results

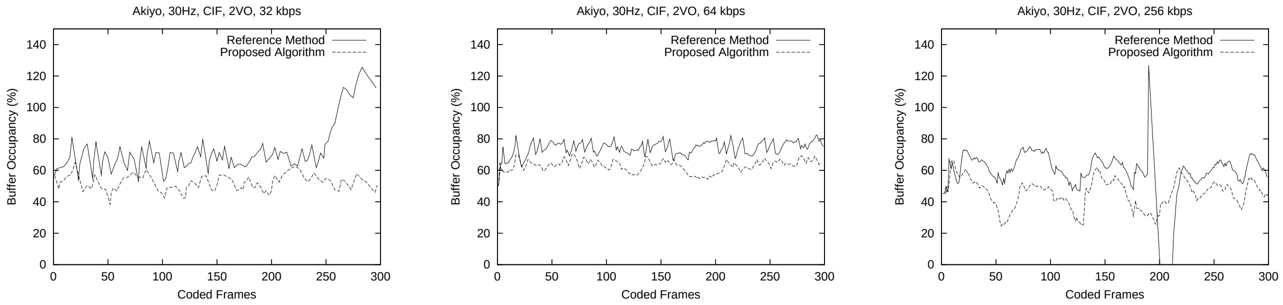
In order to evaluate the performance of the proposed algorithms, we consider the AKIYO, NEWS and COASTGUARD sequence. These sequences are encoded at different bit rates using the standard MPEG-4 rate control algorithm that is implemented as a part of the MPEG-4 reference software [12]. The bit rates that we consider range from 10 kbps to 64 kbps for low bit rate testing conditions and from 32 kbps to 256 kbps for high bit rate test conditions. The sequences are encoded at the full frame rate of the source sequence on the input and the buffer size is set to half the bit rate for the sequence [10], [12].

In Table I, we compared the performance of the proposed algorithm and that of the reference algorithm in terms of R-D values. They are calculated over all the frames, including those frames that are skipped and are simply reconstructed by copying from the previous frame. At the lowest bit rates, especially, the proposed method outperforms the reference method. In the low bit rate simulations, the reference method is forced to skip frames due to buffer constraints, whereas the

TABLE I

COMPARISONS OF PSNR VALUES FOR THE CONSTRAINED OBJECT-BASED RATE ALLOCATION

Sequence	VO	Reference Algorithm (VM5)					Frame-based Algorithm				
		32	48	64	128	256 (kbps)	32	48	64	128	256 (kbps)
AKIYO	0	39.13	39.15	39.23	41.87	45.88	39.24	39.23	39.50	42.60	45.71
	1	29.12	29.65	30.53	34.37	37.10	30.11	30.31	30.55	34.47	37.55
FOREMAN	0	32.21	30.97	31.12	31.14	31.05	33.00	31.88	31.45	31.20	31.16
	1	33.20	30.85	30.40	30.10	30.49	34.87	32.92	31.53	30.42	30.66



(a)

(b)

(c)

Fig. 10. Comparisons of Buffer Occupancy for the Frame-based Rate Allocation. (a) 32 kbps (b) 64 kbps (c) 256 kbps

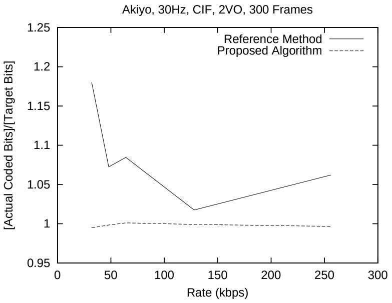


Fig. 11. Ratio of Actual Coded Bits to Target Bits for the Frame-based Rate Allocation

proposed method skips frames based on buffer constraints as well as the minimum distortion criteria.

Fig. 10 shows the buffer occupancy for the proposed method and the reference method from 32 kbps to 256 kbps. As we can see from the plots in Fig. 10, the buffer occupancy for the proposed algorithm is quite stable over the broad range of testing conditions and is always under 100%, although the test sequence has a large frame size. From these results, we can say that the buffer has little chance of overflow/underflow. As shown in Fig. 10(a) and Fig. 10(c), the buffer in the MPEG-4 reference method experiences at least one overflow.

Another key aspect of the proposed algorithm is that the actual coded bits in the proposed algorithm are similar to the target bits over a wide range of bit rates. Fig. 11 shows the ratio of the actual coded bits to the target bits.

In Table II, we compared the performance of the proposed object-based rate allocation algorithm and that of the reference algorithm in terms of R-D values. Table II shows that the proposed method outperforms the MPEG-4 reference method. While the MPEG-4 reference method is forced to skip VOP's due to buffer constraints, the proposed method skips VOP's based on buffer constraints as well as the minimum distortion criteria. We should note that it is not necessary for each time instant to include every object in the proposed algorithm. We observe that lower quantizer's are automatically assigned to a more interesting foreground object that has a higher motion.

Fig. 12 shows the ratio of the actual coded bits to the target bits for the object-based rate allocation. From Fig. 12, we can also observe that actual coded bits are well matched to the target bits in our proposed algorithm, while there are significant fluctuations in the MPEG-4 reference method.

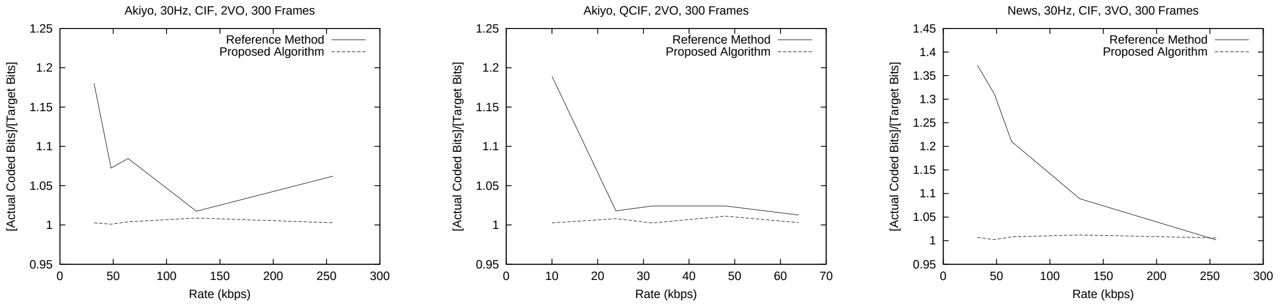
Fig. 13 shows the buffer occupancy for the proposed object-based method and the reference at different sequences and bit rates. The initial buffer level is set to 0 before coding the first I-VOP. As we can see in Fig. 13, the buffer occupancy for the proposed algorithm is quite stable over the broad range of testing conditions and is always under 100%, although the test sequences with a large frame size have been encoded. The occupancy has approximately 60% on average and variations of about 20%. From these results, we can say that the buffer has little chance of overflow or underflow, although we do not show other simulation results. As shown in Fig. 13(a), Fig. 13(b) and Fig. 13(e), the buffer in the MPEG-4 reference method experiences at least one overflow.

Fig. 14 and Fig. 15 show the results of the proposed algorithm for solving the composition problem. Fig. 14 is one example of the composition problem generated by the foreground and background object. If two objects construct the whole frame such as FOREMAN, the existence of the background object can be easily detected. Fig. 15 is one example of the composition problem generated by arbitrarily-shaped object without background object. This test is conducted on

TABLE II

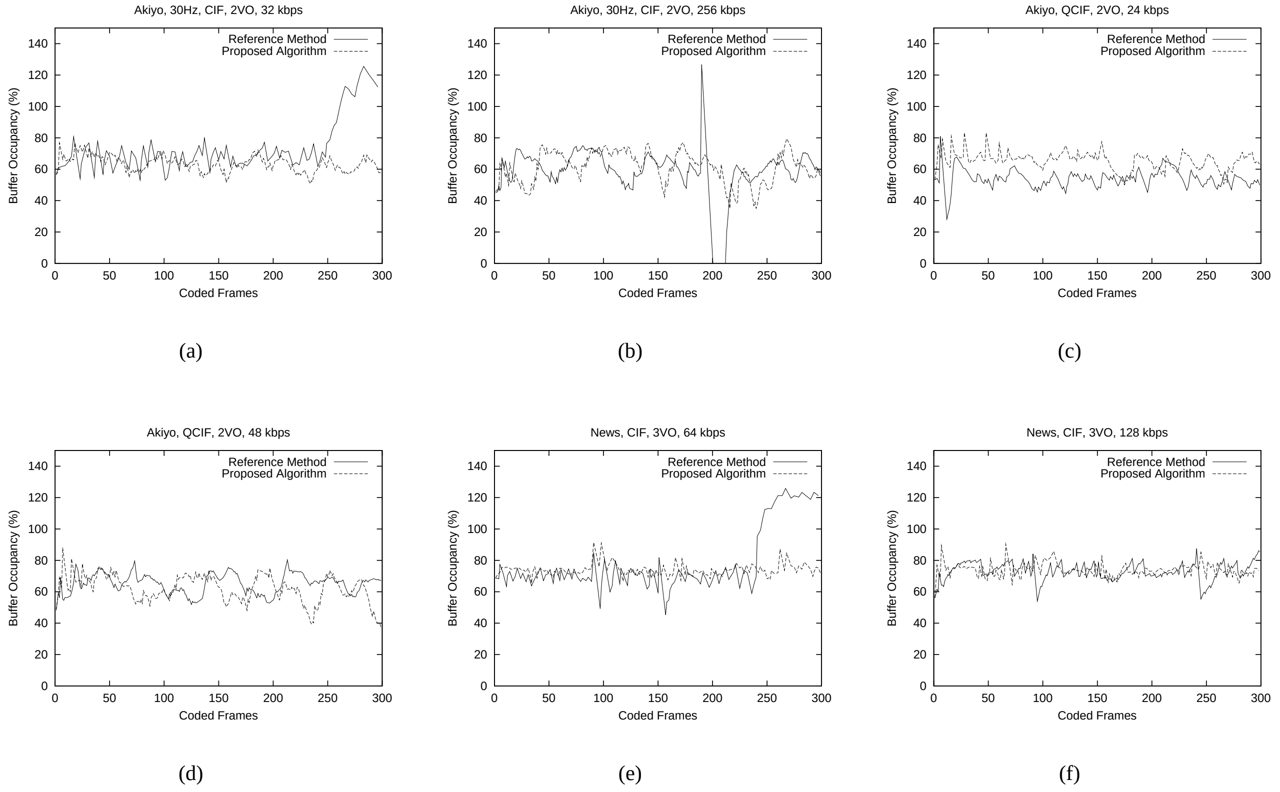
COMPARISONS OF PSNR VALUES FOR THE UNCONSTRAINED OBJECT-BASED RATE ALLOCATION

Sequence	VO	Reference Algorithm (VM5)					Object-based Algorithm				
		32	48	64	128	256 (kbps)	32	48	64	128	256 (kbps)
AKIYO (CIF)	0	39.13	39.15	39.23	41.87	45.88	39.34	40.31	40.93	44.26	45.70
	1	29.12	29.65	30.53	34.37	37.10	30.39	30.82	32.46	34.90	38.43
NEWS (CIF)	0	37.32	37.28	37.30	37.29	42.03	37.36	37.47	37.64	41.89	44.39
	1	29.05	29.03	29.00	29.36	33.88	29.11	29.23	29.15	30.83	35.48
	2	29.25	29.00	28.95	29.46	33.41	29.58	29.40	29.73	32.63	35.95
		10	24	32	48	64 (kbps)	10	24	32	48	64 (kbps)
AKIYO (QCIF)	0	36.63	36.62	36.60	38.04	40.71	36.67	39.63	41.84	44.86	45.25
	1	27.84	28.96	30.26	32.44	33.91	28.94	31.02	32.34	34.14	36.00



(a) (b) (c)

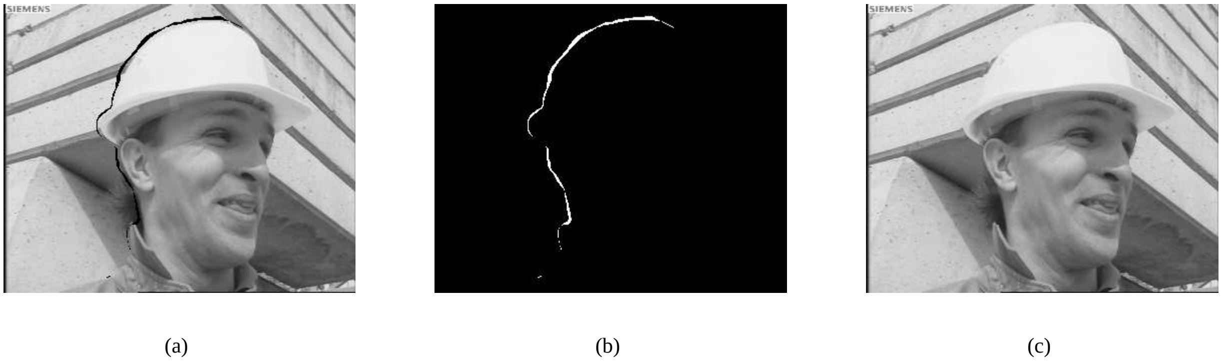
Fig. 12. Ratio of Actual Coded Bits to Target Bits for the Object-based Rate Allocation. (a) AKIYO, CIF (b) AKIYO, QCIF (c) NEWS, CIF.



(a) (b) (c)

(d) (e) (f)

Fig. 13. Comparisons of Buffer Occupancy for the Object-based Rate Allocation. (a) AKIYO, CIF, 32kbps (b) AKIYO, CIF, 256kbps (c) AKIYO, QCIF, 24kbps (d) AKIYO, QCIF, 48kbps (e) NEWS, CIF, 64kbps (f) NEWS, CIF, 128kbps



(a) (b) (c)

Fig. 14. Solution to Composition Problem for a frame with Full-rectangular Object. (a) Initial frame (b) Detected Holes (c) Recovered frame.

two video objects (VO1 and VO2) of the COASTGUARD sequence. VO1 is a large boat with some motion and complex shape, while VO2 is a small boat. VO2 goes in the opposite direction of VO1.

The white areas in Fig. 14(b) shows the result of the hole detection algorithm using (40). Fig. 14(c) shows the result of the proposed hole recovery algorithm. As shown in Fig. 14(c), there is no problem with object composition and no degradation of the picture quality by human visual system.

Fig. 15(b) and Fig. 15(d) show the detected holes and the recovered frame, respectively, when a search range is not applied. In Fig. 15(b), the white area and the grey area represent the detected holes and skipped object, respectively. The pixels in the white area should be recovered from the grey area. However, there exist the area that can be against the human visual system. Fig. 15(c) and Fig. 15(e) show the results of the hole detection algorithm with the restriction and its recovered frame, respectively. These results indicate that the reconstructed frame is more reasonable by human eyes for the case when the hole regions are recovered within a search range.

VII. Conclusions

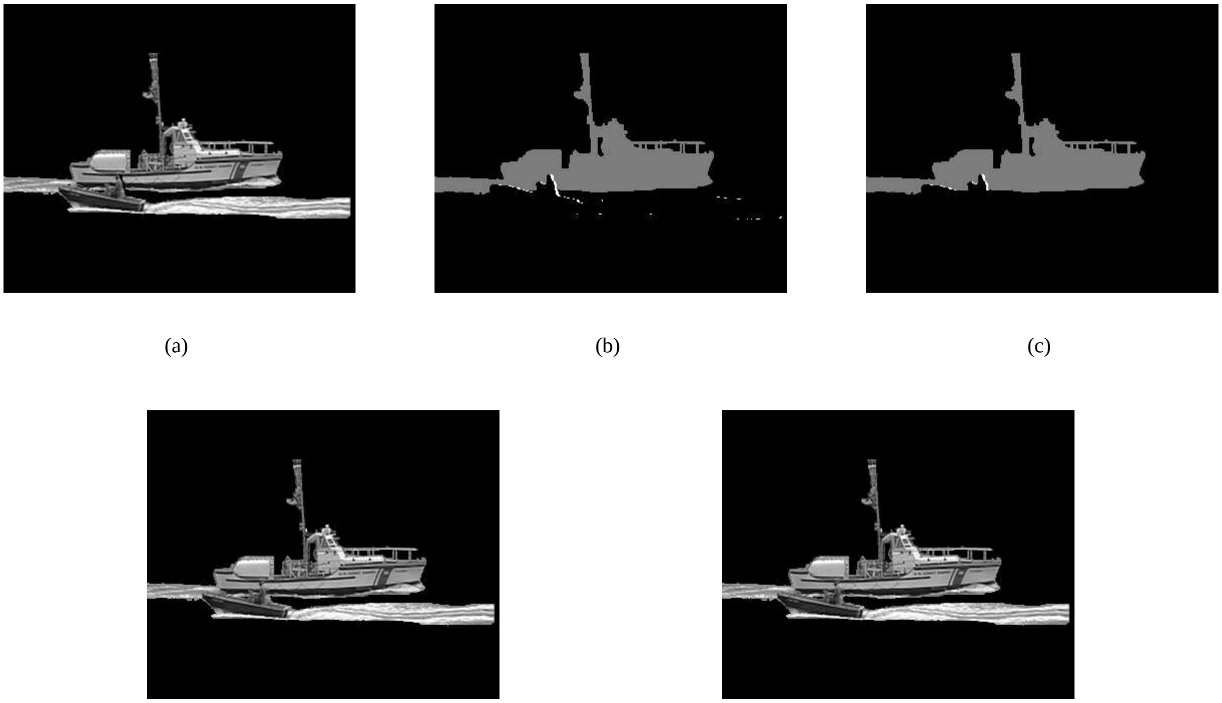
In this paper, we have proposed bit allocation algorithms that consider the trade-off between coded quality and temporal rate. In order to enable the trade-off between them to be made, we have proposed models that estimate the rate-distortion characteristics for coded frames and objects, as well as skipped frames and objects. Based on the proposed models, we have also proposed rate control algorithms for frame-based and object-based coding that minimize the overall distortion considering frameskip. For object-based coding, we considered an algorithm in which the temporal rates of each object were constrained and another algorithm in which the temporal rates were unconstrained. In addition, we have proposed solutions to the composition problem for the unconstrained coding scenario.

The distortion for coded frames/objects is expressed by classic rate-distortion models. For the distortion of skipped frames/objects, we derived an approximation from the principles of optical flow. This approximation is expressed by a function of the second order statistics of the motion and spatial gradient and provides reasonably accurate estimates for sequences with low to moderate motion. For the frame-based coding, we have proposed a rate control scheme that minimizes the distortion for video coding with frameskip.

In the constrained object-based framework, we have proposed the rate allocation method where the temporal rate of all object is constrained to be same, but the bit allocation is performed at object-level. In this framework, we have proposed the distortion model for the case that we can easily encounter during the encoding process. We have also proposed a distortion model that is applicable when the current quantizer is too large, especially in lower bit rates. It should be noted that the trade-off between spatial and temporal quality is automatically achieved by the encoder. Simulation results show that the proposed algorithm has an improved performance over the MPEG-4 reference algorithm, while the actual coded bits in the proposed algorithm are almost the same as the target bits over the entire bit rates.

In the unconstrained object-based framework, we have proposed the rate allocation method where bit allocation is performed at the object level and temporal rates of different objects may vary. In contrast to the constrained framework, the proposed algorithm allows the encoder to code a subset of objects because the proposed bit allocation is performed at the object level. By simulating the proposed algorithm at different test sequences, we observed that the proposed algorithm improved the coding efficiency about 1-2 dB than the MPEG-4 reference algorithm, while the actual coded bits in the proposed algorithm are almost the same as the target bits over a broad range of testing conditions.

Finally, we have described the composition problem associated with the object-based coding and rate allocation and proposed the methods to overcome this problem. The proposed algorithm is based on first detection changes in the shape boundaries over the time at the encoder, then employing a hole detection and recovery algorithm at the decoder. Although any holes in the reconstructed frame are negligible and do not impact the PSNR values by employing the shape hint measures at the encoder, we considered the hole detection and recovery algorithm to exclude the effect of the human visual sensitivity.



(a) (b) (c)

(d) (e)

Fig. 15. Solution to Composition Problem for a frame without Full-rectangular Object. (a) Reconstructed frame (b) Detected Holes : No Search Range (c) Detected Holes : Search Range (d) Recovered frame : No Search Range (e) Recovered frame : Search Range

References

[1] ISO/IEC 11172-2:1993, "Information Technology - Coding of moving pictures and associated audio for digital storage media at up to about 1.5 Mbit/s," Part 2:Video.

[2] ISO/IEC 13818-2:1996, "Information Technology - Generic Coding of moving pictures and associated audio," Part 2:Visual.

[3] ITU-T. Recommendation H.263, "Video Coding for Low Bit Rate Communication," 1998.

[4] ISO/IEC 14496-2:1998, "Information Technology - Coding of audio/video objects," Part 2:Visual.

[5] MPEG-2 Video Test Model 5, ISO/IEC JTC/SC29/WG11 MPEG93/457, April 1993.

[6] T. Weigand, M. Lightstone, D. Mukherjee, T.G. Campbell, and S.K. Mitra, "Rate-distortion optimized mode selection for very low bit-rate video coding and the emerging H.263 standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 6, pp. 182-190, Apr. 1996.

[7] H. Sun, W. Kwok, M. Chien and C.H. John Ju, "MPEG coding performance improvement by jointly optimizing coding mode decision and rate control," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 7, pp. 449-458, June 1997.

[8] J. Lee and B.W. Dickenson, "Rate-distortion optimized frame type selection for MPEG encoding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 7, pp. 501-510, June 1997.

[9] F.C. Martins, W. Ding and E. Feig, "Joint control of spatial quantization and temproal sampling for very low bit rate video," *Proc. ICASSP*, May 1997.

[10] A. Vetro, H. Sun, and Y. Wang, "MPEG-4 rate control for multiple video objects," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 9, pp. 186-199, Feb. 1999.

[11] H.J. Lee, T. Chiang, and Y.Q. Zhang, "Scalable Rate Control for MPEG-4 Video," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 10, pp. 878-894, Sep. 2000.

[12] ISO/IEC 14496-5:2000, "Information Technology - Coding of audio/video objects," Part 5:Reference Software.

[13] N.S. Jayant and P. Noll, *Digital Coding of Waveforms*, Prentice Hall, 1984.

[14] H.J. Lee, T. Chiang, and Y.Q. Zhang, "Scalable Rate Control for Very Low Bit Rate (VLBR) Video," *ICIP '97*, Vol. 2, pp. 768-771, 1997.

[15] T. Chiang and Y. Zhang, "A new rate control scheme using quadratic rate-distortion modeling,"*IEEE Trans. Circuits Syst. Video Technol.*, vol. 7, No. 1, pp. 246-250, Feb. 1997.

[16] H.J. Lee, T. Chiang, and Y.Q. Zhang, "Multiple-VO rate control and B-VO rate control," ISO/IEC JTC1/SC29/WG11 MPEG97/M2554, Stockholm, Sweden, July 1997.

[17] A. Vetro, H. Sun, and Y. Wang, "Object-based transcoding for adaptable video content delivery," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 11, pp. 387-401, Mar. 2001.

[18] A. Vetro and H. Sun, "Encoding and transcoding of multiple video objects with variable temporal resolution," *Proc. IEEE Int'l Symp. on Circuits and Systems*, Sydney, Australia, May 2000.

[19] H.M. Hang and J.J Chen, "Source model for transform video coder and its application - Part I: Fundamental theory," *IEEE Trans. Circuits Syst. Video Technol.*, vol.7, no.2, pp. 287-298, April 1997.

[20] W. Ding and B. Liu, "Rate control of MPEG video coding and recording by rate-quantization modeling," *IEEE Trans. Circuits and Syst. for Video Technol.*, vol.6, no.1, pp. 12-20, Feb 1996.

[21] Z. He and S. K. Mitra, "A unified rate-distortion analysis framework for transform coding," *IEEE Trans. Circuits and Syst. for Video Technol.*, vol.11, no.12, pp. 1221-1236, Dec 2001.

[22] B. Erol and F. Kossentini, "Automatic key video object plane selection using shape information in the MPEG-4 compressed domain," *IEEE Trans. Multimedia*, Vol. 2, No. 2, pp. 129-138, June 2000.

[23] A. Vetro, "Object-based encoding and transcoding," Ph.D. Disseration, Dept. of Electrical Engineering, Polytechnic University, Brooklyn, NY, June 2001.

Jeong-Woo Lee received the B.S. degree in Information & Telecommunication Engineering from Jeonbuk National University, Jeonju, Korea, in 1996, and the M.S. degree in Information and Communications Engineering from Kwanju Institute Science and Technology (K-JIST), Kwangju, Korea, in 1998. He is currently working toward the Ph.D. degree in Information and Communications Department of K-JIST. His research interests include digital video coding algorithms and implementations for H.263, MPEG-2 and MPEG-4, rate control algorithm for video coding and scalable video compression.

Anthony Vetro (S'92-M'96) received the B.S., M.S. and Ph.D. degrees in Electrical Engineering from Polytechnic University, Brooklyn, NY.

He joined Mitsubishi Electric Research Laboratories, Murray Hill, NJ, in 1996, and is currently a Senior Principal Member of the Technical Staff. Upon joining Mitsubishi, he worked on algorithms for down-conversion decoding. More recently, his work has focused on the encoding and transport of multimedia content, with emphasis on video transcoding, rate-distortion modeling and optimal bit allocation. He has published more than fifty papers in these areas and holds ten U.S. patents. Since 1997, he has been an active participant in MPEG, involved in the development of the MPEG-4 and MPEG-7 standards. He is now an editor for Part 7 of the MPEG-21 standard, Digital Item Adaptation.

Dr. Vetro has been a member of the Technical Program Committee for the International Conference on Consumer Electronics since 1998, serving as Publicity Chair in 1999 and 2000, and Tutorials Chair in 2002. He serves on the AdCom of the IEEE Consumer Electronics Society and on the Publications Committee of the IEEE TRANSACTIONS ON CONSUMER ELECTRONICS. He is also a member of the Editorial Board for the Journal of VLSI Signal Processing Systems and a member of the Technical Committee on Visual Signal Processing and Communications of the IEEE Circuits and Systems Society.

Yao Wang received the B.S. and M.S. degrees in electrical engineering from Tsinghua University, Beijing, China, in 1983 and 1985, respectively, and the Ph.D. degree in electrical engineering from University of California at Santa Barbara in 1990.

Since 1990, she has been on the faculty of Polytechnic University, Brooklyn, NY, and is presently a Professor of Electrical Engineering. From 1992 to 1996, she was a part-time consultant with AT&T Bell Laboratories, Holmdel, NJ, and has been with AT&T Labs - Research, Red Bank, NJ, since 1997. She was on sabbatical leave at Princeton University, Princeton, NJ, in 1998. Her current research interests include image and video compression for unreliable networks, motion estimation, object-oriented video coding, signal processing using multimodal information and image reconstruction problems in medical imaging.

Dr. Wang is presently an Associate Editor for IEEE TRANSACTIONS ON MULTIMEDIA and for the Journal of Visual Communications and Image Representation. She has previously served as an Associate Editor for IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY. She is a member of the Technical Committee on Multimedia Signal Processing of the IEEE Signal Processing Society and the Technical Committee on Visual Signal Processing and Communications of the IEEE Circuits and Systems Society. She has served on the organizing/technical committees of several international conferences and workshops and as Guest Editor for several special issues related to image and video coding. She won the New York City Mayors Award for Excellence in Science and Technology for the Year 2000 in the young investigator category.

Yo-Sung Ho received the B.S. and M.S. degrees in electronic engineering from Seoul National University, Seoul, Korea, in 1981 and 1983, respectively, and the Ph.D. degree in electrical and computer engineering from the University of California, Santa Barbara, in 1990.

He joined ETRI (Electronics and Telecommunications Research Institute), Daejon, Korea, in 1983. From 1990 to 1993, he was with Philips Laboratories, Briarcliff Manor, NY, where he was involved in development of the Advanced Digital High-Definition Television (AD-HDTV) system. In 1993, he rejoined the technical staff of ETRI and was involved in development of the Korean DBS digital television and high-definition television systems. Since 1995, he has been with Kwangju Institute of Science and Technology (K-JIST), where he is currently Professor of Information and Communications Department. His research interests include digital image and video coding, image analysis and image restoration, advanced coding techniques, digital video and audio broadcasting, and content-based signal representation and processing.