Cost-Distortion Optimized Unequal Error Protection for Object-Based Video Communications

Haohong Wang, *Member, IEEE*, Fan Zhai, *Member, IEEE*, Yiftach Eisenberg, *Member, IEEE*, and Aggelos K. Katsaggelos, *Fellow, IEEE*

***Abstract*—Object-based video coding is a relatively new technique to meet the fast growing demand for interactive multimedia applications. Compared with conventional frame-based video coding, it consists of two types of source data: shape information and texture information. Recently, joint source-channel coding for multimedia communications has gained increased popularity. However, very limited work has been conducted to address the problem of joint source-channel coding for object-based video. In this paper, we propose a cost-distortion optimal unequal error protection (UEP) scheme for object-based video communications. Our goal is to achieve the best video quality (minimum total expected distortion) with constraints on transmission cost and delay in a lossy network environment. The problem is solved using Lagarangian relaxation and dynamic programming. The performance of the proposed scheme is tested using simulations of a narrow-band block-fading wireless channel with additive white Gaussian noise and a simplified differentiated services Internet channel. Experimental results indicate that the proposed UEP scheme can significantly outperform equal error protection methods.**

***Index Terms*—Differentiated service (DiffServ) network, joint source-channel coding, lossy network, MPEG-4 standard, object-based video, rate distortion, unequal error protection (UEP), video coding, video communications, wireless channel.**

I. Introduction

VIDEO communications over unreliable networks has emerged as an active and challenging area of research and development. A major problem in this area is how to efficiently allocate communication resources in order to achieve the best video quality. For example, in wireless video communications, mobile devices normally have a limited battery supply. This limited energy is consumed in the processing, transmission, and displaying of the video sequence. In this paper, we consider how the average transmission power used by a modulation scheme directly affects the channel characteristics and therefore affects the received video quality. As another example, in a differentiated services (DiffServ) Internet, allocating resources discriminately for aggregated traffic flows is utilized to support various classes of quality of service (QoS). Typically, in using a pricing model, higher QoS classes cost more, but have smaller probabilities of packet loss than lower QoS classes.

Since video packets may contribute differently to the overall video quality, unequal error protection (UEP) [1]–[7] is a natural way of protecting transmitted video data. The idea is to allocate more resources to the parts of the video sequence that have a greater impact on video quality, while spending less resources on parts that are less significant. In [1], a priority encoding transmission scheme is proposed to allow a user to set different priorities of error protection for different segments of the video stream. This scheme is suitable for MPEG video, in which there is a decreasing importance among I, P, and B frames. In general, error protection can come from various sources such as forward error correction (FEC), retransmission, and transmission power adaptation. In [2], a generic UEP FEC scheme is proposed, utilizing the knowledge that almost all video packet formats have more important data closer to the packet header. Thus, the algorithm requires that better protection be applied to the data closer to the packet header. In [3], an optimal UEP scheme for layered video coding was proposed to provide an optimal bit allocation between source coding and channel coding. In [4], the tradeoff between transmission energy consumption and video quality for wireless video communications is studied, where the goal is to minimize the energy needed to transmit a video sequence with an acceptable level of video quality and tolerable delay. By assuming that the transmitter knows the relationship between the transmission power and the probability of the packet loss, the transmission power can be dynamically adjusted to control the level of protection provided for each packet [4]. Similarly, in [5], [6], the cost-distortion problem in a DiffServ network is studied by assigning unequal protection (prices) to the different packets.

In recent years, the increasing demand for multimedia applications such as networked video games and interactive digital TV indicates the growing interests in content-based interactivity. MPEG-4 [8] has become the first international multimedia communication standard to enable content-based interactivity by supporting object-based video coding. In object-based coding, the video data are composed into shape and texture information, which have completely different stochastic characteristics and bit rate proportion. For example, the shape data generally only occupies 0.5%–20% of the total bit rate [9], but can have a stronger impact on the reconstructed video quality than texture. The unbalanced contribution of shape and texture information makes UEP a natural solution to protect this type of video sequence in lossy networks. In [10]–[13], UEP schemes have been explored and applied to the transmission of MPEG-4 compressed video bit streams over error-prone

Manuscript received January 11, 2004; revised December 21, 2004. This paper was presented in part at ICIP'04. This paper was recommended by Associate Editor J. Arnold.

H. Wang is with Qualcomm, Inc., San Diego, CA 92126 USA (e-mail: haohongw@qualcomm.com)

F. Zhai is with Texas Instruments Inc, Dallas, TX 75235 USA (e-mail: fzhai@ti.com).

Y. Eisenberg is with BAE Systems, Merrimack, NH 03047 USA (e-mail: Yiftach.eisenberg@baesystems.com).

A. K. Katsaggelos is with the Department of Electrical and Computer Engineering, Northwestern University, Evanston, IL 60208 USA (e-mail: aggk@ece.northwestern.edu).

Digital Object Identifier 10.1109/TCSVT.2005.857305

wireless channels. In their work, the video is compressed using a standard block-based approach with rectangular shape, i.e., there is no shape information contained in the bit stream. Thus, the resulting bit stream is divided into partitions labeled header, motion, and texture, which are assumed to have a decreasing order of importance. Therefore, the header and motion bits receive higher levels of protection, and the texture bits receive the lowest level of protection. In [11], the I-frame data is treated as the same level of importance as header and motion data. In [12], the I-frame DCT data is subdivided into dc coefficients and ac coefficients, where dc is considered to have a higher subjective importance than ac. In [10]–[12], UEP is implemented using different channel coding rates, while in [13] it is done by separating the partitions into various streams and transmitting those streams over different carrier channels meeting different QoS levels. It is reported in [10]–[13] that the adoption of UEP results in a better performance than equal error protection (EEP) methods. However, those works have not considered video with arbitrarily shaped video objects. Furthermore, the above work is based on pre-encoded video. Thus, it does not consider optimal source coding, nor does it incorporate the error concealment strategy used at the decoder into the UEP framework.

To the best of our knowledge, there has been very limited reported work on joint source-channel coding for object-based video with arbitrarily shaped video objects. One important reason is that arbitrarily shaped video objects make the video processing and transmission much more complicated. In [14], refreshment need metrics were proposed for shape and texture to determine when these components need to be refreshed in order to improve the decoded video quality. The metrics consider the error vulnerability of the shape/texture data to channel errors and the concealment difficulty to recover the corrupted shape/texture. A rate-distortion optimal source-coding scheme was proposed recently for solving the bit allocation problem in object-based video coding [15], [16]. The experimental results in there indicate that, for some applications, the shape may have a stronger impact on the reconstructed video quality than texture. This result directly motivates the unequal protection of the shape and texture components of the video objects in video transmission. In this paper, we propose a general cost-distortion optimal UEP scheme for object-based video communications. We jointly consider source coding, packet classification, and error concealment within the framework of cost-distortion optimization. In addition, our scheme considers possible error protection from different sources for various (wireless or DiffServ) network channels. It is important to point out that the scope of this paper is limited to packet lossy networks, that is, we assume that packets are either received error-free or lost. For wireless applications, we assume that packets with errors are not available to multimedia applications.

The rest of the paper is organized as follows. Section II provides an overview of object-based video coding. In Section III, the problem of UEP for object-based video transmission is formulated. Section IV provides the optimal solution to the problem utilizing Lagrangian relaxation and dynamic programming. The implementation details are described in Section V, and experimental results are presented in Section VI. We draw conclusions in the last section.

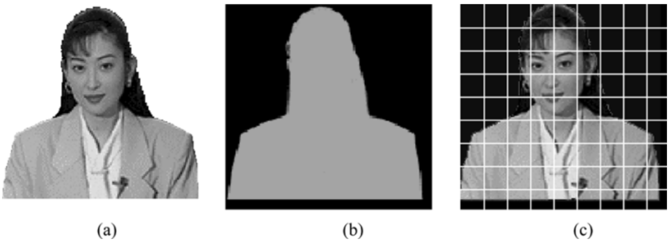


Fig. 1. Example of VO composed of shape and texture. (a) VO. (b) Shape. (c) Texture.

II. Overview of Object-Based Video Encoding

The history of object-based video coding can be traced back to 1985, when a region-based image coding technique was first published in [17]. In that work, the segmentation of the image was transmitted explicitly, and each segmented region was encoded by transmitting its contour as well as the value defining the luminance of the region. The underlying assumption is that the contours of regions are very important for subjective image quality, whereas the texture of the regions is less important. This concept has also been extended to video encoding [18]. The coder in [18] is very well suited for coding of objects with flat texture, however, the texture details within a region may get lost. In 1989, an object-based analysis-synthesis coder (OBASC) was developed [19], in which the image sequence is divided into arbitrarily shaped moving objects, which are encoded independently. The motivation behind this approach is that the shape of moving objects is more important than the texture, and the geometric distortions are less annoying to a human observer than the coding artifacts of block-based coders. OBASC was mainly successful for simple video sequences.

In recent years, object-based video coding has become an important topic in the field of visual communication. One major reason for this is the requirement of content-based interactivity and content-based scalability in modern interactive multimedia applications. MPEG-4 [7] is the first international multimedia standard that relies on object-based video representation. The central concept in MPEG-4 is that of video objects (VOs). Each VO is characterized by intrinsic properties such as shape, texture and motion. For an arbitrarily shaped video object, a frame of a VO is called a video object plane (VOP). The VOP encoder essentially consists of separate encoding schemes for shape and texture (see Fig. 1). It is important to point out that the standard does not describe the method for creating VOs, that is, they may be created in various ways depending on the application.

The purpose of using shape is to achieve an object-based video representation, as well as to possibly provide better subjective quality and increased coding efficiency (see [20] for a recent review of shape coding). The shape information for a VOP, also referred to as the alpha-plane, is specified by a binary array corresponding to the rectangular bounding box of the VOP specifying whether an input pixel belongs to the VOP or not, and a set of transparency values ranging from 0 (completely transparent) to 255 (opaque). In MPEG-4, the binary shape information is coded utilizing the macroblock-based structure, by which binary alpha data are grouped within 16×16 binary alpha blocks (BABs). BABs are classified into transparent, opaque, and border macroblocks. To reduce the bit rate, a lossy representation of a BAB might be adopted. Accordingly, the original BAB is successively downsampled by a conversion ratio factor of two or four and then upsampled back to the full resolution. Finally, each BAB (if neither transparent nor opaque) is coded using context-based arithmetic encoding (CAE).[1]

In MPEG-4, the texture information for a VOP is available in the form of a luminance (Y) and two chrominance (U, V) components. To encode texture data, the VOPs are divided into 8×8 blocks followed by two-dimensional (2-D) 8×8 discrete cosine transforms (DCTs). The resulting DCT coefficients are quantized and then zigzag scanned to form a one-dimensional (1-D) string for entropy coding. For predictive VOPs (P-VOPs), the block-based texture data are motion estimated to find a motion vector and its corresponding motion-compensated residual. The motion vectors are coded differentially, while the residual data are coded as intracoded texture data.

III. Problem Formulation

The problem at hand is to choose coding parameters for the shape and texture of a VOP so as to minimize the total expected distortion, given a cost constraint and a transmission delay constraint in a lossy network environment. This objective can also be represented by

$$\begin{aligned} &\text{Minimize} \quad E[D_{\text{tot}}] \\ &\text{Subject to} \quad C_{\text{tot}} \le C_{\max} \text{ and } T_{\text{tot}} \le T_{\max} \end{aligned} \tag{1}$$

where $E[D_{\text{tot}}]$ is the expected total distortion for the frame, C_{tot} is the total cost for a frame, T_{tot} is the total transmission delay for the frame, $C_{\max}$ is the maximum allowable cost for the frame, and $T_{\max}$ is the maximum amount of time that can be used to transmit the entire frame. We assume that there exists a higher level controller that assigns a bit budget and a cost budget to each frame in order to meet any of the delay constraints imposed by the application. Therefore, the value of $C_{\max}$ and $T_{\max}$ can vary from frame to frame, but are known constants in (1).

A. System Model

We consider an MPEG-4 compliant object-based video application, where the video is encoded using different algorithms for shape and texture. As mentioned in [9], compared to texture data, the shape data requires relatively fewer bits to encode but has a very strong impact on the video quality. Therefore, it is natural to imagine that the UEP scheme for shape and texture may provide improved performance over an EEP scheme. However, implementing a UEP scheme is not straightforward because, in the MPEG-4 video packet syntax, the shape and texture data are placed in the same packet (using a *combined packetization* scheme). If data partitioning is enabled, a motion marker is placed between the shape and texture data for resynchronization. One way to enable UEP is to use a *separated packetization* scheme, where the shape and texture are packed into separate packets. In a similar way as proposed in [10] and [11], we insert an adaptation layer between the MPEG-4 video application and the network, which can reorganize the MPEG-4 compressed bitstream into corresponding shape packets and texture packets. In addition, the adaptation layer can optimally add some forward error protection to those packets. Fig. 2 shows the architecture of the proposed video transmission system. For a wireless network using an H.223 MUX [10], we simply replace the standard adaptation layer in the H.223 multiplexing protocol with our new layer. At the receiver side, the adaptation layer merges the shape and texture packets and makes the output bit stream compatible with the MPEG-4 syntax.

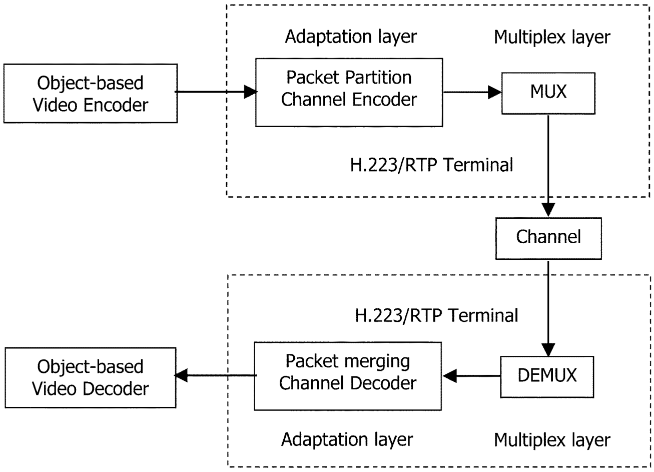

Adaptation layer
Multiplex layer
Object-based Video Encoder
Packet Partition Channel Encoder
MUX
H.223/RTP Terminal
Channel
H.223/RTP Terminal
Object-based Video Decoder
Packet merging Channel Decoder
DEMUX
Adaptation layer
Multiplex layer

Fig. 2. System block diagram.

In the following text, the *separated packetization* scheme is used as the default packetization scheme. The coded video frame is divided into 16×16 macroblocks, which are numbered in scan line order and divided into groups called slices. For each slice, there is a shape packet and a corresponding texture packet. Let I be the number of slices in the given frame and i the slice index. For each macroblock, both coding parameters for shape and texture are specified. We use μ_{S_i} and μ_{T_i}, respectively, to denote the coding parameters for all macroblocks in the ith shape and texture packets and use $B_{S_i}(\mu_{S_i})$ and $B_{T_i}(\mu_{T_i})$, respectively, to denote the corresponding encoding bit rates of these packets. It is important to point out that each packet is independently decodable in our system, that is, each packet has enough information for decoding and is independent of other packets in the same frame. This guarantees that a lost packet will not affect the decoding of other packets in the same frame. Of course, errors may propagate from one frame to the next due to motion compensation.

B. Channel Model

In addition to the source coding parameters, we assume that the transmission parameters may also be adapted per packet. By adapting the transmission parameters, we can control the effective channel characteristics, such as the probability of packet loss. Another way to view this is that each encoded packet can be sent over a set of possible transmission channels. Each transmission channel is classified using a set of parameters, e.g., the probability of packet loss and transmission rate. Let us denote by π_{S_i} and π_{T_i} the selected service classes for the ith shape and

[1] It is important to note that CAE introduces dependencies between neighboring pixels both within the same frame as well as the previous frame (for inter-coded macroblocks). As discussed in Section V-B, these dependencies make calculating the expected end-to-end distortion difficult.

texture packet, respectively. Similarly, let $\rho(\pi_{S_i})$ and $\rho(\pi_{T_i})$ denote the corresponding probability of packet loss and $R(\pi_{S_i})$ and $R(\pi_{T_i})$ the corresponding transmission rate. Then the total transmission time per frame is represented by

$$T_{\text{tot}} = \sum_{i=1}^{I} \left[\frac{B_{S_i}(\mu_{S_i})}{R(\pi_{S_i})} + \frac{B_{T_i}(\mu_{T_i})}{R(\pi_{T_i})} \right]. \tag{2}$$

Let $C(\pi_{S_i})$ and $C(\pi_{T_i})$ denote respectively the transmission cost per bit for the ith shape and texture packets. The total cost used to transmit all the packets in a frame is therefore

$$C_{\text{tot}} = \sum_{i=1}^{I} [B_{S_i}(\mu_{S_i})C(\pi_{S_i}) + B_{T_i}(\mu_{T_i})C(\pi_{T_i})]. \tag{3}$$

In a DiffServ network, the cost represents the price for each QoS channel, e.g., cents per kilobyte. We assume that the service level can be prespecified in the service level agreement (SLA) between the Internet service provider (ISP) and the users [21]. Typically, a set of parameters is used to describe the state of each service class, including the transmission rate bound and probability of packet loss. In this setting, a cost is associated with each service class as specified in the SLA. By adjusting the prices for each service class, the network can influence the class a user selects. The sender classifies each packet according to its importance in order to better utilize the available network resources.

In a wireless network, the cost we consider in this work is represented by energy per bit, i.e.,

$$C(\pi_{S_i}) = E_{b,S_i} = \frac{P(\pi_{S_i})}{R(\pi_{S_i})}$$

and

$$C(\pi_{T_i}) = E_{b,T_i} = \frac{P(\pi_{T_i})}{R(\pi_{T_i})} \tag{4}$$

where $P(\pi_{S_i})$ and $P(\pi_{T_i})$ are the corresponding transmission power for the ith shape and texture packet, respectively. The exact relationship between transmission power, transmission rate, and the probability of packet loss varies for different channel models. We provide one example in Section VI.

C. Expected Distortion

We assume that the transmitter only knows the probability with which a packet has arrived at the receiver. Thus, the distortion at the receiver is a random variable. Let $E[D_i]$ represent the expected distortion at the receiver for the ith slice. In this case

$$\begin{aligned} E[D_i] = &[1-\rho(\pi_{S_i})][1-\rho(\pi_{T_i})]E[D_{R,i}] \\ &+ [1-\rho(\pi_{S_i})]\rho(\pi_{T_i})E[D_{LT,i}] \\ &+ \rho(\pi_{S_i})[1-\rho(\pi_{T_i})]E[D_{LS,i}] \\ &+ \rho(\pi_{S_i})\rho(\pi_{T_i})E[D_{L,i}] \end{aligned} \tag{5}$$

where $E[D_{R,i}]$ is the expected distortion for the ith slice if both the shape and texture packets are received correctly at the decoder, $E[D_{LT,i}]$ is the expected distortion if the texture packet is lost, $E[D_{LS,i}]$ is the expected distortion if the shape packet is lost, and $E[D_{L,i}]$ is the expected distortion if both the shape and texture packets are lost. Clearly, $E[D_{R,i}]$ depends only on the source coding parameters for the ith packet, while $E[D_{LT,i}]$, $E[D_{LS,i}]$ and $E[D_{L,i}]$ depend on the concealment strategy used at the decoder.

Note that the problem formulation and solution approach presented in this paper are general. Therefore, the techniques developed here are applicable to various concealment strategies used by the decoder. The only assumption we make is that the concealment strategy is also known at the encoder. A common concealment strategy is to conceal the missing macroblock by using the motion information of its neighboring macroblocks. When the neighboring macroblock information is not available, the lost macroblock is typically replaced with the macroblock from the previous frame at the same location. It is also important to note that the formulation presented here is applicable to various distortion metrics. In our experimental results we use the expected mean squared error (MSE), as is commonly done in the literature [4], [6], [22].

D. Optimization Formulation

The optimization problem (1) can be rewritten as

$$\begin{aligned} \underset{\{\mu_{s_i},\mu_{T_i},\pi_{S_i},\pi_{T_i}\}}{\text{Minimize}} \quad & E[D_{\text{tot}}] \\ \text{subject to:} \quad & \sum_{i=1}^{I} [B_{S_i}(\mu_{S_i})C(\pi_{S_i}) + B_{T_i}(\mu_{T_i})C(\pi_{T_i})] \\ & \leq C_{\max} \end{aligned}$$

and

$$\sum_{i=1}^{I} \left[\frac{B_{S_i}(\mu_{S_i})}{R(\pi_{S_i})} + \frac{B_{T_i}(\mu_{T_i})}{R(\pi_{T_i})} \right] \leq T_{\max}. \tag{6}$$

In our work, we assume that the processing and propagation delays are constant and can therefore be ignored in this formulation. The only delay we are concerned with is the transmission delay. It is also important to point out that the optimization is restricted to the frame level. In other words, we do not attempt to optimally allocate the resources among the different frames of a video sequence, due to complexity considerations.

IV. Optimal Solution

In this section, we present an optimal solution for problem (6). We use the Lagrange multiplier method to relax the cost and delay constraints. The relaxed problem can then be solved using a shortest path algorithm.

The Lagrangian relaxation method leads to a convex hull approximation to the constrained problem (6). Let U be the set of all possible decision vectors u_i for the ith slice $(i = 1, 2, \ldots, I)$, where $u_1 = (\mu_{S_i}, \mu_{T_i}, \pi_{S_i}, \pi_{T_i})$. We first define a Lagrangian cost function

$$\begin{aligned} J_{\lambda_1,\lambda_2}(u) &= E[D_{\text{tot}}] + \lambda_1 C_{\text{tot}} + \lambda_2 T_{\text{tot}} \\ &= \sum_{i=1}^{I} \Bigg\{ E[D_i] \\ &\quad + \lambda_1 [B_{S_i}(\mu_{S_i})C(\pi_{S_i}) + B_{T_i}(\mu_{T_i})C(\pi_{T_i})] \\ &\quad + \lambda_2 \left[\frac{B_{S_i}(\mu_{S_i})}{R(\pi_{S_i})} + \frac{B_{T_i}(\mu_{T_i})}{R(\pi_{T_i})} \right] \Bigg\} \end{aligned} \tag{7}$$

where λ_1 and λ_2 are the Lagrange multipliers. It can easily be derived from [23] and [24] that, if there exists a pair λ_1^* and λ_2^* such that $u^* = \arg[\min_u J_{\lambda_1} \cdot_{\lambda_2} \cdot(u)]$, which leads to $C_{\text{tot}} = C_{\max}$ and $T_{\text{tot}} = T_{\max}$, then u^* is also an optimal solution to (6). Therefore, the task of solving (6) is converted into an easier one, which is to find the optimal solution to the unconstrained problem

$$\min \sum_{i=1}^{I} \Bigg\{ E[D_i] + \lambda_1 \left[B_{S_i}(\mu_{S_i}) C(\pi_{S_i}) + B_{T_i}(\mu_{T_i}) C(\pi_{T_i}) \right] + \lambda_2 \left[\frac{B_{S_i}(\mu_{S_i})}{R(\pi_{S_i})} + \frac{B_{T_i}(\mu_{T_i})}{R(\pi_{T_i})} \right] \Bigg\}. \tag{8}$$

Most decoder concealment strategies introduce dependencies between slices. For example, if the concealment algorithm uses the motion vector of the above macroblock to conceal the lost macroblock, then it would cause the calculation of the expected distortion of the current slice to depend on its previous slice. Without loss of the generality, we assume that the concealment strategy will cause the current slice to depend on its previous slices $(a \geq 0)$. To implement the algorithm for solving the optimization problem (8), we define a cost function $G_k(u_{k-a}, \ldots, u_k)$, which represents the minimum total cost, delay, and distortion up to and including the kth slice, given that $u_{k-a}, \ldots, u_k$ are decision vectors for the $(k-a)$th to kth slices. Therefore, $G_I(u_{I-a}, \ldots, u_I)$ represents the minimum total cost, delay, and distortion for all the slices of the frame, and thus

$$\min_u J_{\lambda_1, \lambda_2}(u) = \min_{u_{I-a}, \ldots, u_I} G_I(u_{I-a}, \ldots, u_I). \tag{9}$$

The key observation for deriving an efficient algorithm is the fact that given $a+1$ decision vectors $u_{k-a-1}, \cdots, u_{k-1}$ for the $(k-a-1)$th to $(k-1)$th slices, and the cost function $G_{k-1}(u_{k-a-1}, \ldots, u_{k-1})$, the selection of the next decision vector u_k is independent of the selection of the previous decision vectors $u_1, u_2, \ldots, u_{k-a-2}$. This is true since the cost function can be expressed recursively as

$$\begin{aligned} & G_k(u_{k-a}, \ldots, u_k) \\ & = \min_{u_{k-a-1}, \ldots, u_{k-1}} \Bigg\{ G_{k-1}(u_{k-a-1}, \ldots, u_{k-1}) + E[D_k] \\ & \quad + \lambda_1 \cdot \left[B_{S_k}(\mu_{S_k}) C(\pi_{S_k}) + B_{T_k}(\mu_{T_k}) C(\pi_{T_k}) \right] \\ & \quad + \lambda_2 \left[\frac{B_{S_k}(\mu_{S_k})}{R(\pi_{S_k})} + \frac{B_{T_k}(\mu_{T_k})}{R(\pi_{T_k})} \right] \Bigg\}. \end{aligned} \tag{10}$$

The recursive representation of the cost function above makes the future step of the optimization process independent from its past step, which is the foundation of dynamic programming. More information about dynamic programming and rate-distortion theory can be found in [24].

The problem can be converted into a graph theory problem of finding the shortest path in a directed acyclic graph (DAG) [24]. The computational complexity of the algorithm is $O(I \times |U|^{a+1})$ ($|U|$ is the cardinality of U), which depends directly on the value of a and $|U|$. For most cases, a is a small number, so the algorithm is much more efficient than an exhaustive search algorithm which has exponential computational complexity. When the $|U|s$ for separated packetization ($|U|_{\text{sep}}$) and combined packetization ($|U|_{\text{comb}}$) are compared, we have that $|U|_{\text{sep}} = |\pi| * |U|_{\text{comb}}$, where $|\pi|$ is the number of available service classes for packet transmission. Therefore the complexity of using separated packetization is $|\pi|^{a+1}$ times higher than using combined packetization. In this sense, $|\pi|$ becomes an important factor that could directly affect the selection of the packetization scheme.

V. Implementation Details

In this section, we demonstrate the potential of the proposed framework by considering two applications, which are: 1) video transmission over a narrow-band block-fading wireless channel with additive white Gaussian noise and 2) a simplified DiffServ-based video transmission system. We present the implementation details regarding packetization, error concealment and expected distortion calculations.

A. Packetization and Error Concealment

The packetization scheme, i.e., the number of macroblocks per packet, is not standardized within the MPEG-4 standard. Some applications pack each macroblock into a packet. This approach provides significant error resilience and encoding flexibility, but suffers from the large transmission overhead required for each packet header. In addition, since each packet must be independently decodable, this approach does not benefit from differential encoding between macroblocks. In other applications, each frame is packed into a separate packet. In this case, the transmission overhead is very small, but the resilience to channel errors is poor, e.g., an uncorrectable local error may cause the entire frame to be discarded. In order to balance error resilience and coding efficiency, we consider packing each row of macroblocks into a packet. In our simulations, we consider both the *combined* packetization scheme and the *separated* packetization scheme. Since the encoder skips those transparent 8×8 blocks in encoding the texture data, the decoding of the texture data is dependent on the shape block transparency. In the separated packetization scheme, we include four extra bits per macroblock in the texture packet, indicating the 8×8 shape block transparency, in order to make each packet independently decodable. Our experiments indicate that the extra bits only occupy a very small proportion of the compressed bitstream, the reduction therefore of this overhead is not one of the major focuses in this paper.

The error concealment strategy in our simulations is identical for both shape and texture packets. To recover the lost shape (texture) information, the decoder uses the shape (texture) motion vector of the neighboring macroblock above as the concealment motion vector. If the concealment motion vector is not available, e.g., because the above macroblock is also lost, then the decoder uses zero motion vector concealment.

B. Expected Distortion

In object-based video communications, video objects are compressed and transmitted separately. At the receiver, the decoder has the flexibility to decide how to combine the video objects in order to compose the VOP. To evaluate the distortion caused by a video object, we assume that the transmitter knows the background VOP on which the transmitted video object will be composed at the receiver. Otherwise, a default background VOP will be adopted. Therefore, the distortion is calculated as the total intensity error of the composed frame.

We use the expected MSE as our objective distortion metric. The expected distortion for the ith slice can be calculated by summing up the expected distortion of all the pixels in the slice

$$E[D_i] = \sum_{j=iN}^{iN+N-1} E[d_j] \tag{11}$$

where $E[d_j]$ is the expected distortion at the receiver for the jth pixel in the VOP, and N is the total number of pixels in the slice. Let us denote by f_n^j the original value of pixel j in VOP n and $\tilde{f}_n^j$ its reconstructed value at the decoder. By definition,

$$\begin{aligned} f_n^j &= s_n^j t_n^j + (1 - s_n^j) g_n^j \\ \tilde{f}_n^j &= \tilde{s}_n^j \tilde{t}_n^j + (1 - \tilde{s}_n^j) g_n^j \end{aligned} \tag{12}$$

and

$$\begin{aligned} E[d_j] &= E[(f_n^j - \tilde{f}_n^j)^2] \\ &= (f_n^j)^2 - 2 f_n^j E[\tilde{f}_n^j] + E[(\tilde{f}_n^j)^2] \\ &= (f_n^j)^2 - 2 f_n^j E[\tilde{s}_n^j \tilde{t}_n^j] - 2 f_n^j g_n^j + 2 f_n^j g_n^j E[\tilde{s}_n^j] \\ &\quad + E[(\tilde{s}_n^j \tilde{t}_n^j)^2] + (g_n^j)^2 - 2 g_n^j E[\tilde{s}_n^j] + (g_n^j)^2 E[(\tilde{s}_n^j)^2] \\ &\quad + 2 g_n^j E[\tilde{s}_n^j \tilde{t}_n^j] - 2 g_n^j E[(\tilde{s}_n^j)^2 \tilde{t}_n^j] \end{aligned} \tag{13}$$

where s_n^j ($s_n^j = 0$ for transparent or 1 for opaque block; only a binary shape is considered in this work) and t_n^j are the corresponding shape and texture component of f_n^j, $\tilde{s}_n^j$ and $\tilde{t}_n^j$ are the corresponding shape and texture component of $\tilde{f}_n^j$, and $\tilde{g}_n^j$ is the background pixel value at the same position.

In calculating $E[d_j]$ in (13), the first and second moments of the reconstructed shape and texture intensity value for the jth pixel are needed. In the following paragraph, we demonstrate how the first moment can be recursively calculated in time. The second moment is computed in a similar fashion, but omitted here, due to lack of space. One of the first works that utilized recursive calculations to accurately estimate the expected end-to-end distortion for frame-based video coding can be found in [22].

1) Separated Packetization Scheme: For the separated packetization scheme, since the shape data and texture data are independently transmitted and decoded

$$\begin{aligned} E[s_n^j t_n^j] &= E[s_n^j] E[t_n^j] \\ E[\tilde{s}_n^j \tilde{t}_n^j] &= E[\tilde{s}_n^j] E[\tilde{t}_n^j]. \end{aligned} \tag{14}$$

This observation enables us to efficiently calculate the expected distortion (13) by recursively calculating the necessary moments for shape and texture independently, i.e.,

$$\begin{aligned} E[\tilde{s}_n^j](I) &= [1 - \rho(\pi_{S_i})] \hat{s}_n^j + \rho(\pi_{S_i})[1 - \rho(\pi_{S_{i-1}})] E[\tilde{s}_{n-1}^k] \\ &\quad + \rho(\pi_{S_i}) \rho(\pi_{S_{i-1}}) E[\tilde{s}_{n-1}^j] \end{aligned} \tag{15}$$

$$\begin{aligned} E[\tilde{t}_n^j](I) &= [1 - \rho(\pi_{T_i})] \hat{t}_n^j + \rho(\pi_{T_i})[1 - \rho(\pi_{T_{i-1}})] E[\tilde{t}_{n-1}^k] \\ &\quad + \rho(\pi_{T_i}) \rho(\pi_{T_{i-1}}) E[\tilde{t}_{n-1}^j] \end{aligned} \tag{16}$$

$$\begin{aligned} E[\tilde{s}_n^j](P) &= [1 - \rho(\pi_{S_i})] E[\tilde{s}_{n-1}^m] \\ &\quad + \rho(\pi_{S_i})[1 - \rho(\pi_{S_{i-1}})] E[\tilde{s}_{n-1}^k] \\ &\quad + \rho(\pi_{S_i}) \rho(\pi_{S_{i-1}}) E[\tilde{s}_{n-1}^j] \end{aligned} \tag{17}$$

$$\begin{aligned} E[\tilde{t}_n^j](P) &= [1 - \rho(\pi_{T_i})](\hat{e}_n^j + E[\tilde{t}_{n-1}^m]) \\ &\quad + \rho(\pi_{T_i})[1 - \rho(\pi_{T_{i-1}})] E[\tilde{t}_{n-1}^k] \\ &\quad + \rho(\pi_{T_i}) \rho(\pi_{T_{i-1}}) E[\tilde{t}_{n-1}^j] \end{aligned} \tag{18}$$

where shape and texture are intra coded in (15) and (16), and $\hat{s}_n^j$ and $\hat{t}_n^j$ are the encoder reconstructed shape and texture of the jth pixel. If the jth pixel is lost, but its concealment motion vector is available, then it is concealed using pixel k in frame $n-1$. In (17) and (18), shape and texture are inter coded, $\hat{e}_n^j = \hat{t}_n^j - \hat{t}_{n-1}^j$, and pixel j in frame n is predicted from pixel m in frame $n-1$ (given the motion vector).

Recall that, in our framework, it is assumed that each packet is independently decodable. It is important to point out that this assumption does not hold when CAE coding is used for the shape information [25]. MPEG-4 inter-mode CAE coding uses a special correction strategy, in which a template containing pixels from the previous VOP is used to decode the current VOP. This clearly violates the independently decodable video packet rule. Should the pixels in the reference VOP be in error, the decoder will not be able to properly parse the bitstream of the current VOP even though it is received correctly. Therefore, in this work we do not consider inter-mode CAE shape coding.

2) Combined Packetization: For the *combined packetization* scheme, the task of calculating $E[\tilde{s}_n^j \tilde{t}_n^j]$ is quite complicated. As an example, we derive $E[\tilde{s}_n^j \tilde{t}_n^j]$ for the case when shape and texture are inter-mode coded

$$\begin{aligned} E[\tilde{s}_n^{j_s} \tilde{t}_n^{j_t}](P,P) &= (1 - \rho_i)(E[\tilde{s}_{n-1}^{m_s}] \hat{e}_n^{j_t} + E[\tilde{s}_{n-1}^{m_s} \tilde{t}_{n-1}^{m_t}]) \\ &\quad + \rho_i (1 - \rho_{i-1}) E[\tilde{s}_{n-1}^{k_s} \tilde{t}_{n-1}^{k_t}] \\ &\quad + \rho_{i-1} \rho_i E[\tilde{s}_{n-1}^{j_s} \tilde{t}_{n-1}^{j_t}] \end{aligned} \tag{19}$$

where $\rho_i = \rho(\pi_{S_i}) = \rho(\pi_{T_i})$, and the subscripts s and t in j_s, j_t, k_s, k_t, m_s, and m_t are used to distinguish shape from texture, because the concealment motion vectors of shape could be different from that of texture. It is important to recognize that there are inter-pixel cross-correlation terms in (19), which requires computing and storing all inter-pixel cross-correlation values for all frames in the video sequence. The amount of computation and storage is infeasible even for moderate sized frames. In order to reduce computational complexity, a model-based cross-correlation approximation method is proposed in [26] to estimate $E[\tilde{s}_n^j \tilde{t}_n^j]$ in terms of $E[\tilde{s}_n^j]$, $E[\tilde{t}_n^j]$, $E[(\tilde{s}_n^j)^2]$, $E[(\tilde{t}_n^j)^2]$, and standard deviations σ_s and σ_t. In this paper, we consider two models, in which the value of $E[\tilde{s}_n^j \tilde{t}_n^j]$ is bounded by 0 and $\sqrt{E[(\tilde{s}_n^j)^2] E[(\tilde{t}_n^j)^2]}$.

Model I: $\hat{s}_n^j$ and $\hat{t}_n^j$ are uncorrelated, so $E[\tilde{s}_n^j \tilde{t}_n^j] = E[\tilde{s}_n^j] E[\tilde{t}_n^j]$.

Model II: $\hat{t}_n^j = a + b \hat{s}_n^j$, where a and b are unknown constants ($b \geq 0$). It is not hard to derive that $E[\tilde{s}_n^j \tilde{t}_n^j] = E[\tilde{s}_n^j] E[\tilde{t}_n^j] + \sigma_s \sigma_t$.

The accuracy of these models is verified. We encode the first 120 frames of the "Bream" and "Children" video sequences with various encoding patterns, and transmit them over channels with different packet loss patterns ranging from 1% to 20%. The estimates are compared to the actual decoded distortion averaged over 100 random realizations, and the results are shown in Fig. 3. It is evident that the proposed model provides a highly accurate estimate of the decoder distortion.

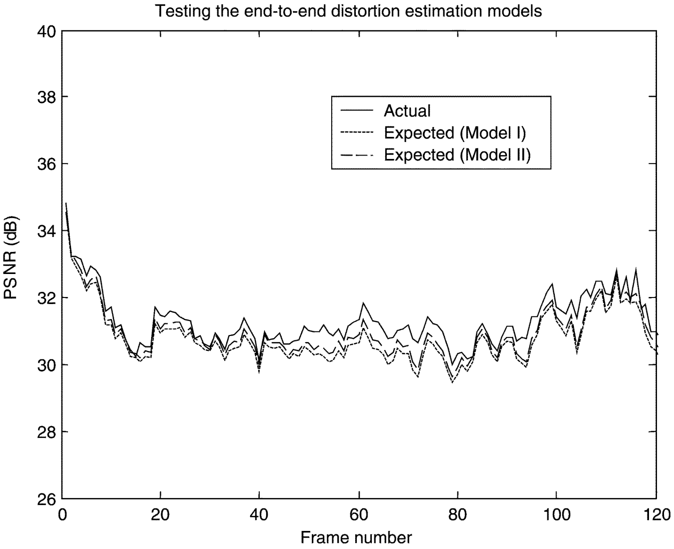


Fig. 3. Verification for distortion estimation model.

VI. Experimental Results

The main objective of the experiments presented here is to compare three error protection schemes, which are: 1) UEP-UST, an unequal error protection scheme using the *separated* packetization scheme, where the shape and texture data are placed in separate packets and therefore can be transmitted over different service channels; 2) UEP-EST, an unequal error protection scheme using *combined* packetization (i.e., the shape and texture are placed in the same packet) where the packets can be transmitted over different service channels; and 3) EEP, an equal error protection scheme using combined packetization, where all of the packets are transmitted over the same service channel.

Our simulations are based on MPEG-4 VM18.0 [8]. The available intra-mode quantizers are of step size 2, 4, 6, 8, 10, 14, 18, 24, 30, and the available inter-mode quantizers are of step size 3, 5, 7, 11, 15, 19 and 25. The texture component of each macroblock can be coded as INTRA or INTER mode. The shape can be coded as transparent, opaque, or boundary mode. For each boundary BAB, the scan type and resolution (conversion ratio of 1, 1/2, or 1/4) are also selected. As discussed earlier, inter-mode shape coding has not been considered here because it violates the assumption that each packet is independently decodable [25]. We assume that the first frame in the sequence is coded as Intra mode with a quantization step size of 6 and that enough protection is used so that it arrives correctly at the decoder. This assumption makes the initial conditions of all the experiments identical. In our simulations, we calculate the expected decoder distortion by using the actual decoded distortion averaged over 100 random realizations.

A. Wireless Channel Simulations

1) Channel Model: In our simulation for video transmission over wireless channels, we assume that each packet is sent over a narrow-band block-fading channel with additive white Gaussian noise (AWGN) [27]. We further assume the channel fading for each packet is independent and can be modeled by a random variable H. Thus, the received signal $y(t)$ can be represented by

$$y(t) = \sqrt{H}x(t) + n(t) \tag{20}$$

where $x(t)$ is the transmitted signal, and $n(t)$ is an AWGN process with power spectral density N_0. We assume that H stays fixed during the transmission of a packet and varies randomly between packets. Each realization h of H is chosen according to the *a priori* distribution $f_H(h|\theta)$, where θ is the channel state information (CSI) parameters known by the transmitter. Here we assume $\sqrt{H}$ is Rayleigh distributed and assume that $\theta = E[H]$, thus

$$f_H(h|\theta) = \frac{1}{\theta}e^{-h/\theta}, \qquad h \geq 0. \tag{21}$$

In our implementation, we assume that a packet is dropped if the capacity of the channel realization during that block is less than or equal to the information rate. For the ith shape packet, the capacity of the channel over which this packet is sent is

$$\delta(\pi_{Si}) = W\log_2\left(1 + \frac{h_{Si}P(\pi_{Si})}{N_0W}\right) \tag{22}$$

where W is the channel bandwidth. Therefore, the probability of loss for the ith shape packet is

$$\begin{aligned}\rho(\pi_{S_i}) &= \Pr\{R(\pi_{S_i}) \geq \delta(\pi_{S_i})\}\\ &= \Pr\left\{R(\pi_{S_i}) \geq W\log_2\left(1 + \frac{h_{S_i}P(\pi_{S_i})}{N_0W}\right)\right\}\\ &= \Pr\left\{h_{S_i} \leq \frac{N_0W}{P(\pi_{S_i})}\left(2^{R(\pi_{S_i})/W} - 1\right)\right\}\\ &= 1 - e^{-(N_0W(2^{R(\pi_{S_i})/W}-1))/(\theta\cdot P(\pi_{S_i}))}.\end{aligned} \tag{23}$$

Inversely, the power can be represented by

$$P(\pi_{S_i}) = -\frac{N_0W\cdot\left(2^{R(\pi_{S_i})/W} - 1\right)}{\theta\cdot\ln\rho(\pi_{S_i})}. \tag{24}$$

Similar results can be derived for the ith texture packet and the ith packet in the combined packetization.

2) Results: We encode the QCIF "Children" sequence at 10 fps and set N_0W/θ, $W = 5$ MHz, and $R = 200$ kb/s for (23) and (24). We use six classes of service channels with powers equals to 1–4, 6, and 10 W. In addition, we consider two different background VOPs (see Fig. 4 for the two reconstructed and composed frames).

In the first experiment, we use a black background VOP as shown in Fig. 4(a). We compare the UEP-UST and UEP-EST schemes with an optimal EEP reference system. Fig. 5 shows the cost-distortion (C-D) curves for these settings. Each point on the C-D curve of the optimal EEP system is obtained by trying all of the fixed power levels and choosing the one that achieves the best quality for each cost constraint. The results in Fig. 5 indicate that jointly adapting the source-coding parameters along with the selection of the transmission channel can provide significant gains in expected PSNR over EEP methods. Typically, in the UEP approaches, those packets vulnerable to packet loss (hard to be recovered by the concealment strategy employed) but

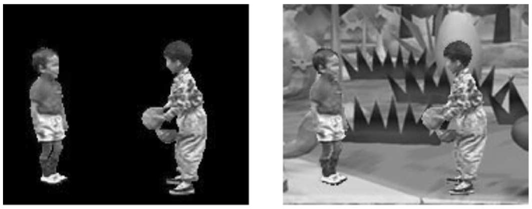

(a) (b)

Fig. 4. Reconstructed and composed "Children" frame 7 on different backgrounds.

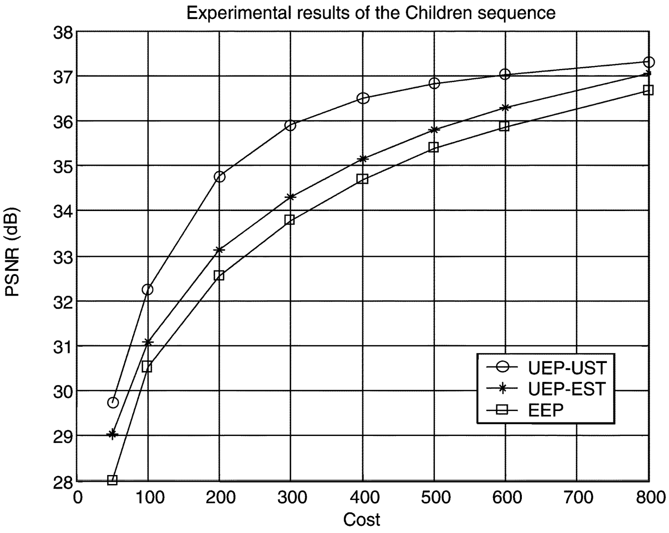


Fig. 5. Comparison of C-D curves for the UEP-UST, UEP-EST, and EEP schemes [based on Fig. 4(a)].

robust to compression (acceptable distortion from quantization or other approximation processing) are sent through the better protected channel. That is, a higher compression ratio might be used in the higher cost channels than the lower cost channels in order to efficiently distribute the total cost among the various packets. Furthermore, the UEP-UST approach outperforms the UEP-EST scheme because the UEP-UST approach has increased flexibility to providing unequal protection for shape and texture packets.

Fig. 6 shows the distributions of the shape and texture packets among the six transmission channels for the UEP-UST approach when the cost constraint $C_{\max}$ equals 100, 200, 300, and 800. The results indicate that the shape packets are better protected than texture packets. During the increasing of $C_{\max}$ from 100 to 300, the shape packets are more frequently selected than the texture packets to be transmitted through the higher cost channels. This is because shape packets have lower bit consumption but strong impact on the video quality. As shown in Fig. 6, when $C_{\max} = 300$, at least 80% of shape packets are transmitted over the most expensive channel, while over 60% of texture packets are transmitted over the two least expensive channels. In other words, the optimization process chooses to allocate more protection to the shape, because it impacts the end-to-end distortion more than the texture. Fig. 7 shows the distribution of the shape and texture bits among the six transmission channels according to the cases shown in Fig. 6. Clearly, the shape data represent a relatively small portion of the bitstream but are better protected than the texture data.

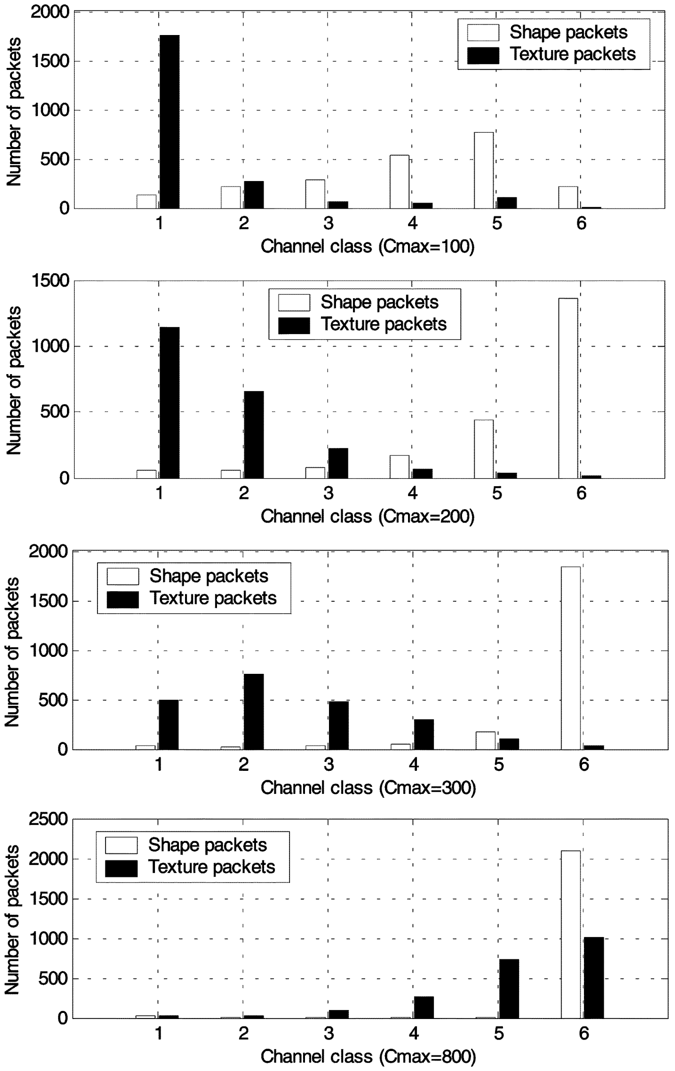


Fig. 6. Distribution of shape and texture packets in the UEP-UST system.

In the second experiment, we use the background shown in Fig. 4(b) instead of the black background in Fig. 4(b) and follow the same procedure as the first experiment. Fig. 8 shows the comparison of C-D curves using this new background. As expected, the UEP schemes both outperformed the EEP approach. However, in this case, the UEP-UST approach only has a slight gain over UEP-EST. In Fig. 4(b), the contrast of background and foreground is much weaker than that in Fig. 4(a). This directly decreases the importance of the shape information, and thus decreases the advantage of using UEP-UST compared to UEP-EST. Fig. 9 shows the distributions of the shape and texture packets by UEP-UST when $C_{\max}$ equals 100, 200, 300, and 500. It indicates that the shape is still better protected than texture, however, the protection is not as strong as that is seen in Fig. 6. In Fig. 6, over 80% of shape packets are transmitted over the most expensive channel when $C_{\max} = 300$. This does not occur until $C_{\max} = 500$ in Fig. 9. In Fig. 8, the UEP-EST approach slightly outperforms the UEP-UST approach when $C_{\max} = 600$. This is because the UEP-UST approach has more overhead than the UEP-EST approach.

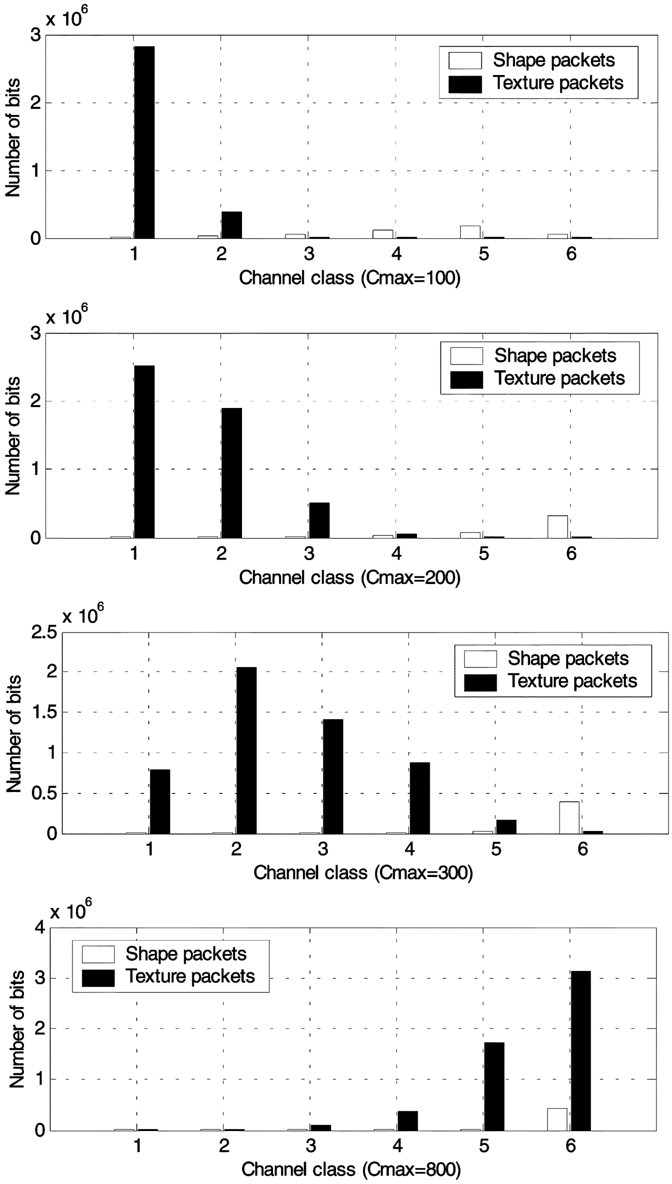


Fig. 7. Distribution of shape and texture bits in the UEP-UST system.

B. Simplified DiffServ Network Simulations

In this second set of experiments, we consider video transmission over a differentiated services network. We simulate a simplified DiffServ network as an independent time-invariant packet erasure channel. Packet loss in the network is modeled as a Bernoulli random process. In addition, a packet is considered lost if it does not arrive at the decoder on time.

There are four QoS channels available, whose parameters are defined in Table I. The costs for each class are set proportional to the average throughput of the class, which takes into account the transmission rate and probability of packet loss. In this experiment, a compressed RTP header (5 Bytes) has been added to each packet [28].

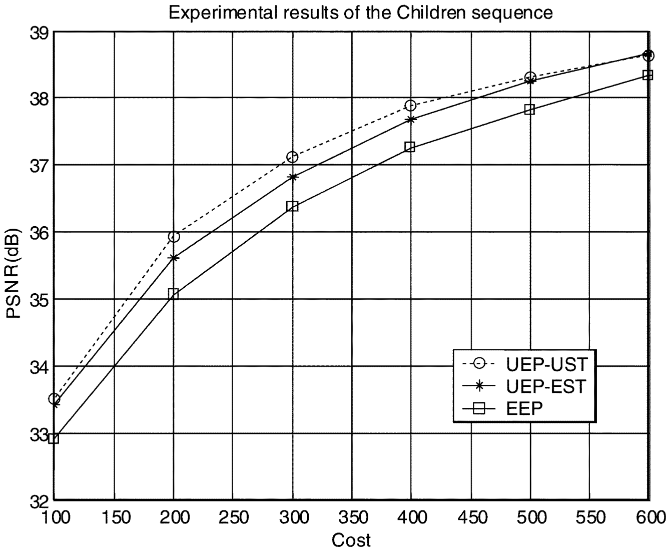


Fig. 8. Compare UEP-UST with UEP-EST and EEP [based on Fig. 4(b)].

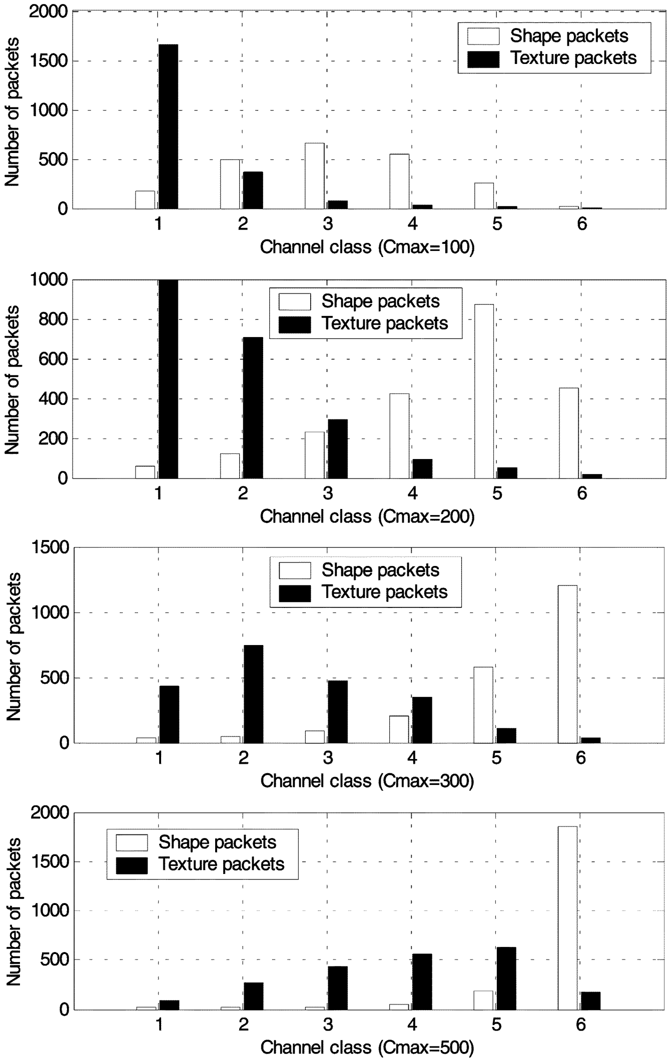


Fig. 9. Distribution of shape and texture packets in the UEP-UST system.

We encode the QCIF "Bream" sequence at 30 fps and transmitted it over the simulated DiffServ network. At the decoder, we use two different background VOPs for composition (see Fig. 10). Fig. 11 shows the C-D curves for all schemes when we use the background shown in Fig. 10(a). As expected, UEP-UST outperforms UEP-EST and both of approaches outperform the

TABLE I
PARAMETERS OF FOUR SERVICE CLASSES

Class	1	2	3	4
Probability of packet loss	0.2	0.1	0.05	0.01
Transmission rate (Kbps)	315	420	525	630
Cost (microcents/Kilobits)	10	30	60	100

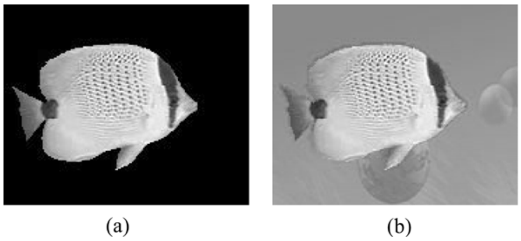

(a) (b)

Fig. 10. Reconstructed and composed "Bream" frame 4 on different background.

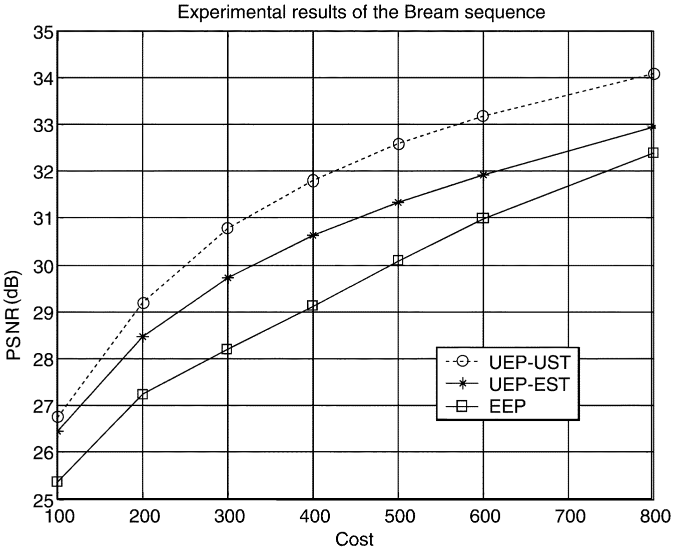


Fig. 11. Compare UEP-UST with UEP-EST and EEP [based on Fig. 10(a)].

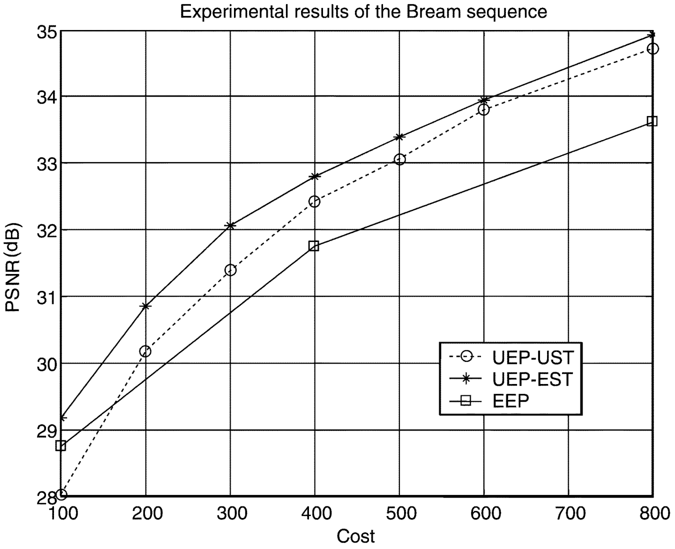


Fig. 12. Compare UEP-UST with UEP-EST and EEP [based on Fig. 10(b)].

EEP system. Fig. 12 shows the C-D curves when we use the other background. For this background, the UEP-EST scheme outperformed the UEP-UST approach. The reason is twofold: 1) the contrast of the background and foreground in Fig. 10(b) is small, which reduces the importance of shape, and directly reduces the advantage of UEP-UST over UEP-EST and 2) in the UEP-UST scheme, shape and texture are packed in separate packets, which doubles the overhead because an RTP header is required for each packet. Therefore, the UEP-UST approach spends more bits on overhead data than the UEP-EST and EEP schemes. This suggests that the benefits of unequal error protection for shape and texture information in object-based video communications are dependent on the contrast between an object and the background, as well as the amount of overhead required transmitting shape and texturing information in separate packets. Therefore, increasing the slice size for separate packetization could be another effective way to reduce the proportion of the overhead, thus in turn increase the benefits of the unequal error protection scheme.

VII. CONCLUSION

A cost-distortion optimal UEP scheme was proposed for object-based video communications. The proposed scheme jointly considers source encoding, packet classification and error concealment. Lagarangian relaxation and dynamic programming are used to solve the optimization problem. Two packetization schemes were considered in which the shape and texture information are either *combined* into the same packet, or divided into *separate* packets. Unequal error protection was studied using both packetization schemes. Several methods for estimating the end-to-end distortion were proposed and analyzed. Experiments were conducted using simulations of a narrow-band block-fading wireless channel with AWGN and a DiffServ Internet channel. Experimental results indicate that the proposed UEP schemes provide better end-to-end video quality than EEP methods. In addition, the separate packetization scheme, in which shape and texture packets receive unequal error protection, has a significant advantage over the combined packetization scheme when there is a sufficient contrast between the background and foreground of the frame.

REFERENCES

[1] A. Albanese, J. Blomer, J. Edmonds, M. Luby, and M. Sudan, "Priority encoding transmission," *IEEE. Trans. Inf. Theory*, vol. 42, no. 6, pp. 1737–1744, Nov. 1996.

[2] A. Li, J. Fahlen, T. Tian, L. Bononi, S. Kim, J. Park, and J. Villasenor, "Generic uneven level protection algorithm for multimedia data transmission over packet-switched network," in *Proc. ICCCN*, Oct. 2001, pp. 340–360.

[3] L. P. Kondi, F. Ishtiaq, and A. K. Katsaggelos, "Joint source/channel coding for SNR scalable video processing," *IEEE Trans. Image Process.*, vol. 11, no. 9, pp. 1043–1052, Sep. 2002.

[4] Y. Eisenberg, C. E. Luna, T. N. Pappas, R. Berry, and A. K. Katsaggelos, "Joint source coding and transmission power management for energy efficient wireless video communications," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 12, no. 6, pp. 441–424, Jun. 2002.

[5] A. Sehgal and P. A. Chou, "Cost-distortion optimized streaming media over DiffServ networks," in *Proc. IEEE Int. Conf. Multimedia and Expo*, Lausanne, Switzerland, Aug. 2002, pp. 857–860.

[6] F. Zhai, C. E. Luna, Y. Eisengberg, T. N. Pappas, R. Berry, and A. K. Katsaggelos, "Joint source coding and packet classification for real-time video transmission over differentiated service networks," *IEEE Trans. Multimedia*, vol. 7, no. 4, pp. 716–726, Aug. 2004.

[7] U. Horn, K. Stuhlmüller, M. Link, and B. Girod, "Robust internet video transmission based on scalable coding and unequal error protection," *Image Commun.*, vol. 15, no. 1–2, pp. 77–94, Sep. 1999.

[8] "MPEG-4 Video VM 18.0,", Pisa, Italy, ISO/IEC JTC1/SC29/WG11 N3908, 2001.

[9] H. Wang, G. M. Schuster, and A. K. Katsaggelos, "Operational rate-distortion optimal bit allocation between shape and texture for MPEG-4 video coding," in *Proc. IEEE Int. Conf. Multimedia and Expo*, Baltimore, MD, Jul. 2003, pp. 257–260.

[10] W. R. Heinzelman, M. Budagavi, and R. Talluri, "Unequal error protection of MPEG-4 compressed video," in *Proc. Int. Conf. Image Processing*, Oct. 1999, pp. 530–534.

[11] J. Cai, Q. Zhang, W. Zhu, and C. W. Chen, "An FEC-based error control scheme for wireless MPEG-4 video transmission," in *Proc. IEEE WCNC*, Chicago, IL, Sep. 2000, pp. 1243–1247.

[12] M. G. Martini and M. Chiani, "Proportional unequal error protection for MPEG-4 video transmission," in *Proc. ICC*, Helsinki, Iceland, Jun. 2001, pp. 1033–1037.

[13] S. Worrall, S. Fabri, A. H. Sadka, and A. M. Kondoz, "Prioritization of data partitioned MPEG-4 video over mobile networks," *Eur. Trans. Telecommun.*, vol. 12, no. 3, pp. 169–174, May/Jun. 2001.

[14] L. D. Soares and F. Pereira, "Refreshment need metrics for improved shape and texture object-based resilient video coding," *IEEE Trans. Image Process.*, vol. 12, no. 3, pp. 328–340, Mar. 2003.

[15] H. Wang, G. M. Schuster, and A. K. Katsaggelos, "Object-based video compression scheme with optimal bit allocation among shape, motion and texture," in *Proc. IEEE Int. Conf. Image Processing*, Barcelona, Spain, Sep. 2003, pp. 785–788.

[16] ——, "Rate-distortion optimal bit allocation for object-based video coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 15, no. 9, pp. 1113–1123, Sep. 2005.

[17] M. Kunt, "Second-generation image coding techniques," *Proc. IEEE*, vol. 73, no. 4, pp. 549–574, Apr. 1985.

[18] W. Li and M. Kunt, "Morphological segmentation applied to displaced difference coding," *Signal Process.*, vol. 38, pp. 45–56, Jul. 1994.

[19] H. Musmann, M. Hotter, and J. Ostermann, "Object-oriented analysis-synthesis coding of moving images," *Signal Process.: Image Commun.*, vol. 1, pp. 117–138, Oct. 1989.

[20] A. K. Katsaggelos, L. Kondi, F. W. Meier, J. Ostermann, and G. M. Schuster, "MPEG-4 and rate distortion based shape coding techniques," *Proc. IEEE*, vol. 86, no. 6, pp. 1126–1154, Jun. 1998.

[21] B. E. Carpenter and K. Nichols, "Differentiated service in the Internet," *Proc. IEEE*, vol. 90, no. 9, pp. 1479–1494, Sep. 2002.

[22] R. Zhang, S. L. Regunathan, and K. Rose, "Video coding with optimal inter/intra-mode switching for packet loss resilience," *IEEE J. Sel. Areas Commun.*, vol. 18, no. 6, pp. 966–976, Jun. 2000.

[23] H. Everett, "Generalized Lagrange multiplier method for solving problems of optimum allocation of resources," *Oper. Res.*, vol. 11, pp. 399–417, 1963.

[24] G. M. Schuster and A. K. Katsaggelos, *Rate-Distortion Based Video Compression: Optimal Video Frame Compression and Object Boundary Encoding*. Norwell, MA: Kluwer, 1997.

[25] N. Brady, "MPEG-4 standardized methods for the compression of arbitrarily shaped video objects," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 9, no. 8, pp. 1170–1189, Nov. 1999.

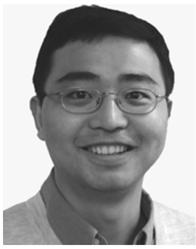
[26] H. Wang and A. K. Katsaggelos, "Robust network-adaptive object-based video encoding," in *Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Processing*, Montreal, QC, Canada, May 2004.

[27] L. Ozarow, S. Shamai, and A. Wyner, "Information theoretic considerations for cellular mobile radio," *IEEE Trans. Veh. Technol.*, vol. 43, no. 2, pp. 359–378, May 1994.

[28] S. Casner and V. Jacobson. (1999) RFC 2508—Compressing IP/UDP/RTP headers for low-speed serial links. The Internet Society. [Online]. Available: http://www.faqs.org/rfcs/rfc2508.html

[29] H. Wang, F. Zhai, Y. Eisenberg, and A. K. Katsaggelos, "Optimal object-based video communications over differentiated services networks," in *Proc. IEEE Int. Conf. Image Processing*, Singapore, Oct. 2004, pp. 3257–3260.

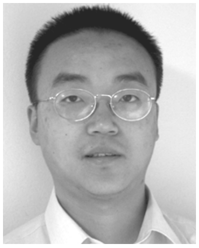
[30] H. Wang, Y. Eisenberg, F. Zhai, and A. K. Katsaggelos, "Joint object-based video encoding and power management for energy efficient wireless video communications," in *Proc. IEEE Int. Conf. Image Processing*, Singapore, Oct. 2004, pp. 2543–2546.



Haohong Wang (S'03–M'04) received the B.S. degree in computer science and the M.Eng. degree in computers and their application from Nanjing University, Nanjing, China, in 1994 and 1997, respectively, the M.S. degree in computer science from the University of New Mexico, Albuquerque, in 1998, and the Ph.D. degree in electrical and computer engineering from Northwestern University, Evanston, IL, in 2004.

He was with the Speech and Image Processing Laboratory, AT&T Research Laboratories, Florham Park, NJ, during the summer of 1998. From 1999 to 2001, he was a Member of Technical Staff with Catapult Communications, Inc., Schaumburg, IL. Since 2004, he has been with Qualcomm Inc., San Diego, CA. His research involves the areas of computer graphics, human–computer interaction, image/video analysis and compression, and multimedia signal processing and communications. Has has published approximately 30 publications in referred journals and international conferences and is the inventor of ten U.S. pending patents. He is currently serving as a Guest Editor for the *Wireless Communications and Mobile Computing* Special Issue on Video Communications for 4G Wireless Systems, and he is a co-editor of *Computer Graphics* (Beijing, China: Publishing House of Electronics Industry, 1997).

Dr. Wang is a member of the IEEE Visual Signal Processing and Communications Technical Committee and the IEEE Multimedia Communications Technical Committee. He is the Technical Program Co-Chair of the 2005 IEEE WirelessCom Symposium on Multimedia over Wireless (Maui, HI), and the 2006 International Symposium on Multimedia over Wireless (Vancouver, Canada). He will serve as Technical Program Co-Chair of the 16th International Conference on Computer Communications and Networks (ICCCN'07) in 2007.



Fan Zhai (S'99–M'04) received the B.S. and M.S. degrees from Nanjing University, Nanjing, China, in 1996 and 1998, respectively, and the Ph.D. degree from Northwestern University, Evanston, IL, in 2004, all in electrical engineering.

He is currently a System Engineer with the Digital Video Department, Texas Instruments, Inc., Dallas, TX. His research interests include image and video signal processing and compression, multimedia communications and networking, and multimedia analysis.

Yiftach Eisenberg (S'02–M'04) received the B.S. degree from the University of Illinois at Urbana-Champaign in 1999, and the M.S. and Ph.D. degrees from Northwestern University, Evanston, IL, in 2001 and 2004, respectively, all in electrical engineering.

In 2000, he was a Visiting Researcher with Motorola Laboratories, Schaumburg, IL, in the Multimedia Research Laboratory. He is currently a Senior Engineer at BAE Systems, Advanced Systems and Technology, Merrimack, NH. His current research interests include signal processing, video compression and analysis, and multimedia communications and networking.

Aggelos K. Katsaggelos (S'80–M'85–SM'92–F'98) received the Diploma degree in electrical and mechanical engineering from the Aristotelian University of Thessaloniki, Thessaloniki, Greece, in 1979 and the M.S. and Ph.D. degrees in electrical engineering from the Georgia Institute of Technology, Atlanta, in 1981 and 1985, respectively.

In 1985, he joined the Department of Electrical and Computer Engineering, Northwestern University, Evanston, IL, where he is currently a Professor, holding the Ameritech Chair of Information Technology. He is also the Director of the Motorola Center for Communications. During the 1986–1987 academic year, he was an Assistant Professor with the Department of Electrical Engineering and Computer Science, Polytechnic University, Brooklyn, NY. His current research interests include image and video recovery, video compression, motion estimation, boundary encoding, computational vision, and multimedia signal processing and communications. He is the editor of *Digital Image Restoration* (Heidelberg, Germany: Springer–Verlag, 1991), coauthor of *Rate-Distortion Based Video Compression* (Norwell, MA: Kluwer, 1997), and co-editor of *Recovery Techniques for Image and Video Compression and Transmission* (Norwell, MA: Kluwer, 1998). He is the co-inventor of nine international patents. He has served as an area editor for the journal *Graphical Models and Image Processing* (1992–1995) and is a member of the Editorial Boards of the Marcel Dekker Signal Processing Series, *Applied Signal Processing* and the *Computer Journal*, and a member of the Associate Staff, Department of Medicine, Evanston Hospital, Evanston, IL.

Dr. Katsaggelos is an Ameritech Fellow and a member of SPIE. He is a member of the Editorial Board of the IEEE PROCEEDINGS, and a member of the IEEE Technical Committees on Visual Signal Processing and Communications, and Multimedia Signal Processing. He has served as an Associate Editor for the IEEE TRANSACTIONS ON SIGNAL PROCESSING (1990–1992), a member of the Steering Committees of the IEEE TRANSACTIONS ON IMAGE PROCESSING (1992–1997) and the IEEE TRANSACTIONS ON MEDICAL IMAGING (1990–1999), a member of the IEEE Technical Committee on Image and Multi-Dimensional Signal Processing (1992–1998), the Board of Governors of the Signal Processing Society (1999–2001), the Publication Board of the IEEE Signal Processing Society (1997–2002), the IEEE TAB Magazine Committee (1997–2002), and Editor-in-Chief of the *IEEE Signal Processing Magazine* (1997–2002). He has served as the General Chairman of the 1994 Visual Communications and Image Processing Conference (Chicago, IL) and as Technical Program Co-Chair of the 1998 IEEE International Conference on Image Processing (Chicago, IL). He is the recipient of the IEEE Third Millennium Medal (2000), the IEEE Signal Processing Society Meritorious Service Award (2001), and an IEEE Signal Processing Society Best Paper Award (2001).