Multidescription Video Streaming with Optimized Reconstruction-Based DCT and Neural-Network Compensations

Xiao Su and Benjamin W. Wah, *Fellow, IEEE*

***Abstract*—Packet and compression losses are two sources of quality losses when streaming compressed video over unreliable IP networks, such as the Internet. In this paper, we propose two new approaches for concealing such losses. First, we present a joint sender-receiver approach for designing transforms in multidescription coding (MDC). In the receiver, we use a simple interpolation-based reconstruction algorithm, as sophisticated concealment techniques cannot be employed in real time. In the sender, we design an *optimized reconstruction-based discrete cosine transform* (ORB-DCT) with an objective of minimizing the mean squared error, assuming that some of the descriptions are lost and that the missing information is reconstructed by simple averaging at the destination. Second, we propose an *artificial neural network* to compensate for compression losses introduced in MDC. Experimental results show that our proposed algorithms perform well in real Internet tests.**

***Index Terms*—Artificial neural networks, error concealment, Internet, interpolation-based reconstruction, multidescription coding, video streaming.**

I. INTRODUCTION

THE INTERNET is a packet-switched, best-effort delivery service, with no guarantee on the quality of service. As a result, packets carrying real-time data may be dropped or arrive too late to be useful.

Traditional video compression algorithms are not robust to transmission errors. The sole objective of compression is to maximize coding gain, assuming error-free channels. Most video coding schemes rely on temporal-difference coding to achieve coding efficiency, thereby introducing a pervasive dependency structure into a bit stream. Hence, losses due to dropped packets or late arrivals result in the loss of subsequent dependent frames, leading to visual artifacts that can be long lasting and annoying.

A. Previous Work

To deliver video data over the Internet in real time with high quality, an active research area is to develop simple, effective error-concealment and robust coding strategies.

Error-concealment strategies can be classified by whether redundant information is added in transmissions.

1) In strategies based on redundant transmissions, one way is to insert forward error-correction codes (FEC) into packets before transmitting them and recover data in case of losses [1]–[3]. Other examples include the use of parity codes to protect every n packets by a redundant packet, *sequential protection* [2] that protects a packet by putting redundant information of previous packets into the current packet, and *fast lossy Internet image transmission* [3] that utilizes joint source channel coding to have good trade-offs of bits allocated between data and redundancy. All these schemes, however, rely on *a priori* channel-loss models that are not well defined for the Internet.

   Another strategy based on redundant transmissions exploits the time constraints of applications and arranges retransmissions in such a way that additional delay will not cause significant degradation in perception [4]. For instance, temporal dependencies of frames can be rearranged in order for a displayed frame to be referenced in the decoding of its subsequent dependent frames at a later time. Its difficulty is that it needs to adapt dynamically, even within a connection, the distance a frame is separated from its reference frame in order to achieve good quality. Moreover, retransmissions require increased bandwidth that is already a scarce resource in real-time video applications.

2) Nonredundant transmissions, in contrast, recover lost data from that received using inherent redundancies of source data [5], [6].

   One class of strategies exploit source data properties, such as edge orientations and geometric structures, in order to perform recovery. Besides being computationally expensive, these algorithms do not work well because they focus only on decoders at the receiver side, without relating to properties of encoders at the sender side.

   Another class of strategies add error resilience to coding algorithms. Coding algorithms are almost always used in video transmissions because video data has ample redundancies and is compressed before it is sent. There are two types of nonredundant robust coding algorithms that are resilient to errors in transmissions.

   a) In *layered coding* [7], data is partitioned into a base layer and a few enhancement layers. The base layer

Manuscript received May 31, 2000; revised December 1, 2000. This work was supported by a grant from Conexant Systems, Inc. The associate editor coordinating the review of this paper and approving it for publication was Prof. Jenq-Neng Hwang.

The authors are with the Department of Electrical and Computer Engineering and the Coordinated Science Laboratory, University of Illinois at Urbana-Champaign, Urbana, IL 61801 USA (e-mail: xiao-su@manip.crhc.uiuc.edu; wah@manip.crhc.uiuc.edu).

Publisher Item Identifier S 1520-9210(01)01867-3.

contains visually important video data that can be used to produce video output of acceptable quality, whereas the enhancement layers contain complementary information that allows higher-quality video data to be generated. In networks with priority support, the base layer is normally assigned a higher priority so that it has a larger chance to be delivered error free when network conditions worsen. Layered coding has been popular with ATM networks but may not be suitable for Internet transmissions for two reasons. First, the Internet does not provide priority deliveries for different layers. Second, when the packet-loss rate is high and part of the base layer is lost, it is hard to reconstruct the lost data since no redundancy is present.

b) *Multidescription coding* (MDC) divides video data into equally important streams such that the decoding quality using any subset is acceptable, and that better quality is obtained by more descriptions. It is assumed in MDC that the probability of losing all the descriptions is small. MDC has been implemented in several ways [8], [9]. A scalar-quantizer [8] applies two side-scalar quantizers in order to produce two descriptions. In order to minimize reconstruction errors when both descriptions are received, it then maps a proper subset of index pairs formed from side quantizers to central-quantizer intervals. The difficulties with this approach are that optimal index assignments are hard to achieve in real time, and that suboptimal approaches, such as A2 index assignment [8], introduce a large overhead in bit rate [10]. Instead of putting each pixel in every description, a *pair-wise correlating-transform* (PCT) [9] approach has been proposed to introduce correlations in each pair of transform coefficients and distribute the two coefficients resulted from PCT into two descriptions. This approach has high coding efficiency when both descriptions are available but has mediocre reconstruction quality with one description. It is, therefore, not applicable in an error-prone environment like the Internet because the ultimate perceived quality may be dominated by the reconstruction quality of one description.

B. Our Proposed System

In our work, we *interleave* adjacent pixels of a group of blocks (GOB—to be discussed in Section V-C) in the original video stream into multiple descriptions, code each description using a nonredundant error-concealment coding scheme, and assign different descriptions of the GOB to distinct packets in transmission to the destination. We call packets that carry related descriptions from one GOB an *interleaved set* and the number of related descriptions from one GOB the *interleaving factor*. Moreover, each packet may carry more than one descriptions from different GOBs. For example, with an interleaving factor of two, packet 0 may carry the first description of the first two GOBs, whereas packet 1 may carry the other description. We assume that packets in an interleaved set are transmitted sequentially one after another.

There are two possible sources of errors when descriptions are decompressed and combined (or reconstructed in case of loss) to render frames for playback at the receiver: *reconstruction errors* due to interpolation when some descriptions are lost, and *quantization errors* due to losses introduced in the compression algorithm when MDC is used.

To reduce reconstruction errors at the receiver when some of the descriptions are lost, we design coders at the sender for each description using a joint sender-receiver approach, instead of using previous approaches that design coders independent of reconstruction methods. The coder at the sender applies an *optimized reconstruction-based discrete cosine transform* (ORB-DCT) that minimizes reconstruction error when some of the descriptions are lost and reconstructed using average interpolation from descriptions received. We have adopted a simple reconstruction algorithm at the receiver in order to facilitate real-time playback. This combined approach leads to high reconstruction quality than that using one description, with only moderate increase in bit rate (about 20% to 30%) as compared to single-description coding. This overhead is insignificant when compared to other methods, such as FEC-based redundancy methods, for the same level of error resilience.

To compensate for quantization errors introduced in compression in MDC when all descriptions are received at the receiver, we design an artificial-neural-network (ANN) based compensation method to enhance the quality of signals received. We train an ANN offline and generalize it to sequences that are not part of the training set.

This paper is organized as follows. Section II studies packet-loss patterns of Internet transmissions in order to motivate our design of ORB-DCT in Section III. Section IV presents a study of compression losses in MDC and proposes an ANN to compensate for such losses. Finally, Section V describes our experimental results, and Section VI concludes the paper.

II. Loss Behavior in the Internet

To design an efficient error-resilient video streaming system, we conducted a series of experiments in order to answer the following pertinent questions.

- Are packet losses random or bursty?
- If they are bursty, what interleaving factor should adjacent pixels be separated in order to make the probability of unrecoverable losses sufficiently small?

From a site in Champaign (`trace12.crhc.uiuc.edu`), we chose two destination sites for our experiments: one to Berkeley (`daedalus.cs.berkeley.edu`) for representing short connections, and another to China (`public.qd.sd.cn`) for representing long-distance connections. Using a sustained bandwidth of 10 Kbytes/s that is suitable for most Internet connections, we sent packets (at the rate of 20 packets per second, each with 500 bytes) from the host in Champaign to the echo port of each of the destinations. We then recorded the sequence numbers and sending and arrival times of packets echoed back and determined packet losses based on the sequence numbers recorded. Finally, we determined the loss rate and the cumulative distribution function (CDF) of burst lengths. The packet-loss rate estimated was likely to be pessimistic since each packet traversed a round trip.

Fig. 1 depicts typical CDFs of burst lengths of connections to both sites measured at 10 pm their local time on November 19, 1999. (Due to space limitation, we do not show the CDFs at other times.) The results lead to the following conclusions.

- Packet losses are bursty. Isolated packet losses in the Champaign-China (resp. Champaign-Berkeley) connection are only 11% (resp. 39%) of total losses.
- Burst lengths are usually very small. For the Champaign-China (resp. Champaign-Berkeley) connection, the probability of bursty losses of length 4 (resp. 2) or less was as high as 94% (resp. 96%).

The CDF of burst lengths alone is not sufficient to determine the interleaving factor. In general, an interleaving factor i allows reconstructions by interpolation of bursty losses either of length $i-1$ packets or less when losses are from the same interleaved set, or of length in the range $[i,\,(2i-2)]$ when losses are from different interleaved sets.

Let the total number of packets sent be n_p and the interleaving factor be i. Over all the interleaved sets, assuming that losses of j consecutive packets, $j \leq i$, happen m_j^i times, then the total number of packets lost is n_s (independent of i), where

$$n_s = \sum_{j=1}^{i} j \times m_j^i. \tag{1}$$

Given that all the packets in an interleaved set are lost, $\Pr(fail|loss,\, i)$, the conditional probability that the content of a packet cannot be recovered by reconstruction using interleaving factor i, can be derived from (1) as follows:

$$\Pr(fail|loss,\, i) = \frac{m_i^i i}{n_s}. \tag{2}$$

$\Pr(fail|i)$, the unconditional probability that a packet cannot be reconstructed in the stream received based on interleaving factor i, can be computed as follows:

$$\Pr(fail|i) = \Pr(fail|loss,\, i)\Pr(loss) = \frac{m_i^i i}{n_s}\frac{n_s}{n_p} = \frac{m_i^i}{n_p}. \tag{3}$$

Fig. 2 shows that $\Pr(fail|loss,\, i)$ drops quickly with increasing interleaving factor i. For the Champaign-China connection, the loss rate can be as high as 50% for some part of the day, but the probability of not able to reconstruct a lost packet is held under 8% with an interleaving factor of 4 and is very close to zero at other times. For the Champaign-Berkeley connection, the failure probability is upper bounded by 5% using an interleaving factor of 2 and is negligible when using larger interleaving factors.

Based on the statistics collected, we conclude that packet losses are bursty with small burst lengths, and that packet losses can be concealed effectively by interleaving and reconstruction using an interleaving factor of four for most Internet transmissions.

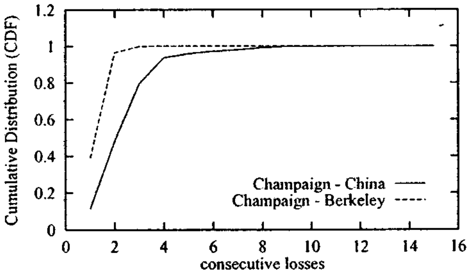

Fig. 1. Cumulative distribution function (CDF) of consecutive packet losses in connections to China and to Berkeley at 10 p.m. their local time on November 19, 1999.

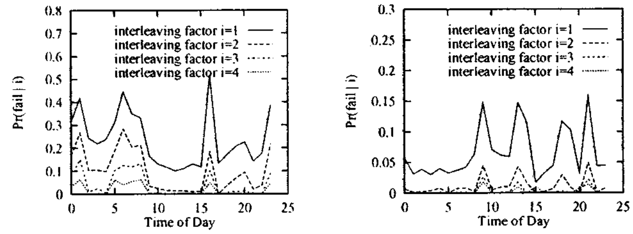

Fig. 2. $\Pr(fail|i)$, probability of bursty losses that cannot be recovered conditioned on interleaving factor i, at different times on November 19, 1999.

III. ORB-DCT To Overcome Bursty Losses

Although interleaving is effective for concealing bursty losses, a simple scheme that codes interleaved streams may not work well because the original DCT and quantizer are not necessarily the best for reconstructing lost streams. In this section we propose a new optimized reconstruction-based DCT (ORB-DCT) that takes into account the reconstruction process at the receiver.

Fig. 3 shows a simplified diagram of the basic building blocks in our proposed transform-based system. It is based on existing state-of-the-art video codecs that have three stages: transformation, lossy quantization, and lossless entropy coding. The transformation stage concentrates the energy into the first few transform coefficients and decorrelates the coefficients; the quantizer causes a controlled loss of information; and the entropy coder removes residual redundancies among quantized symbols. Our goal is to find a new transform T' in order to minimize reconstruction error $\mathcal{E}_r$ after average interpolation, based on fixed quantization Q, inverse quantization IQ, and inverse DCT T^{-1}. (We do not indicate the entropy coder in Fig. 3 as it is lossless.)

$$\mathcal{E}_r = \left\| \underbrace{Interpolate(T^{-1}(IQ(\mathbf{c})))}_{decompression+reconstruction} - \mathbf{x} \right\|^2. \tag{4}$$

In order to keep our decoders standard-compliant so that existing decoders at receivers can be used, we assume fixed inverse quantization IQ and inverse DCT T^{-1}.

With quantization in place, the minimization of $\mathcal{E}_r$ becomes an integer optimization problem, where $\mathbf{c}$ in (4) takes integer values. Such optimizations are computationally prohibitive in real time. In the following, we derive an approximate solution

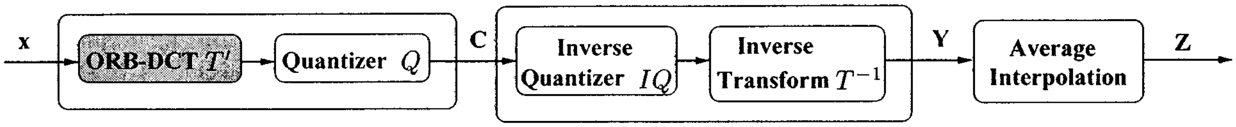

Fig. 3. Basic building blocks of a modified codec. (The shaded block is our proposed ORB-DCT.)

that does not take into account quantization effects. Specifically, the objective to be optimized in the approximation is

$$\mathcal{E}_r = ||Interpolate(T^{-1}(\mathbf{c})) - \mathbf{x}||^2. \tag{5}$$

The resulting transform is called optimized reconstruction-based DCT (ORB-DCT). In the following, we derive ORB-DCT based on partitioning video data into two descriptions and in Section III-C, extensions to four descriptions.

A. ORB-DCT for Intracoded Blocks

Assume that the original frame is divided into blocks of 8×16 pixels. After ORB-DCT, $\mathbf{X}$ is transformed into $\mathbf{C_1}$ and $\mathbf{C_2}$, each of size 8×8, that correspond to blocks of odd-numbered and even-numbered pixels. Since the derivations are similar, we only show the derivations for $\mathbf{C_1}$.

Our objective is to find $\mathbf{C_1}$ to minimize $\mathcal{E}_r$. After inverse DCT, output $\mathbf{Y_1}$ is calculated as

$$\mathbf{Y_1} = \sum_{i=1}^{8} \sum_{j=1}^{8} C_{i,j} \mathbf{b_i} \mathbf{b_j}^T \tag{6}$$

where $C_{i,j}$ is the (i, j)th element of $\mathbf{C_1}$, and $\mathbf{b_i}$ is the ith basis vector of DCT.

Putting (6) in matrix form yields

$$\mathbf{Y_1} = (\mathbf{p_1}\ \mathbf{p_2}\ \cdots\ \mathbf{p_8})_{8\times 8} \tag{7}$$

where $\mathbf{p_k} = \sum_{i=1}^{8} \sum_{j=1}^{8} C_{i,j} \mathbf{b_i} b_{j,k}$, $k = 1, \cdots, 8$, and $b_{j,k}$ is the kth component of basis vector $\mathbf{b_j}$. The set of interpolated pixels $\mathbf{Z}$ is obtained by inserting even-numbered columns as the average of columns from $\mathbf{Y_1}$, with the boundary column duplicated:

$$\mathbf{Z} = \left(\mathbf{p_1}\ \frac{\mathbf{p_1}+\mathbf{p_2}}{2}\ \mathbf{p_2}\ \frac{\mathbf{p_2}+\mathbf{p_3}}{2}\ \cdots\ \mathbf{p_8}\ \mathbf{p_8}\right)_{8\times 16}. \tag{8}$$

$\mathbf{Z}$ can also be expressed as

$$\mathbf{Z} = \sum_{i=1}^{8} \sum_{j=1}^{8} C_{i,j} \mathbf{b_i} \mathbf{e_j}^T \tag{9}$$

where $\mathbf{e_j} = (b_{j,1}\ ((b_{j,1} + b_{j,2})/2)\ b_{j,2}\ \cdots\ b_{j,8}\ b_{j,8})^T$. We define $\mathbf{e_j}$ as an extended basis vector for reconstruction purpose. The distortion between the original and the received and reconstructed pixels is

$$\mathcal{E}_r = \left\| \sum_{i=1}^{8} \sum_{j=1}^{8} C_{i,j} \mathbf{b_i} \mathbf{e_j}^T - \mathbf{X} \right\|^2. \tag{10}$$

To minimize $\mathcal{E}_r$ with respect to $\mathbf{C}$, we first linearize each matrix using operator $\mathcal{L}(\cdot)$ into a vector by a raster-scan order, i.e., following the first row by the second row in a matrix, and so on. The following notations are defined after linearization:

$$\vec{\mathbf{u}} = \mathcal{L}\left\{(C_{i,j})_{(8\times 8)}\right\}; \qquad \vec{\mathbf{v}}_{\mathbf{8(i-1)+j}} = \mathcal{L}\left\{\mathbf{b_i}\mathbf{e_j}^T_{(8\times 16)}\right\};$$
$$\vec{\mathbf{w}} = \mathcal{L}\left\{(X_{i,j})_{(8\times 16)}\right\}; \qquad i, j = 1, \cdots, 8.$$

We further define matrix $\mathbf{V}$ as

$$\mathbf{V} = (\vec{\mathbf{v}}_{\mathbf{1}}\ \vec{\mathbf{v}}_{\mathbf{2}}\ \vec{\mathbf{v}}_{\mathbf{3}}\ \cdots\ \vec{\mathbf{v}}_{\mathbf{64}}). \tag{11}$$

Then (10) can be rewritten as follows:

$$\mathcal{E}_r = ||\mathbf{V}\vec{\mathbf{u}} - \vec{\mathbf{w}}||^2, \tag{12}$$

where $\mathbf{V}$ is a 128×64 matrix, $\vec{\mathbf{u}}$, a 64×1 vector, and $\vec{\mathbf{w}}$, a 128×1 vector. Since the linear system of equations $\mathbf{V}\vec{\mathbf{u}} = \vec{\mathbf{w}}$ is an over-determined one, there exists at least one least-square solution $\vec{\mathbf{u}}$ that minimizes (12), according to the theory of linear algebra [11]. Specifically, the solution $\vec{\mathbf{u}}$ with the smallest length $|\vec{\mathbf{u}}|^2$ can be found by first performing SVD decomposition of matrix $\mathbf{V}$:

$$\mathbf{V} = \mathbf{S}[\text{diag}(w_j)]\mathbf{D}^T, \qquad j = 1, 2, \cdots, 64 \tag{13}$$

where $\mathbf{S}$ is a 128×64 column-orthogonal matrix, $[\text{diag}(w_j)]$, a 64×64 diagonal matrix with positive or zero elements (singular values), and $\mathbf{D}$, a 64×64 orthogonal matrix. Then the least-square solution can be expressed as

$$\vec{\mathbf{u}} = \mathbf{D}[\text{diag}(1/w_j)]\mathbf{S}^T\vec{\mathbf{w}}. \tag{14}$$

In the above diagonal matrix $[\text{diag}(1/w_j)]$, element $1/w_j$ is replaced by zero if w_j is zero. Therefore, ORB-DCT is a product of three matrices: $\mathbf{D}$, $[\text{diag}(1/w_j)]$, and $\mathbf{S}^T$.

To derive the ORB-DCT transform for $\mathbf{C_2}$, simply replace $\mathbf{e_j}$, $j = 1, 2, \cdots, 8$, in (9) by

$$\mathbf{e_j} = \left(b_{j,1}\ b_{j,1}\ \frac{b_{j,1}+b_{j,2}}{2}\ b_{j,2}\ \cdots\ \frac{b_{j,7}+b_{j,8}}{2}\ b_{j,8}\right)^T.$$

The rest of the steps are similar.

B. ORB-DCT for Intercoded Blocks

For intercoded blocks, output $\mathbf{Y_1}$ after inverse DCT, as shown in (6), is the residual block after motion prediction. Denote its corresponding reference block as

$$\mathbf{R} = (\mathbf{r_1}\ \mathbf{r_2}\ \cdots\ \mathbf{r_8})_{8\times 8}. \tag{15}$$

Then the interpolated data $\mathbf{Z}$ is the sum of two terms after motion compensation:

$$\begin{aligned}\mathbf{Z} &= \left(\mathbf{p_1}\ \frac{\mathbf{p_1}+\mathbf{p_2}}{2}\ \mathbf{p_2}\ \frac{\mathbf{p_2}+\mathbf{p_3}}{2}\ \cdots\ \mathbf{p_8}\ \mathbf{p_8}\right)\\ &\quad+\left(\mathbf{r_1}\ \frac{\mathbf{r_1}+\mathbf{r_2}}{2}\ \mathbf{r_2}\ \frac{\mathbf{r_2}+\mathbf{r_3}}{2}\ \cdots\ \mathbf{r_8}\ \mathbf{r_8}\right)\\ &=\sum_{i=1}^{8}\sum_{j=1}^{8}C_{i,j}\mathbf{b_i}\mathbf{e_j}^T+\mathbf{R}'. \qquad (16)\end{aligned}$$

Substituting (16) into (10) yields the reconstruction error for intercoded blocks:

$$\mathcal{E}_r=\left\|\sum_{i=1}^{8}\sum_{j=1}^{8}C_{i,j}\mathbf{b_i}\mathbf{e_j}^T-(\mathbf{X}-\mathbf{R}')\right\|^2. \qquad (17)$$

To derive ORB-DCT in this case, we note that only vector $\vec{\mathbf{w}}$ is different as compared to the case of intracoded blocks. From (14), it is obvious that the transform itself does not depend on $\vec{\mathbf{w}}$; therefore, ORB-DCT retains the same form.

In short, ORB-DCT can be applied uniformly to both intra and intercoded blocks. For intracoded blocks, it is applied to an original block $\mathbf{X}$ to produce transform coefficients $\mathbf{C_i}$, $i=1, 2$; for intercoded blocks, it is applied to the interpolated motion-predicted block $(\mathbf{X}-\mathbf{R}')$.

Like DCT, ORB-DCT is also a row-column-separable transform. To compute a transform coefficient of ORB-DCT by a row-column approach, it takes 40 floating-point multiplications and 37 floating-point additions. In the future, we plan to study fast implementations of ORB-DCT, similar to what was done in deriving fast DCT.

C. Handling Burst Lengths of Four

As described in Section II, bursty losses of length greater than two are likely for transcontinental connections. In this section we describe two ways to handle cases with a maximum burst length of four. We do not describe methods to handle longer burst lengths because such cases are infrequent and end-to-end delays will be intolerable.

We can partition video data into four descriptions by interleaving the original frame $\vec{\mathbf{z}}$ in the horizontal direction into two streams, $\vec{\mathbf{z}}_{\mathbf{h_1}}$ and $\vec{\mathbf{z}}_{\mathbf{h_2}}$, and then by interleaving and transformations in the vertical direction to get two additional descriptions. In a different way, we can also get four descriptions by first partitioning in the vertical direction and then in horizontal direction. The four descriptions, $\vec{\mathbf{z}}_{\mathbf{h_1},\mathbf{v_1}}$, $\vec{\mathbf{z}}_{\mathbf{h_1},\mathbf{v_2}}$, $\vec{\mathbf{z}}_{\mathbf{h_2},\mathbf{v_1}}$ and $\vec{\mathbf{z}}_{\mathbf{h_2},\mathbf{v_2}}$, are then sent in distinct packets to the receiver.

First, assume that only one out of three interleaved descriptions, say Description 1 ($\vec{\mathbf{z}}_{\mathbf{h_1},\mathbf{v_1}}$), is received. The remaining three descriptions can be restored as follows:

$$\hat{z}_{i,j}=\begin{cases}\dfrac{(z_{i,j-1}+z_{i,j+1})}{2} & \hat{z}_{i,j}\in\vec{\mathbf{z}}_{\mathbf{h_2},\mathbf{v_1}}\\[2ex] \dfrac{(z_{i-1,j}+z_{i+1,j})}{2} & \hat{z}_{i,j}\in\vec{\mathbf{z}}_{\mathbf{h_1},\mathbf{v_2}}\\[2ex] \dfrac{\begin{array}{l}(z_{i-1,j-1}+z_{i-1,j+1}\\ \quad+z_{i+1,j-1}+z_{i+1,j-1})\end{array}}{4} & \hat{z}_{i,j}\in\vec{\mathbf{z}}_{\mathbf{h_2},\mathbf{v_2}}\end{cases} \qquad (18)$$

where $z_{i,j}$ is the value of the pixel in row i and column j. The transformed values of Description 1 in order to achieve the optimal reconstruction in (18) can be derived as outlined in Sections III-A and III-B. In a similar way, we need to derive transformations when no, two, or three descriptions are lost. Since it is impossible to know the specific loss pattern for an interleaved set until it is received at the receiver and it will be either overly optimistic or overly pessimistic if one loss pattern is selected *a priori*, the method is impractical for use on the Internet.

Second, we can carry out the following operations based on the inverse flow of the interleaving process in order to reconstruct any missing descriptions.

1) If one out of the four interleaved descriptions is received, say $\vec{\mathbf{z}}_{\mathbf{h_1},\mathbf{v_1}}$, then $\vec{\mathbf{z}}_{\mathbf{h_1},\mathbf{v_2}}$ can be reconstructed optimally by taking averages along the vertical direction of pixels from $\vec{\mathbf{z}}_{\mathbf{h_1},\mathbf{v_1}}$. By taking averages along the horizontal direction, $\vec{\mathbf{z}}_{\mathbf{h_2},\mathbf{v_1}}$ and $\vec{\mathbf{z}}_{\mathbf{h_2},\mathbf{v_2}}$ can then be recovered.
2) If two out of the four interleaved descriptions are received, then there are two possible scenarios. If the lost descriptions are from the same horizontally interleaved group, say $\vec{\mathbf{z}}_{\mathbf{h_1},\mathbf{v_1}}$ and $\vec{\mathbf{z}}_{\mathbf{h_1},\mathbf{v_2}}$, then they can be reconstructed by averaging of their horizontal neighbors. If the lost descriptions do not belong to the same horizontally interleaved description, say $\vec{\mathbf{z}}_{\mathbf{h_1},\mathbf{v_1}}$ and $\vec{\mathbf{z}}_{\mathbf{h_2},\mathbf{v_2}}$, then they can be reconstructed optimally by taking averages of their respective vertical neighbors.
3) If three out of the four descriptions are received, then the lost description can be reconstructed by taking averages along the vertical direction.

In short, the second method can be generalized easily to 2^m-way interleaving, $m>0$. It is flexible because the transformation at the sender does not depend on the loss pattern at the receiver. For this reason, we have adopted this approach in our experiments.

IV. Neural Networks for Compensating Quantization Errors

ORB-DCT described in the last section is used to minimize reconstruction errors when at least one description is lost during transmission and is reconstructed at the receiver. However, when all the descriptions are received at the receiver, perfect reconstruction is still not possible. These errors are due to the quantization of ORB-DCT coefficients at the sender.

In this section, we first characterize the compression loss of MDC when all the descriptions are received at the receiver. Our study shows that compression errors are highly nonlinear and complex and cannot be compensated by a linear process. We then describe in detail our proposed artificial neural-network (ANN) architecture.

A. Quantization Errors in MDC

Since coding gain [12] is proportional to autocorrelation ρ of a video source and ρ is reduced after pixel-based interleaving, we expect the PSNR of an MDC system to be smaller than that of the original system at the same bit rate. This phenomenon is illustrated in Table I that compares ρ and PSNR of a horizontally 2-way deinterleaved stream and the original noninterleaved stream. The experiments were done on two sequences in CIF (352×288) YUV format: *missa* (Miss America) consisting of 150 frames and representing a typical video conferencing sequence with slow head-and-shoulder movements, and *football* consisting of 90 frames featuring a fast-action movie.

In order to improve PSNR of a deinterleaved stream, we need better understanding of the effect of deinterleaving on individual pixels. Fig. 4 illustrates the fluctuations observed in a smooth area and an area with a sharp transition after deinterleaving, using pixel values from a horizontal line of an image in *missa*. (The curves showing the ANN-compensated stream are discussed in Section V-B.) We observe that deinterleaved pixels incur spurious fluctuations in pixel values because different interleaved streams have different quantization losses. Such spurious fluctuations may be removed by suitable filtering.

In the following we propose to train a feedforward multilayer ANN to perform such filtering. We use ANNs rather than adaptive linear filters because compression and decompression are highly nonlinear and complex and ANNs have been proven effective in finding nonlinear mappings between inputs and outputs. The nonlinearity of an ANN lies in the use of a nonlinear activation function in each neuron and one or more hidden layers. Using a learning algorithm, ANNs can be trained to learn the characteristics of and compensate for compression losses.

TABLE I
ADJACENT SAMPLE CORRELATIONS AND PSNRS OF A HORIZONTALLY TWO-WAY DEINTERLEAVED STREAM AND THE ORIGINAL NONINTERLEAVED STREAM

Video	adjacent correlation ρ		PSNR (dB)	
Sequence	original	interleaved	original	interleaved
Missa	0.9964	0.9829	37.48	36.74
football	0.9964	0.9819	31.13	30.16

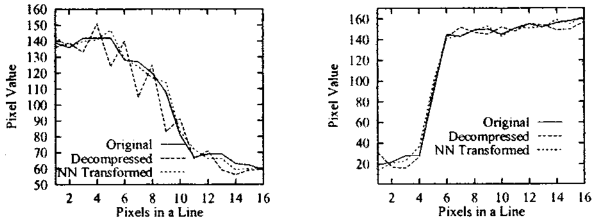

Fig. 4. Comparison of pixel values in the original stream, the deinterleaved stream, and the ANN-compensated stream. Pixel values are taken from a horizontal line in an image of *missa*.

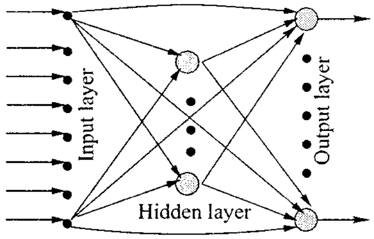

Fig. 5. A three-layer feedforward ANN with full connections between the input-hidden and the hidden-output layers and shortcuts between the input–output layers.

B. Artificial Neural-Network Architecture

Fig. 5 illustrates a three-layer ANN used in our study. It has full interconnections between the input and hidden layers, hidden and output layers, and input and output layers, the latter implementing *shortcuts*. It provides two alternative mappings: a *nonlinear* mapping between the input and output layers if the input to hidden layers have nonzero weights, and a *linear* mapping between the input and output layers if all the input-to-hidden weights are zero and some of the input-to-output weights are nonzero. It adapts to either mapping depending on which mapping leads to smaller training errors.

To facilitate real-time playback, ANN weights are trained in advance by the back-propagation algorithm [13] using a training set derived as follows. Pixels from deinterleaved and decompressed frames serve as inputs, whereas those taken from the original frames (before compression) serve as desired outputs. Since such training tuples do not make *a priori* assumptions about the original image, such as smoothness assumptions commonly used in the literature, we expect that weights learned can capture the common characteristics of compression loss and can generalize well to frames other than those in the training set.

Since decompressed pixel $\hat{g}(i, j)$ is very close to original pixel $g(i, j)$, we have picked suitable initial weights in our experiments to achieve faster convergence in training. This relationship can be realized by initializing the weights between the input and output layers to values close to one and the other weights to values close to zero.

Previous studies [14], [15] show that training will be faster if the activation function of the hidden layer is centered around 0. Hence, we choose *tanh* in the range $[-1, 1]$ as our activation function f_h of the hidden layer:

$$f_h(x) = \frac{2}{1 + \exp(-2x)} - 1; \qquad f_o(x) = x. \tag{19}$$

To allow our ANN realize a linear mapping between the input and output layers, we choose an identity function to implement f_o, the activation function of the output layer.

V. EXPERIMENTAL RESULTS

We have evaluated our ORB-DCT reconstruction and ANN compensation schemes on *missa* and *football* in two scenarios: a synthetic scenario under controlled losses and real Internet tests. In the following, we only show the PSNR of the Y component, the dominant component in human perception.

A. Reconstruction Quality under Controlled Losses

In this section, we show the reconstruction quality under controlled loss scenarios for the ORB-DCT and DCT transforms. (Due to space constraints, only results on 2-way interleaving are shown.) To isolate the effects due to transformations, we first eliminate quantization loss by removing quantization and dequantization in the process.

The left half of Table II compares the reconstruction quality of frames transformed by DCT and by ORB-DCT, assuming that video data is divided into two descriptions along horizontal directions, that only one of the descriptions is received, and that

TABLE II
RECONSTRUCTION QUALITY OF FRAMES IN TERMS OF PSNR (dB) WHEN TRANSFORMED BY ORB-DCT AND DCT AND ONLY ONE OF THE DESCRIPTIONS IS RECEIVED UNDER TWO-DESCRIPTIONS CODING. GAIN IS DEFINED AS THE DIFFERENCE IN PSNR OF ORB-DCT AND THAT OF DCT

Video Sequence	No quantization effects						With quantization effects					
	Odd received			Even received			Odd received			Even received		
	DCT	ORB-DCT	Gain	DCT	ORB-DCT	Gain	DCT	ORB-DCT	Gain	DCT	ORB-DCT	Gain
Missa	39.44	41.31	**1.87**	39.51	41.45	**1.94**	36.20	36.61	**0.41**	36.14	36.59	**0.45**
football	36.05	37.48	**1.43**	36.01	37.47	**1.46**	29.43	29.82	**0.39**	29.40	29.83	**0.43**

quantization effects are ignored. Results along the vertical direction are similar and are not shown. When either the odd-numbered description or the even-numbered description is received, the ORB-DCT transformed frames have consistently better performance (1.4–1.9 dB or 65%–72% of the reconstruction error) than the DCT transformed frames.

Next, we show results on reconstruction quality after including quantization in the right half of Table II. We have modified the H.263 codec from TenetRD (*http://www.nta.no/brukere/DVC/*) to use either ORB-DCT or DCT in the transformation stage. As expected, the quality of frames transformed by ORB-DCT is consistently better than frames transformed by DCT, when half of the descriptions are lost. However, the gain is not as high as cases without quantization. These degradations are caused by the lossy quantization process that made certain irreversible changes to the transformed pixels.

B. Reconstruction Quality with All Descriptions Received

As explained in Section IV-A, the playback quality in MDC is not as good as that in single-description coding when all the descriptions are received without errors. In this section, we show that ANN transformations can effectively compensate for compression losses in MDC.

The ANN used has eight inputs, eight outputs, and three hidden units. The weights were initialized and trained offline using back-propagation based on training tuples taken from the *missa* sequence. Table III compares the performance of the following four cases.

1) The system interleaves the original video sequence into two descriptions, compresses each using DCT, decompresses each, and deinterleaves the two descriptions.
2) The same as in 1), except that the trained ANN is applied after deinterleaving the descriptions. The same ANN is applied to both the *missa* and *football* sequences.
3) The same as in 1), except that DCT is replaced by ORB-DCT in compression.
4) The same as in 3), except that the trained ANN is applied after deinterleaving the descriptions. The same ANN as in 2) is applied.

Table III illustrates an average improvement of 0.3–0.6 dB due to ANN compensations when comparing between Cases a) and b) and between Cases c) and d). Fig. 4 also shows that, after ANN compensations, the pixel values are closer to those in the original frame.

The results indicate that the ANN trained on *missa* generalizes well to *football*. The results of applying an ANN trained from *football* are similar and are not shown.

TABLE III
RECONSTRUCTION QUALITY WHEN ALL THE DESCRIPTIONS ARE CORRECTLY RECEIVED BASED ON TWO-DESCRIPTION CODING. GAIN IS DEFINED AS THE DIFFERENCE IN PSNR OF ORB-DCT&NN AND THAT OF DCT. CASES a) AND c) SHOW THE LOSS INTRODUCED BY COMPRESSION, WHEREAS CASES b) AND d) DEMONSTRATE THE ADDED EFFECT OF COMPENSATION BY ANN

Case	a	b	c	d	Gain in
Video Sequence	DCT	DCT & ANN	ORB -DCT	ORB-DCT & ANN	PSNR (dB)
Missa	36.74	37.06	36.70	37.05	**0.31**
Football	30.16	30.69	30.09	30.67	**0.51**

Note that the quality after ANN compensation is still worse than that of single-description coding. This is the price paid for improved error resilience and robustness.

C. Overview of our Video Streaming Prototype

To evaluate our proposed schemes in real Internet transmissions, we built a video streaming prototype, whose components are shown in Fig. 6. We have discussed our choice of interleaving factors, ORB-DCT transformation, and ANN-based compensation in previous sections. In this section, we present our syntax-based packetization and modifications to the decoder in order to reduce loss propagations in MDC.

In choosing a good packetization strategy, we note that a good strategy should prevent the propagation of errors among packets so that the loss of one packet will not render subsequent packets in an erroneous state.

Our packetization strategy is based on the hierarchical organization of H.263. The top level consists of the *picture* layer divided into a sequence of *groups of blocks* (GOBs), each of which consists of a number of 16×16 macroblocks (MBs). Each MB is then divided into four 8×8 Y blocks, an 8×8 Cr block, and an 8×8 Cb block.

A GOB acts as a basic synchronization point in a coded stream. In most cases, when an error occurs within a GOB, the rest of the GOB will be undecodable, and the decoder has to resume synchronization at the start of the next GOB. As a result, we set our packet boundary corresponding to that of GOBs.

To minimize loss propagations in MDC, we modified the original H.263 decoder as follows. Instead of using the last completely received frame as its reference to an intercoded frame when a reference frame was lost during transmission, we fed the reconstructed subframe back to the motion-compensation loop of the decoder. Our reasons are that lost subframes can be reconstructed when a subset of descriptions are received, and that reconstructed subframes are closer to lost ones than subframes in the last completely received frame due to motion in the video sequence.

D. Tests on the Internet

Using the prototype, we have tested the following strategies for transmissions on the Internet.

- Strategy S1: average reconstruction of frames transformed by ORB-DCT, if any of the interleaved descriptions is lost, or ANN-based compensation if all the descriptions are received;
- Strategy S2: Average reconstruction of frames transformed by DCT;
- Strategy S3: Decoding of frames that are single-description coded.

Among the three strategies, S1 and S2 reflect MDC with error concealment, whereas S3 is the original codec. For a fair comparison under the same traffic conditions, we did trace-based simulations by applying each of the strategies on the same trace of packets collected in real Internet transmissions (see Section II).

Our experiments to apply traffic traces consist of a sender process and a receiver process. The sender process was responsible for compressing and packetizing video frames, and mapping packet losses to GOB losses of each frame. The number of descriptions (two or four) was determined periodically every 0.5 s at the sender according to feedback information on GOB losses of frames from the receiver. In our simulations, we assume that the receiver collected GOB loss information every 0.5 s before sending the information to the sender, and that the network delay was constant at 0.5 s. The receiver process was in charge of decompressing coded streams, deinterleaving them, and performing reconstruction. For every GOB in each frame, any missing information was reconstructed by average interpolation using adjacent pixels. The reconstructed frame was sent back to the decoder as a reference for future intercoded frames. If the entire GOB was lost, it was reconstructed by copying the corresponding GOB from the last received frame.

Fig. 7 compares the reconstruction quality and the corresponding loss rates over a 24-h period for the Champaign-Berkeley connection. Note that the loss rates, although not high in general, are different for *missa* and *football*. This happens because the two coded bit streams were mapped to different number of packets. For both sequences, ORB-DCT, when coupled with ANN compensations, performs the best at all times. For *missa* (resp. *football*), the average PSNR over a 24-h period is 36.41 dB (resp. 30.08 dB) if reconstruction were based on the ORB-DCT stream, and is 35.81 dB (resp. 29.55 dB) if reconstruction were based on the DCT stream.

Fig. 8 shows the corresponding results for the Champaign-China connection. The loss rates of this connection range between 10% and 50% in most cases. As expected, ORB-DCT, together with ANN compensations, yield the best playback quality at all times. For *missa* (resp. *football*), the average PSNR is 34.20 dB (resp. 27.73 dB) for the ORB-DCT stream, and 33.18 dB (resp. 26.87 dB) for the original DCT stream.

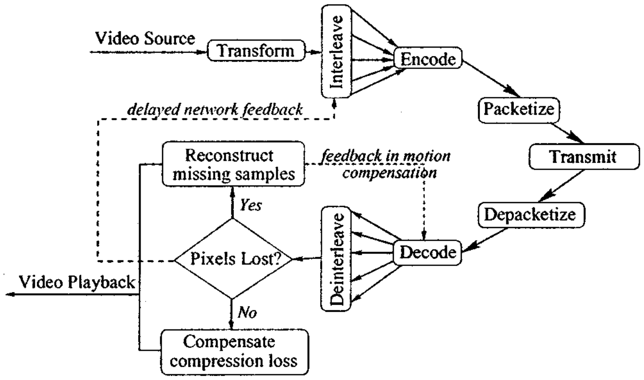

Fig. 6. Components of our video streaming prototype.

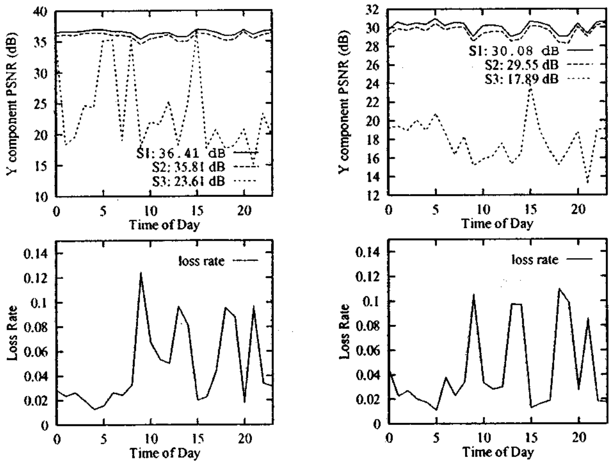

Fig. 7. Comparisons of reconstruction quality and the corresponding loss rates over a 24-h period for the Champaign-Berkeley connection.

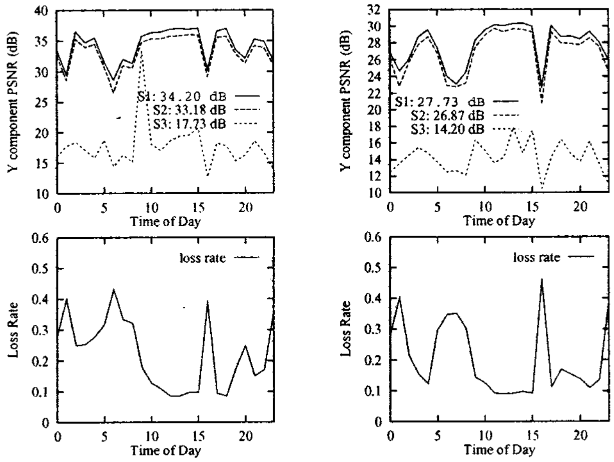

Fig. 8. Comparisons of reconstruction quality and the corresponding loss rates over a 24-h period for the Champaign-China connection.

The above graphs show that the playback quality of single-description coded streams is very poor, although they have high PSNRs in an error-free environment (see Table I). Therefore, single-description coded streams using the original H.263 codec are not suitable for transmission in a lossy environment like the Internet.

It is interesting to note that under real loss situations, the gains of S1 and S2 for both *missa* and *football* are higher than those in the synthetic scenarios in Sections V-A and V-B. This is not surprising because in real tests, we always fed reconstructed frames that were lost back to the motion-compensation loop, and the improvement of reconstruction quality due to these feedbacks accrued as the video sequence was played. In contrast, feedbacks were not possible in synthetic scenarios, since one or more streams were consistently lost.

The fraction of interleaved sets that were correctly received (those for which ANN compensations were applied), can roughly be calculated as one minus the loss rate. Our experimental results show that, for those interleaved sets that were received correctly, ANN compensations were performed on between 50% to 98% of the interleaved sets. These indicate that ORB-DCT performed at the sender and ANN compensations performed at the receiver are equally-important complementary methods that work jointly in improving playback quality.

VI. Conclusions and Future Work

We have presented in this paper multidescription methods to cope with information loss in supporting real-time video streaming over the Internet. First, we have derived an optimized reconstruction-based DCT to minimize distortions if some of the interleaved descriptions were lost, and the missing information is reconstructed using simple interpolation. Second, we have proposed an ANN method to compensate for quantization loss when all the descriptions are received. Experimental results show that the two methods complement each other in improving playback quality. Our future work will be focused on extending the methods to applications under rate constraints and on fast implementations of ORB-DCT.

References

[1] E. W. Biersack, "Performance evaluation of forward error correction in ATM networks," *Comput. Commun. Rev.*, vol. 22, no. 4, pp. 248–257, Oct. 1992.

[2] J. C. Bolot and A. Vega-Garcia, "Control mechanisms for packet audio in the Internet," in *Proc. IEEE Infocom'96*, San Francisco, CA, Apr. 1996, pp. 232–239.

[3] J. M. Danskin, G. M. Davis, and X. Song, "Fast lossy Internet image transmission," in *ACM Multimedia*, San Francisco, CA, Nov. 1995.

[4] I. Rhee, "Error control techniques for interactive low-bit rate video transmission over the Internet," in *Proc. SIGCOMM*, 1998.

[5] W. Kwok and H. Sun, "Multi-directional interpolation for spatial error concealment," *IEEE Trans. Consumer Electron.*, vol. 39, pp. 455–460, Aug. 1993.

[6] W. Zeng and B. Liu, "Geometric structure based directional filtering for error concealingment in image/video transmission," in *Proc. SPIE Wireless Data Transmission at Information Systsmes/Phontonics East'95*, Oct. 1995, pp. 145–156.

[7] M. Normura, T. Fujii, and N. Ohta, "Layered packet-loss protection for variable rate coding using DCT," in *Proc. of Int. Workshop on Packet Video*, Sept. 1988.

[8] V. A. Vaishampayan, "Design of multiple description scalar quantizer," *IEEE Trans. Inform. Theory*, vol. 39, pp. 821–834, May 1993.

[9] Y. Wang, M. T. Orchard, and A. R. Reibman, "Multiple description image coding for noisy channels by pairing transform coefficients," in *Proc. IEEE First Workshop Multimedia Signal Processing*, June 1997, pp. 419–424.

[10] Y. Wang and Q. Zhu, "Error control and concealment for video communications: A review," *Proc. IEEE*, vol. 86, pp. 974–997, May 1998.

[11] L. Rade and B. Westergren, *Mathematics Handbook for Science and Engineering*. Cambridge, MA: Studentlitteratur Birkhauser, 1995.

[12] N. Jayant and P. Noll, *Digital Coding of Waveforms: Principles and Applications to Speech and Video*. Englewood Cliffs, NJ: Prentice-Hall, 1984.

[13] S. Haykin, *Neural Networks: A Comprehensive Foundation*, 2nd ed. Englewood Cliffs, NJ: Prentice-Hall, 1999.

[14] M. Brown, P. C. An, C. J. Harris, and H. Wang, "How biased is your multi-layer perceptron?," in *World Congr. Neural Networks*, 1993, pp. 507–511.

[15] Y. LeCun, I. Kanter, and S. A. Solla, "Eigenvalues of covariance matrices: Application to neural network learning," *Phys. Rev. Lett.*, vol. 66, no. 18, pp. 2396–2399, 1991.

Xiao Su received the B.S. degree in computer science and engineering from Zhejiang University, Hangzhou, China, in 1994, and the M.S. degree in computer science from the University of Illinois at Urbana-Champaign (UIUC) in 1997. She is currently pursuing the Ph.D. degree in computer science at UIUC. Her research interests include video coding and transmission, computer networking and wireless communications, with emphasis on error concealment schemes for robust video streaming.

Benjamin W. Wah (SM'85–F'91) received his Ph.D. degree in computer science from the University of California, Berkeley, in 1979.

He is currently the Robert T. Chien Professor of Electrical and Computer Engineering, and a Research Professor of the Coordinated Science Laboratory and the Beckman Institute, University of Illinois at Urbana-Champaign. Previously, he had served on the faculty of Purdue University, West Lafayette, IN (1979–1985), as a Program Director at the National Science Foundation (1988–1989), as Fujitsu Visiting Chair Professor of Intelligence Engineering, University of Tokyo, Japan (1992), and as McKay Visiting Professor of Electrical Engineering and Computer Science, University of California, Berkeley (1994). His current research interests are in nonlinear search and optimization, knowledge engineering, multimedia signal processing, and parallel and distributed processing. He was the Honorary Editor-in-Chief of *Knowledge and Information Systems*. He currently serves on the editorial boards of *Information Sciences*, *International Journal on Artificial Intelligence Tools*, *Journal of VLSI Signal Processing*, *Parallel Algorithms and Applications*, and *Neural Processing Letters*.

Dr. Wah was the Editor-in-Chief of the IEEE Transactions on Knowledge and Data Engineering between 1993–1996.He had chaired a number of international conferences and was the International Program Committee Chair of the IFIP World Congress in 2000. He has served in the IEEE in various capacities and is currently the 2001 President of the IEEE Computer Society. He is a Fellow of the Society for Design and Process Science. In 1989, he was awarded a University Scholar of the University of Illinois; in 1998, he received the IEEE Computer Society Technical Achievement Award; and in 2000, the IEEE Millennium Medal.