Three-Dimensional Subband Coding Techniques for Wireless Video Communications

Hong Man, *Member, IEEE*, Ricardo L. de Queiroz, *Senior Member, IEEE*, and Mark J. T. Smith, *Fellow, IEEE*

***Abstract*—This paper presents a new 3-D subband coding framework that is able to achieve a good balance between high compression performance and channel error resilience. Various data transform methods for the decorrelation of video were examined and compared, including subband filtering, discrete cosine transforms, discrete wavelet transforms, and lapped transforms. The coding stage of the algorithm is based on a generalized adaptive quantization framework, which is applied to the 3-D transformed coefficients. More specifically, it features a simple coding structure based on quadtree coding and lattice vector quantization techniques. In typical applications, good performance at high compression ratios is obtained often without entropy coding. Furthermore, because temporal decorrelation is absorbed by the transform, traditional motion compensated prediction becomes not necessary, which results in a significant computational advantage over standard video coders. The error-resilience feature is achieved through classifying the compressed data streams into separated sub-streams with different error sensitivity levels. This enables a good adaptation to different channel models according to their noise statistics and error-protection protocols. Experimental results have shown that the subband video coder is able to achieve highly competitive performance relative to MPEG-2 in both noiseless and noisy environments. Furthermore, lapped transforms are shown experimentally to outperform the other transforms in the 3-D subband environment. The subband coding framework provides a practical solution for video communications over wireless channels, where efficiency, error resilience, and computational simplicity are vital in providing superior quality of service.**

***Index Terms*—3-D subband coding, adaptive quantization, error resilient, lapped transform, video coding, wavelet transform.**

I. Introduction

MAJOR issues associated with wireless video applications include limited bandwidth, channel noise, and computation. Data rates for high resolution digital video are generally very large, and therefore are typically compressed before transmission. Wireless transmission may induce high bit error rates (BERs), which can be problematic for compression algorithms. At the same time, wireless users more and more rely on portable computing devices, which have only limited computing power for video processing and transmission. In a wireless video communication system, it is particularly desirable to achieve high compression and computation efficiency while maintaining an error-resilient coding structure.

Current high-definition television (HDTV) broadcasting standards (`ATSC A/53A`, `ITU-R BO.1211`, `ITU-T J.83`, `ETS 300 421`, etc.) normally consist of a standard video coder (e.g., MPEG-2) concatenated with a channel coder. This has been a typical approach adopted in many video applications involving wireless media. Most standard video coding methods [1], [2], namely `ISO MPEG-(1,2,4)` and the `ITU H.(261,263,26L)`, have very similar coding structures. They generally contain a discrete cosine transform (DCT) [3] coding approach in the spatial domain and a motion-compensated predictive (MCP) coding approach in the temporal domain. Over the years of standards development, compression efficiency has been improved, although at the expense of extra computation complexity. However, little attention has been paid to error resilience.

In this paper, we study the effectiveness of 3-D transform-based video coding methods with emphasis on compression, computation, and error resilience. Although many 3-D transform coding methods using DCTs, general subband decompositions, and discrete wavelet transforms (DWTs) [4]–[8] have been proposed over the last decade, they have not yet found major application because of their high computational load, and increased memory requirements, for a moderate compression performance improvement. The objective of this work is to demonstrate the advantage of 3-D transforms over MCP in certain video applications.

Fig. 1 shows a block diagram of the encoding/decoding structure of the proposed video coder. It consists of two sequential stages: the 3-D transform for data decorrelation, and the quantization/coding stage which is applied to the transformed coefficients. Several 3-D transform methods have been implemented in this work using the DCT, the uniform subband decomposition (USB), the DWT, and two lapped transforms (LTs) known as the lapped orthogonal transform (LOT) [9], [10] and the lapped bi-orthogonal transform (LBT) [11], [12]. We have also applied an adaptive quantization technique in the second stage. The implementations were engineered to exploit the computational efficiencies of the 3-D transforms in combination with the error-resilience properties of the adaptive quantization technique in order to maximize the overall performance of the video coding system.

The remainder of this paper is organized as follows. Section II provides a brief description of the transform methods examined in this study, while Section III introduces the novel adap-

Manuscript received August 15, 2001; revised April 15, 2002. This work was supported by the New Jersey Commission on Science and Technology and by the U.S. Army Research Office under Grant DAAH04-96-1-0161.

H. Man is with the School of Electrical and Computer Engineering, Stevens Institute of Technology, Hoboken, NJ 07030 USA (e-mail: hman@stevens-tech.edu).

R. L. de Queiroz is with Xerox Corporation, M/S 128-27E, Webster, NY 14580 USA (e-mail: queiroz@ieee.org).

M. J. T. Smith is with the School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA 30332-0250 USA (e-mail: mark.smith@carnegie.gatech.edu).

Publisher Item Identifier 10.1109/TCSVT.2002.800316.

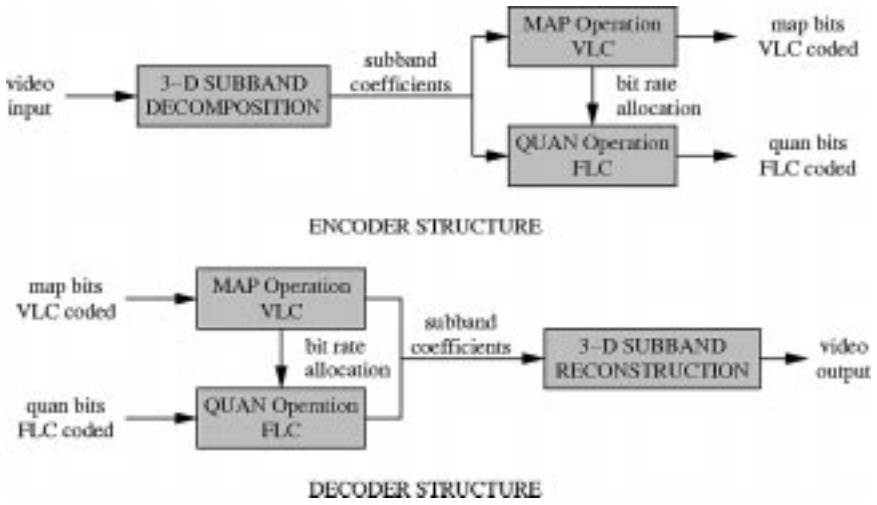

Fig. 1. Encoding/decoding structure of the subband video coding algorithm.

tive quantization and coding technique. Section IV discusses the error-resilience features of the video coder, and Section V presents the simulation results along with a brief analysis. The paper ends with some conclusions in Section VI.

II. 3-D Data Transform

Transforms are generally used in compression systems to achieve data decorrelation and energy compaction. Popular transforms of this type (e.g., DCT, USB, DWT, LOT, and LBT) commonly have a frequency selectivity property, i.e., they partition the input data into a series of frequency bands that collectively cover the entire spectrum. If the input is a natural image or video signal, most of the energy will be packed into the low frequency subbands after the decomposition. At the same time, the transformed coefficients become less correlated than the original data samples. These features can help the coding stage achieve very high compression efficiency.

The DCT has been widely adopted in the image and video coding standards because it provides excellent energy compaction for images and can be implemented with fast algorithms. For a 1-D signal broken into a sequence of M-sample blocks, the forward and inverse M-point DCT can be expressed as

$$\mathbf{y}_m = \mathbf{D}\mathbf{x}_m \quad \text{and} \quad \mathbf{x}_m = \mathbf{D}^T\mathbf{y}_m \tag{1}$$

where $\mathbf{x}_m$ is the mth input block and $\mathbf{y}_m$ is its transformed version. The $M \times M$ transform matrix $\mathbf{D}$ has entries

$$d_{ij} = \sqrt{\frac{2}{M}}\, k_i \cos\left(\frac{(2j+1)i\pi}{2M}\right) \tag{2}$$

where $k_0 = 1/\sqrt{2}$ and $k_i = 1$ for $1 \le i \le M-1$.

The DCT is usually implemented on a block-by-block basis, as shown in Fig. 2. Likewise, the coding process operates mostly within each nonoverlapping block. This approach leads to "blocking artifacts" at high compression ratios. In such a situation, the edges of the data blocks become visible and the appearance of the picture frames becomes noticeably degraded.

The LOT was introduced to overcome this problem by performing the transform using data collected via a sliding window

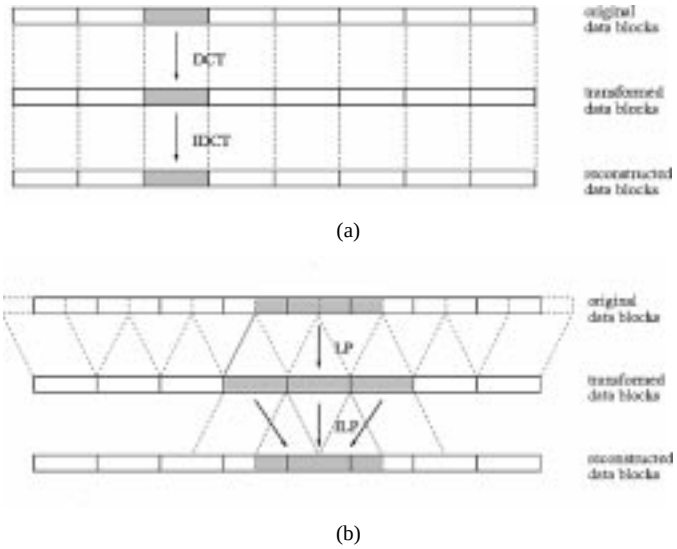

Fig. 2. Operating procedures of the DCT and the lapped transforms. (a) DCT is performed with each data block. (b) Lapped transform (LOT or LBT) is performed on overlapped data blocks.

applied to the signal. The adjacent input data blocks overlap with each other, as depicted in Fig. 2. Recognizing there is freedom in selecting the amount of overlap, for simplicity in this paper we will focus on LTs whose overlap factor is 2, i.e., the input data block as well as the transform basis have length $L = 2M$. In this case, the forward and inverse LT matrices are represented by the $M \times 2M$ matrices $\mathbf{P}$ and $\mathbf{Q}$, respectively, such that

$$\mathbf{P} = [\mathbf{P_0} \quad \mathbf{P_1}] \quad \text{and} \quad \mathbf{Q} = [\mathbf{Q_0} \quad \mathbf{Q_1}]. \tag{3}$$

Thus, the forward and inverse M-point LT can be expressed as

$$\mathbf{y}_m = \mathbf{P_0}\mathbf{x}_{m-1} + \mathbf{P_1}\mathbf{x}_m \quad \text{and} \quad \mathbf{x}_m = \mathbf{Q_1^T}\mathbf{y}_{m-1} + \mathbf{Q_0^T}\mathbf{y}_m. \tag{4}$$

In the popular LOT case, the LT matrix is constructed as

$$\mathbf{P}_{\text{LOT}} = \begin{bmatrix} \mathbf{I}_{M/2} & \mathbf{0} \\ \mathbf{0} & \mathbf{V}_R \end{bmatrix} \begin{bmatrix} \mathbf{D}_e - \mathbf{D}_o & \mathbf{J}_{M/2}(\mathbf{D}_e - \mathbf{D}_o) \\ \mathbf{D}_e - \mathbf{D}_o & -\mathbf{J}_{M/2}(\mathbf{D}_e - \mathbf{D}_o) \end{bmatrix} \tag{5}$$

where

- $\mathbf{D}_e$ $M/2 \times M$ matrix with the even-symmetric basis functions of the DCT matrix $\mathbf{D}$;
- $\mathbf{D}_o$ matrix of the same size with the odd-symmetric basis functions;
- $\mathbf{I}_{M/2}$ identity matrix;
- $\mathbf{J}_{M/2}$ reversing or counter-identity matrix, i.e., the only nonzero elements are 1s along the secondary (reverse) diagonal.
- $\mathbf{V}_R$ $M/2 \times M/2$ orthogonal matrix, which can be designed to achieve good coding gain and to be efficiently implemented [9], [10], [13].

Both matrices $\mathbf{J}_{M/2}$ and $\mathbf{V}_R$ have size $M/2 \times M/2$. The LOT is said to be orthogonal so that $\mathbf{Q}_{\mathrm{LOT}} = \mathbf{P}_{\mathrm{LOT}}$.

With overlapping bases, the amplitude transitions across block edges become smoother than in the DCT case. At the same time, because the forward transform takes more input samples to produce the same amount of output coefficients, it can improve the data decorrelation capability as compared to the DCT.

Although the LOT can significantly reduce the "blocking artifacts," it may still produce some visible block edges. A small modification of the LOT yields another transform: the LBT [11], which is aimed at minimizing the edge discontinuity. With this modification, the bases are no longer orthogonal, and the forward and inverse transforms become different. For $L = 2M$, an LBT forward transform matrix $\mathbf{P}_{\mathrm{LBT}}$ is given by

$$\mathbf{P}_{\mathrm{LBT}} = \begin{bmatrix} \mathbf{I}_{M/2} & \mathbf{0} \\ \mathbf{0} & \mathbf{V}_R \end{bmatrix} \begin{bmatrix} \mathbf{D}_e - \boldsymbol{\Upsilon}\mathbf{D}_o & \mathbf{J}_{M/2}(\mathbf{D}_e - \boldsymbol{\Upsilon}\mathbf{D}_o) \\ \mathbf{D}_e - \boldsymbol{\Upsilon}\mathbf{D}_o & -\mathbf{J}_{M/2}(\mathbf{D}_e - \boldsymbol{\Upsilon}\mathbf{D}_o) \end{bmatrix} \tag{6}$$

and the corresponding inverse transform matrix is

$$\mathbf{Q}_{\mathrm{LBT}} = \begin{bmatrix} \mathbf{I}_{M/2} & \mathbf{0} \\ \mathbf{0} & \mathbf{V}_R \end{bmatrix} \cdot \begin{bmatrix} \mathbf{D}_e - \boldsymbol{\Upsilon}^{-1}\mathbf{D}_o & \mathbf{J}_{M/2}(\mathbf{D}_e - \boldsymbol{\Upsilon}^{-1}\mathbf{D}_o) \\ \mathbf{D}_e - \boldsymbol{\Upsilon}^{-1}\mathbf{D}_o & -\mathbf{J}_{M/2}(\mathbf{D}_e - \boldsymbol{\Upsilon}^{-1}\mathbf{D}_o) \end{bmatrix} \tag{7}$$

where $\boldsymbol{\Upsilon} = \mathrm{diag}\{\sqrt{2}, 1, \ldots, 1\}$ and $\boldsymbol{\Upsilon}^{-1} = \mathrm{diag}\{1/\sqrt{2}, 1, \ldots, 1\}$. The LBT can be implemented in the same way as the LOT, and it generally achieves better energy compaction performance than the LOT. Further improvement can also be achieved by using the generalized LBT (GLBT) [12], [14], [13] in which

$$\mathbf{P}_{\mathrm{LBT}} = \begin{bmatrix} \mathbf{U} & \mathbf{0} \\ \mathbf{0} & \mathbf{V} \end{bmatrix} \begin{bmatrix} \mathbf{D}_e - \mathbf{D}_o & \mathbf{J}_{M/2}(\mathbf{D}_e - \mathbf{D}_o) \\ \mathbf{D}_e - \mathbf{D}_o & -\mathbf{J}_{M/2}(\mathbf{D}_e - \mathbf{D}_o) \end{bmatrix} \tag{8}$$

and the corresponding inverse lapped transform matrix is

$$\mathbf{Q}_{\mathrm{LBT}} = \begin{bmatrix} \mathbf{U}^{-1} & \mathbf{0} \\ \mathbf{0} & \mathbf{V}^{-1} \end{bmatrix} \begin{bmatrix} \mathbf{D}_e - \mathbf{D}_o & \mathbf{J}_{M/2}(\mathbf{D}_e - \mathbf{D}_o) \\ \mathbf{D}_e - \mathbf{D}_o & -\mathbf{J}_{M/2}(\mathbf{D}_e - \mathbf{D}_o) \end{bmatrix} \tag{9}$$

where $\mathbf{U}$ and $\mathbf{V}$ can be any $M/2 \times M/2$ nonsingular matrix. The $\mathbf{U}$ and $\mathbf{V}$ represent the degrees of freedom of the GLBT.

From an implementation point of view, the subband decomposition and the DWT represent a different class of transform. These transforms are usually achieved through successive filtering of the original data sequence, as shown in Fig. 3. At each transform level, a typical subband decomposition or DWT applies a low-pass filter (LPF) and a high-pass filter (HPF) separately to the whole input sequence. The two output sequences are down-sampled by two so that their combined length will be the same as the original sequence length. This procedure is repeated for both the low frequency and the high frequency subsequences in the USB, but only to the low frequency subsequences in the DWT. By successive applications of this filtering operation, the original sequence can be transformed into a series of subsequences, each representing a specific frequency subband. The inverse transform is obtained through the synthesis procedure where the subband sequences are up-sampled by two and low-pass or high-pass filtered at each transform level.

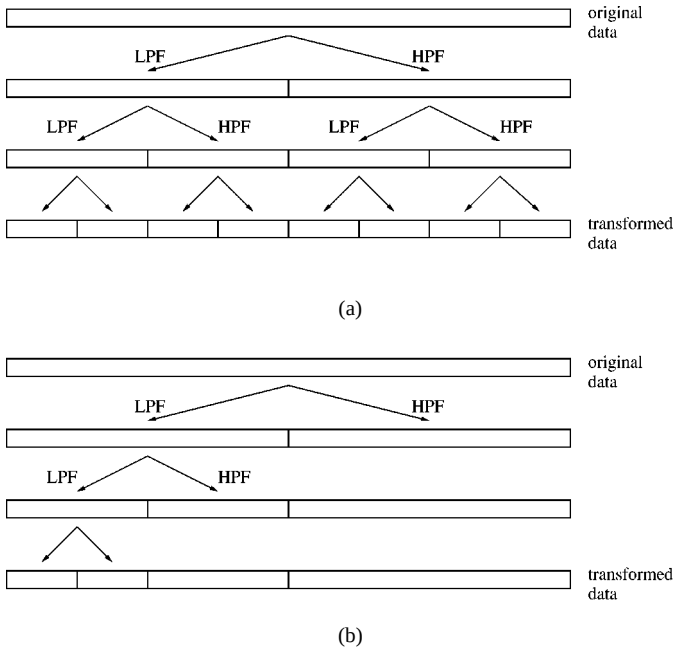

Fig. 3. Operating procedures of the USB and DWT transforms with uniform and pyramid decompositions. (a) Three-level uniform wavelet transform. (b) Three-level pyramid wavelet transform.

It is clear that an M-point DCT, LOT, or LBT can be viewed as an M-band uniform subband decomposition. In this case each subband contains exactly the same number of transformed coefficients, and they are the same frequency components collected from all the transformed data blocks.

These 1-D transform methods can be easily extended to 2-D images by successively performing the 1-D transform on each of the rows, and then on each of the resulting columns. For 3-D video, an extra 1-D transform can be applied in the temporal direction.

In terms of computation, the LBT and LOT are more complex than the DCT but simpler than the DWT and subband decomposition. If we use well-known conservative implementations, one can implement an 8-point DCT using **40** operations (additions or multiplications). An 8-point LOT or LBT can be implemented with **82** or **83** operations. The generalized version of the LBT can be implemented with slightly more than **100** operations. As a reference, the DWT using the biorthogonal 9/7 filters [15] can be implemented using **161** operations for a 3-stage pyramid decomposition. Much more can be said about implementations and techniques for reducing computation for

all these transform methods. We will not discuss those efficient implementations further, except to point out that both the subband/DWT and LOT/LBT can be implemented efficiently using *ladder filters* [16]–[18]. The ladder stages for filter banks can be obtained through the Smith decomposition factors [19]. The best factorization for LTs is still under study.

III. Adaptive Quantization

The proposed video coder is an extension of the subband domain adaptive quantization technique introduced in [20], [21]. Adaptive quantization schemes have demonstrated extraordinary compression efficiencies in subband coding of images and video sequences [22]–[26], [5], [6]. Unlike conventional subband domain quantization approaches, which normally allocate a fixed bit rate for each of the subbands, adaptive quantization techniques are able to apply different quantization rates to different coefficients inside each subband with little or no geometric constraints. This is accomplished by identifying **significant** (or **important**) coefficients inside each subband and performing quantization at bit rates according to the importance of the coefficient. The importance of a certain coefficient is measured by its contribution to the overall quantization fidelity. The quantization rate distribution is then encoded implicitly or explicitly and transmitted to the decoder as a part of the compressed data stream.

In the proposed video coding structure, all the subband coefficients are grouped into pixel blocks, which we call quantization *units*, after which quantization/coding is applied to each unit in relation to its magnitude. The magnitude of each *unit* is represented by its L_2 norm. In order to achieve an embedded bit stream, the quantization/coding stage is carried out in layers. A scale-down factor is used at each successive layer to control the ratio of maximum magnitudes between the subject layer and the previous layer. In our implementations, the scale-down factor is set to one half, therefore the layers essentially become bit planes in magnitude. The significance of a *unit* is determined by comparing its magnitude with a threshold that is associated with the subject layer. The threshold will also become the upper bound of the maximum magnitude of the next layer. For quantization/coding of the coefficients, two operations are performed during each layer pass: the **MAP** operation and the **QUAN** operation.

The **MAP** operation identifies the significant *units* at each layer and codes their locations through a 3-D **quadtree** representation. It produces a sequence of bits in the output bit stream that we call the **map** bits. Fig. 4 illustrates the coding process of the 3-D quadtree structure. For each subband, the significant *units*, or the *units* whose magnitudes are larger than the current threshold value, are coded following this 3-D quadtree structure. If a subband is found to contain one or more significant *units*, the symbol "**1**" is produced, and this subband will be evenly split into $(2 \times 2 \times 2)$ regions. Each such region corresponds to a 3-D quadtree branch. If any of these regions contains one or more significant *units*, the symbol "**1**" is appended, and the region will be further split into $(2 \times 2 \times 2)$ sub-regions (or sub-branches). Otherwise, the symbol "**0**" is appended, and no further tests will be performed on the subject branch. This process continues in a recursive way, until each of the leaf branches corresponds to a region that does not contain any significant *unit* or contains only one *unit*. Once a *unit* is labeled as significant, it will remain significant during the following layer passes. The quadtree structure obtained at the current layer is used for initialization at the following layer, where a new threshold value is calculated by scaling down the current threshold value.

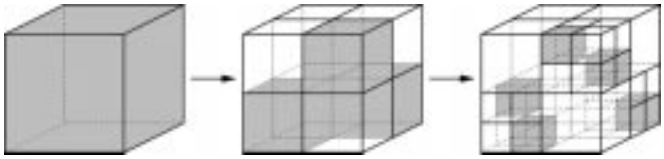

Fig. 4. Example of a 3-D quadtree coding procedure, where shaded regions represent quadtree branches containing one or more significant *unit(s)*.

For each layer pass, the above procedure is equivalent to the recursive application of the following test to each leaf branch which corresponds to a region that contains one or more significant *units*.

1) If a quadtree branch corresponds to a region with one or more significant *units*, then:
   a) Append "**1**" to the **map** bit portion of the data stream.
   b) If the subject region has more than one *unit* (number of units should be a multiple of 8), then:
      — decompose this region into $(2 \times 2 \times 2)$ sub-regions;
      — apply this test from the beginning to each of the eight resulting sub-regions;

   Else:
      — quantize the single *unit* at the same layer. This test will henceforth not be applied to such a *unit* in all the following layers.
2) Else:
   1) Append "**0**" to the **map** bit portion of the data stream.
   2) Applications of this test to this branch end at the subject layer, and the test will be applied again to this branch at the next layer.

Optional arithmetic coding can be used for further compression of the decision symbols from the quadtree coding procedure.

Once a significant *unit* is identified, quantization is performed using a multistage residual lattice vector quantizer (LVQ). We denote this stage as the **QUAN** operation. All the quantization indices are stored in the **quan** bit portion of the output data stream.

An LVQ codebook contains highly structured lattice points that effectively span the signal space. It does not require any training and can be implemented efficiently without codeword storage. Two different LVQs have been designed for vectors with size of $(2 \times 2 \times 1)$ at different quantization stages. Both are derived from the root lattice Z_4. A 6 bits/vector sphere truncated LVQ is used for the first stage quantization, which has the ability to achieve sufficient shape gain. It is obtained by truncating the root lattice at the radius of 3, which produces an LVQ with three energy shells and 64 symbols. The energy measure

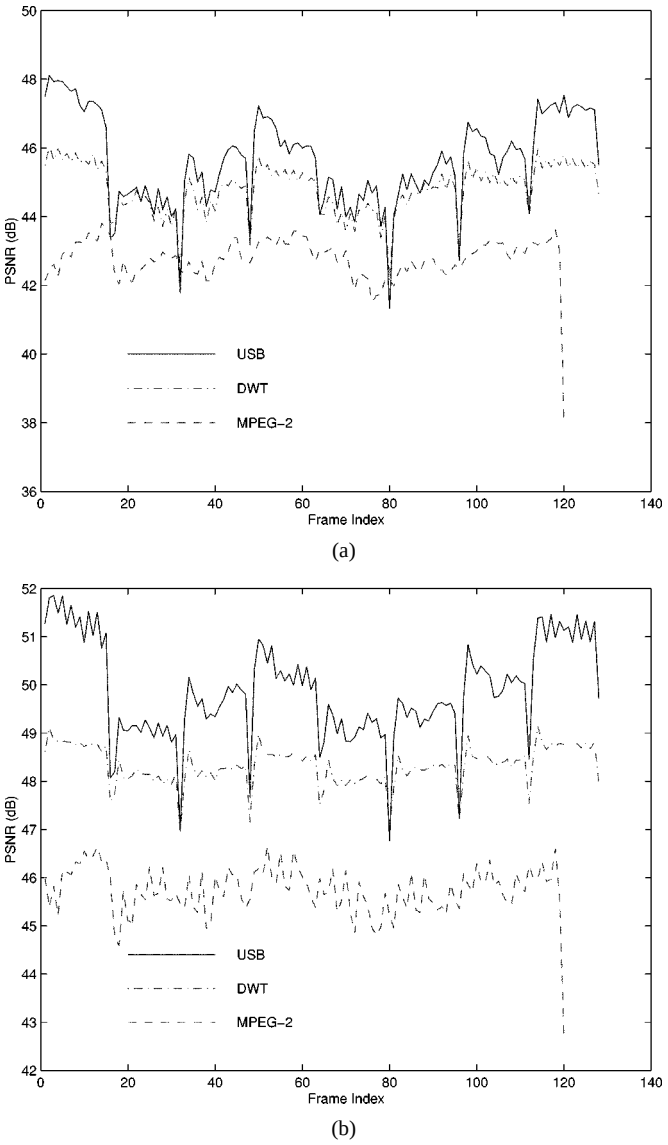

Fig. 5. PSNR performance of the 3-D subband video coders USB, DWT and the MPEG-2 coder for "Akiyo" sequence in noiseless environment. (a) 0.25 bpp or 760320 bps. (b) 0.50 bpp or 1520640 bps.

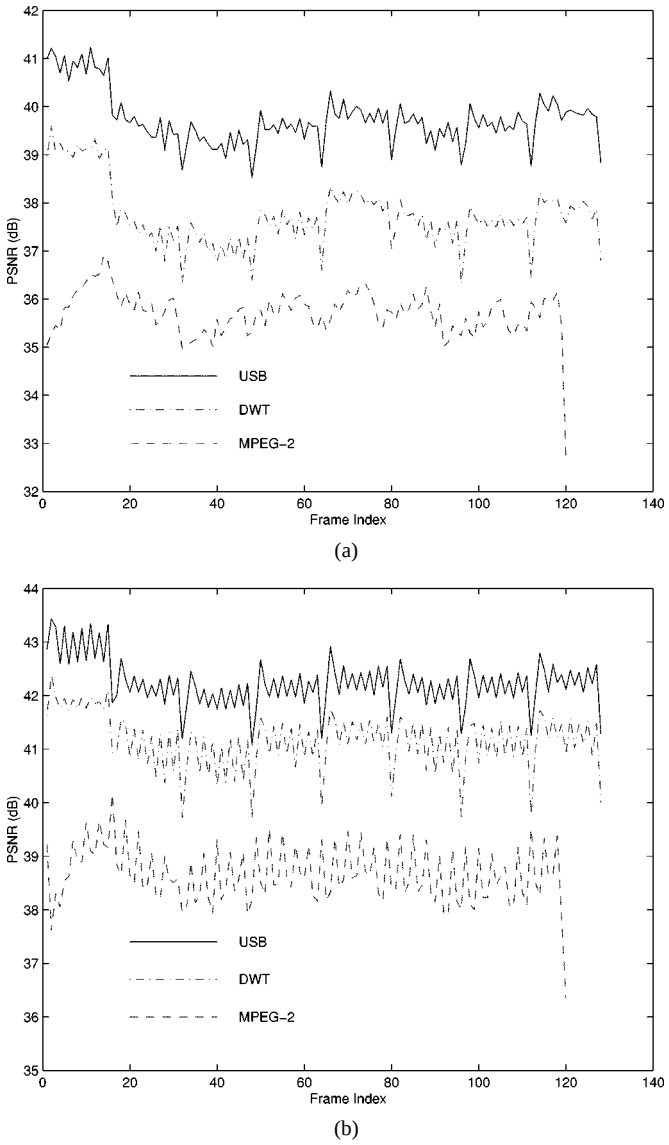

Fig. 6. PSNR performance of the 3-D subband video coders USB, DWT and the MPEG-2 coder for "Hall Monitor" sequence in noiseless environment. (a) 0.25 bpp or 760320 bps. (b) 0.50 bpp or 1520640 bps.

for sphere truncation is the L_2 norm. A 4 bits/vector cubic LVQ is applied to all the successive refinement stages, which guarantees the convergence of all the quantized approximations. Before truncation, the lattice is shifted by the vector (1/2, 1/2, 1/2, 1/2). The truncation radius is set to 1 (in L_∞) so that 16 codewords are obtained for each vector.

For progressive coding, each quantization stage is consistent with each layer pass in the **MAP** operation, and the residual errors of all the quantization *units* are bounded by the threshold of the last layer. In order to obtain channel error resilience, the LVQ quantization indices (i.e., the **quan** bits) are fixed length coded (FLC) as opposed to entropy coded or variable-length coded (VLC).

IV. Error Resilience

Most current video coding algorithms rely heavily on some VLC techniques to achieve high compression performance. However, every VLC technique has error-propagation problems in the case of data transmission over noisy channels. A single bit error may cause a loss of synchronization between the encoder and the decoder. Therefore some digital video broadcasting standards have specified bit error rate (BER) requirements within 10^{-10} to 10^{-11} for the video source coder such as MPEG-2 to operate properly. This BER range is referred to as "quasi-error-free" (QEF), which roughly corresponds to less than one error per transmission hour.

By separating the **map** bits and the **quan** bits into disjoint data sections in the output data stream, the new coder can reduce the possibility of error propagation. Since the **quan** bits are fixed rate coded, only errors occurring in the **map** bits can cause error propagation. Error resilience is achieved because the sizes of the **map** portions are inherently small. For most video sequences at commonly used bit rates, the VLC coded **map** portion only accounts for **20**$\sim$**30**$\%$ of the total bit stream, in contrast to the **100**$\%$ VLC coded data streams produced by MPEG-2 and many other video coders. Thus, our data streams clearly

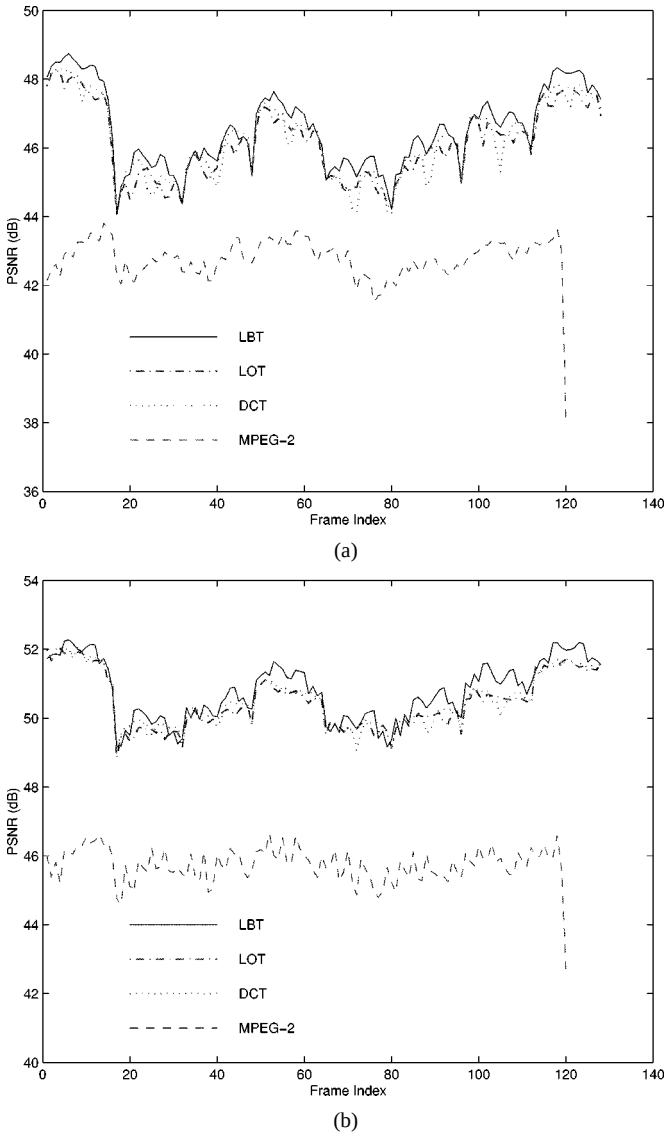

Fig. 7. PSNR performance of the 3-D subband video coders LBT, LOT, DCT and the MPEG-2 coder for "Akiyo" sequence in a noiseless environment. (a) 0.25 bpp or 760320 bps. (b) 0.50 bpp or 1520640 bps.

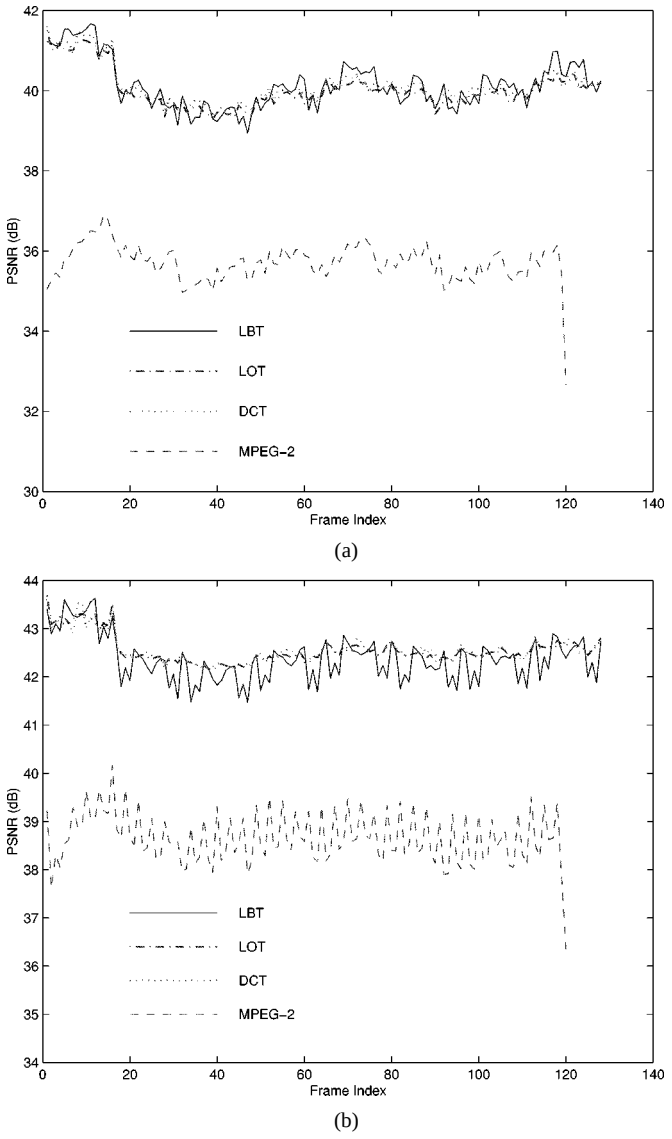

Fig. 8. PSNR performance of the 3-D subband video coders LBT, LOT, DCT and the MPEG-2 coder for "Hall Monitor" sequence in a noiseless environment. (a) 0.25 bpp or 760320 bps. (b) 0.50 bpp or 1520640 bps.

have less chance of manifesting propagational errors. This advantage becomes more obvious when channel BERs are relatively low.

In progressive transmission mode, our coded data streams are organized by coding/quantization layers. At each layer, the expected consistency between the rate distribution in the **map** and the actual data length in the **quan** portion can serve as an inherent error detection for the **map** decoding. After the decoder retrieves a **map** bit portion, it is able to calculate the exact number of bits in the corresponding **quan** bit portion for the current layer. If this calculated number is consistent with the size of the **quan** portion it has received, the decoder can assume there is no propagational error in the received **map** bits. Otherwise, a propagational error has occurred and an error cancellation scheme can be invoked to terminate the propagation.

In the case where channel coding is employed, our coding structure provides a unique way to prioritize the coded data stream for different degrees of channel error protections, which is more efficient and has become available in many communication systems. In order to maintain near zero-tolerance for bit errors in the rate distribution information, channel coders with high protection rates are applied to the **map** bits. Since the size of the **map** bit portion is relatively small, high-channel error protection is usually not costly in the sense of data expansion. On the other hand, bit errors in the **quan** bits only cause localized reconstruction errors, as only fixed rate LVQs are used. Therefore, some low-level channel error protection should be sufficient for the **quan** bits. Such unequal error protection (UEP) approaches usually yield good tradeoffs between source coding and channel coding under the constraint of overall channel bandwidth.

In addition, the LOT, LBT and most filter banks of interest have a distinct feature that they can smooth out a localized distortion over a wide area so that the effect of the distortion becomes less perceivable. This is because of the overlapping nature of LOT, LBT bases and subband filters at the inverse trans-

(a) (b) (c)

Fig. 9. Samples of decoded "Akiyo" frames, i.e., the 100th frame zoomed at the center (100 × 100) block. (a) LBT at 0.25 bps. (b) DWT at 0.25 bpp. (c) MPEG-2 at 0.25 bpp.

(a) (b) (c)

Fig. 10. Samples of decoded "Hall Monitor" frames, i.e., the 100th frame zoomed at the center (100 × 100) block. (a) LBT at 0.25 bps. (b) DWT at 0.25 bpp. (c) MPEG-2 at 0.25 bpp.

form. This feature is particularly helpful at low BERs when only a few isolated distortions may occur and they do not interfere with each other. In fact, LOTs, LBTs, and filter banks can also be useful for error-concealment purposes.

V. Experimental Results

The video encoder consists of a 3-D transform followed by a quantization/coding procedure based on the adaptive quantization just described. The decoder contains the components needed to perform the inverse process. The 3-D transform is achieved through spatial and temporal extensions of five different 1-D transform methods including DCT, LOT, LBT, USB, and DWT. The extensions are symmetric in all three dimensions, i.e., the number of decomposition levels is the same. Eight-point DCT, LOT, and LBT are used to generate $8 \times 8 \times 8$ uniform subbands ($M = 8$), which correspond to a 3-level, 2-band uniform decomposition. The LOT and LBT matrices are based on equations (5) and (6). The 9/7-tap biorthogonal filter bank [15] is used in the USB and the DWT. However, when a data block is shorter than 8, the 2-tap Haar filter bank is used instead.

Every 16 picture frames are grouped together to form the input data cubes. Using the DCT, LOT, or LBT, an 8-band uniform decomposition is obtained in a single stage. In order to improve the energy compaction, an extra level of subband decomposition is performed in the lowest spatial-temporal frequency (or DC) subband. The resulting video coders are denoted as `DCT`, `LOT`, and `LBT`. We also tested two other decompositions. In the first, we replaced the uniform decomposition of the DCT/LOT/LBT by a 3-level, 8-band USB using the 9/7 filters. As in the LT case, the DC subband is further decomposed using one extra subband decomposition. We denote the coder with this transform as `USB`. We expect the `USB` and `LBT` to perform similarly because of their identical 3-D subband structure. We also implemented another coder using a 4-level pyramid decomposition, which is denoted `DWT`. It is clear that the lowest eight subbands in all these implementations have the same size.

The size of the quantization unit in the 3-D subband domain is set to $2 \times 2 \times 1$, in which the 1 is for the temporal dimension. The 3-D quadtree coding is used for the **MAP** operation, and the Z_4 LVQ is used in the **QUAN** operation. For easy implementation, the levels of the quadtree coding are consistent with levels of the subband decomposition. To facilitate comparison, three versions of each different implementation were created. The default version, which is denoted as the `ab` version, uses arithmetic coding in the **MAP** operation, and does not use arithmetic coding in the **QUAN** operation. In the other two versions, one uses arithmetic coding in both operations, which is denoted as the `aa` version, and the other one does not use any arithmetic coding, which is then denoted as the `bb` version. The arithmetic coder is based on [27].

We compared the subband video coders with the MPEG-2 coder in both noiseless and noisy environments. MPEG-2 was selected for the comparison because it has been widely used

TABLE I
PERFORMANCE COMPARISON OF ALL TESTED VIDEO CODERS IN NOISELESS ENVIRONMENT

PSNR (dB)	mean	max	min	mean	max	min
Akiyo	0.25 bpp			0.50 bpp		
LBT-ab	**46.56**	48.75	44.06	**50.79**	52.28	49.17
LOT-ab	46.16	48.24	44.10	50.48	52.02	48.85
DCT-ab	46.22	48.34	44.09	50.53	52.07	49.01
USB-ab	45.65	48.11	41.31	49.90	51.83	46.76
DWT-ab	44.82	46.07	42.14	48.33	49.14	46.97
MPEG-2	42.77	43.81	37.95	45.74	46.65	42.66
Hall	0.25 bpp			0.50 bpp		
LBT-ab	**40.14**	41.67	38.94	42.37	43.63	41.47
LOT-ab	40.03	41.50	39.30	**42.56**	43.69	42.15
DCT-ab	**40.14**	41.61	39.43	**42.56**	43.61	42.18
USB-ab	39.74	41.23	38.52	42.24	43.44	41.07
DWT-ab	37.74	39.59	36.32	41.16	42.41	39.71
MPEG-2	35.71	36.91	32.67	38.66	40.17	36.34

in many video communication systems, and represents a well established and respected benchmark. The Berkeley MPEG-2 implementation was used. In the MPEG-2 setting, each group of pictures (GOP) contains 15 frames, which is close to the 16-frame setting in our coders, and the I/P frame distance is set to 3. The test video sequences were the luminance components of the "Akiyo" and "Hall Monitor" sequences, both in CIF (352 $\times$ 288) resolution and at 30 fps. The test bit rates were 1520640 bps (or 0.5 bpp) and 760320 bps (or 0.25 bpp).

A. Compression Performance With No Channel Noise

Figs. 5–8 show the frame-by-frame peak-signal-to-noise ratio (PSNR) results of all the coders in the noiseless environment. The `DCT`, `LOT`, `LBT`, `USB`, and `DWT` coders are all using their default versions. From these figures we can see that all subband video coders outperform the MPEG-2 coder on the two test sequences, sometimes by as much as 5 dB. A summarized comparison is provided in Table I, in which the PSNR results are averaged over the 120 picture frames. Samples of decoded video frames are shown in Figs. 9 and 10 for perceptual evaluation. We find that the `LBT` coder usually has the best performance. It is closely followed by the LOT and DCT coders. Comparing the `USB` and `DWT` coders, we can see that the uniform decomposition outperforms the pyramid decomposition under our coding structure, at a cost of a significant increase in computation. Considering both efficiency and computation, we realize that there could be an advantage in selecting the LT-based transforms instead of the subband filtering based transforms in a subband image and video coding algorithm. We also notice that there is a significant PSNR decrease at the last frame of each 16-frame group in all of our video coders. This is caused by the simple progressive coding procedure used through the bit-planes of all frames. This last frame performance drop can be addressed by introducing a bit rate allocation procedure to ensure that the target bit rate is met at the end of a bit-plane for all frames. However, the resulting improvement in quality may not justify the increase in computation.

Tables II–V show the PSNR results of three different versions of all coders over the first 120 frames of the two test sequences. The "`ab`," "`aa`," and "`bb`" versions are defined in the previous section. It is noteworthy that the LVQ in the **QUAN** operation has fairly high coding efficiency, and even if the LVQ indices are coded by an additional arithmetic coder, no significant compression improvement is observed. On the other hand, if no arithmetic coding is used in both **MAP** and **QUAN** operations, the overall performance is still superior to that of an MPEG-2 coder. In fact, the performance in this case can be further improved by increasing the levels of subband decomposition and quadtree coding.

All these findings reveal that, for certain applications such as news broadcasting and surveillance, high performance video

TABLE II
PERFORMANCE COMPARISON OF VARIOUS IMPLEMENTATIONS OF THE USB AND DWT CODING ALGORITHMS IN NOISELESS ENVIRONMENT

PSNR (dB)	mean	max	min	mean	max	min
Akiyo	0.25 bpp			0.50 bpp		
USB-ab	45.65	48.11	41.31	49.90	51.83	46.76
DWT-ab	44.82	46.07	42.14	48.33	49.14	46.97
USB-aa	45.92	48.23	41.83	50.13	52.17	47.12
USB-bb	44.42	47.07	40.50	48.82	50.90	44.78
Hall	0.25 bpp			0.50 bpp		
USB-ab	39.74	41.23	38.52	42.24	43.44	41.07
DWT-ab	37.74	39.59	36.32	41.16	42.41	39.71
USB-aa	39.89	41.41	38.60	42.39	43.58	41.16
USB-bb	38.68	40.66	36.91	41.55	43.00	39.99

TABLE III
PERFORMANCE COMPARISON OF VARIOUS IMPLEMENTATIONS OF THE DCT CODING ALGORITHMS IN NOISELESS ENVIRONMENT

PSNR (dB)	mean	max	min	mean	max	min
Akiyo	0.25 bpp			0.50 bpp		
DCT-ab	46.22	48.34	44.09	50.53	52.07	49.01
DCT-aa	46.46	48.47	44.41	50.76	52.39	49.25
DCT-bb	44.89	47.05	42.61	49.38	50.97	47.62
Hall	0.25 bpp			0.50 bpp		
DCT-ab	40.14	41.61	39.43	42.56	43.61	42.18
DCT-aa	40.27	41.76	39.55	42.70	43.79	42.28
DCT-bb	39.02	40.71	37.62	41.90	42.80	41.05

TABLE IV
PERFORMANCE COMPARISON OF VARIOUS IMPLEMENTATIONS OF THE LOT CODING ALGORITHM IN NOISELESS ENVIRONMENT

PSNR (dB)	mean	max	min	mean	max	min
Akiyo	0.25 bpp			0.50 bpp		
LOT-ab	46.16	48.24	44.10	50.48	52.02	48.85
LOT-aa	46.36	48.30	44.34	50.66	52.30	49.05
LOT-bb	45.66	46.95	42.66	49.35	50.97	47.70
Hall	0.25 bpp			0.50 bpp		
LOT-ab	40.03	41.50	39.30	42.56	43.69	42.15
LOT-aa	40.17	41.69	39.44	42.69	43.91	42.29
LOT-bb	38.92	40.63	37.70	41.85	42.80	40.97

TABLE V
PERFORMANCE COMPARISON OF VARIOUS IMPLEMENTATIONS OF THE LBT CODING ALGORITHM IN NOISELESS ENVIRONMENT

PSNR (dB)	mean	max	min	mean	max	min
Akiyo	0.25 bpp			0.50 bpp		
LBT-ab	46.56	48.75	44.06	50.79	52.28	49.17
LBT-aa	46.81	48.91	44.29	51.07	52.49	49.43
LBT-bb	45.48	47.31	43.02	49.61	51.34	47.78
Hall	0.25 bpp			0.50 bpp		
LBT-ab	40.14	41.67	38.94	42.37	43.63	41.47
LBT-aa	40.36	41.77	39.19	42.59	43.82	41.65
LBT-bb	39.15	40.94	38.23	41.75	42.90	40.99

TABLE VI
SOURCE AND CHANNEL CODING BIT RATE ALLOCATIONS FOR "AKIYO" SEQUENCE AT 0.25 BPP

E_b/N_0 = 4.33	MPEG-2 (EEP)	DWT (UEP)			
Conv. Rate	3/4	3/4	5/6	7/8	no
R-S	yes	yes	yes	yes	no
Source Coding	68.6%	16.4%	17.5%	23.8%	23.4%
Channel Coding	31.4%	7.5%	5.4%	6.0%	
E_b/N_0 = 6.77	MPEG-2 (EEP)	DWT (UEP)			
Conv. Rate	7/8	7/8	no	no	
R-S	yes	yes	yes	no	
Source Coding	80.0%	17.5%	47.7%	26.3%	
Channel Coding	20.0%	4.4%	4.1%		

TABLE VII
SOURCE AND CHANNEL CODING BIT RATE ALLOCATIONS FOR "AKIYO" SEQUENCE AT 0.50 BPP

E_b/N_0 = 4.33	MPEG-2 (EEP)	DWT (UEP)			
Conv. Rate	3/4	3/4	5/6	7/8	no
R-S	yes	yes	yes	yes	no
Source Coding	68.6%	13.7%	9.8%	34.5%	24.1%
Channel Coding	31.4%	6.3%	3.0%	8.6%	
E_b/N_0 = 6.77	MPEG-2 (EEP)	DWT (UEP)			
Conv. Rate	7/8	7/8	no	no	
R-S	yes	yes	yes	no	
Source Coding	80.0%	16.3%	44.6%	31.2%	
Channel Coding	20.0%	4.1%	3.8%		

coding can be achieved by a simple 3-D transform coding approach without the complicated motion compensated prediction procedure. Some advantages of the 3-D transform coding scheme include:

- the transform methods usually have fast algorithms for practical hardware and software implementations;
- the encoder and decoder have similar computation complexities;
- the structure allows easy switching among various computational settings.

It is also interesting to note that high coding efficiency does not have to rely on entropy coding. A video coder entirely without entropy coding, such as the arithmetic coding, may have good computational efficiency, and can also avoid some of the associated intellectual property problems.

B. Compression Performance With Channel Noise

Because the error-resilient feature comes largely from the coded data stream partitioning, different transform methods will not produce significantly different results in noisy environment. Therefore in noisy channel simulations, we only tested the `DWT` default coder against the MPEG-2 coder. We applied concatenated forward error-correction codes (FEC), which are specified in the DVB-S standard (*ETS 300 421*), to both our data stream and the MPEG-2 data stream. This FEC coding system contains an inner convolutional code and an outer Reed–Solomon (R–S) code. The convolutional code is obtained from a NASA standard convolutional code ($r = 1/2$, $k = 7$) with puncturing rates of $\{3/4, 5/6, 7/8\}$. The (204, 188, 8) R–S code is a shorten version of the (255, 239, 8) R–S code. A white Gaussian noise channel with BPSK modulation was used as the channel model, and soft-decision Viterbi coding was used at the convolutional decoder. Experiments were performed using channel signal-to-noise ratios E_b/N_0 of 4.33 dB and 6.77 dB, which correspond to BERs of 10^{-2} and 10^{-3} in the uncoded channel. The MPEG-2 data streams were protected at equal error-protection rates in each case, while the UEP scheme was used for our data streams. With the ability to classify our data stream into **map** and **quan** sub-streams, different channel protection rates can be assigned to achieve maximum effectiveness. Tables VI–IX show the bit allocation between the source and channel coding for each simulation. Figs. 11–14 show the noisy channel simulation results in average PSNR. Simulations were repeated 50 times for each test condition, and PSNR values for each frame were calculated as the average of all 50 reconstructions. Table X shows the PSNR results averaged over the first 60 frames of

TABLE VIII
SOURCE AND CHANNEL CODING BIT RATE ALLOCATIONS FOR "HALL MONITOR" SEQUENCE AT 0.25 BPP

E_b/N_0 = 4.33	MPEG-2 (EEP)	DWT (UEP)			
Conv. Rate	3/4	3/4	5/6	7/8	no
R-S	yes	yes	yes	yes	no
Source Coding	68.6%	15.7%	17.9%	23.9%	23.8%
Channel Coding	31.4%	7.2%	5.5%	6.0%	
E_b/N_0 = 6.77	MPEG-2 (EEP)	DWT (UEP)			
Conv. Rate	7/8	7/8	no	no	
R-S	yes	yes	yes	no	
Source Coding	80.0%	17.2%	47.4%	27.1%	
Channel Coding	20.0%	4.3%	4.0%		

TABLE IX
SOURCE AND CHANNEL CODING BIT RATE ALLOCATIONS FOR "HALL MONITOR" SEQUENCE AT 0.50 BPP

E_b/N_0 = 4.33	MPEG-2 (EEP)	DWT (UEP)			
Conv. Rate	3/4	3/4	5/6	7/8	no
R-S	yes	yes	yes	yes	no
Source Coding	68.6%	15.4%	9.8%	32.7%	23.9%
Channel Coding	31.4%	7.1%	3.0%	8.1%	
E_b/N_0 = 6.77	MPEG-2 (EEP)	DWT (UEP)			
Conv. Rate	7/8	7/8	no	no	
R-S	yes	yes	yes	no	
Source Coding	80.0%	17.3%	42.3%	32.5%	
Channel Coding	20.0%	4.3%	3.6%		

TABLE X
NOISY CHANNEL SIMULATION PSNR RESULTS (IN dB) OF THE 3-D SUBBAND VIDEO CODER DWT AND THE MPEG-2 CODER AVERAGED OVER 60 FRAMES

BER	0	10^{-3}	10^{-2}	0	10^{-3}	10^{-2}
Akiyo	0.25 bpp			0.50 bpp		
DWT	44.86	44.21	43.09	48.35	48.00	47.12
MPEG-2	42.73	41.56	40.62	45.76	44.83	44.15
Hall	0.25 bpp			0.50 bpp		
DWT	37.79	37.13	36.16	41.19	40.70	39.79
MPEG-2	35.67	34.49	33.53	38.67	37.76	37.15

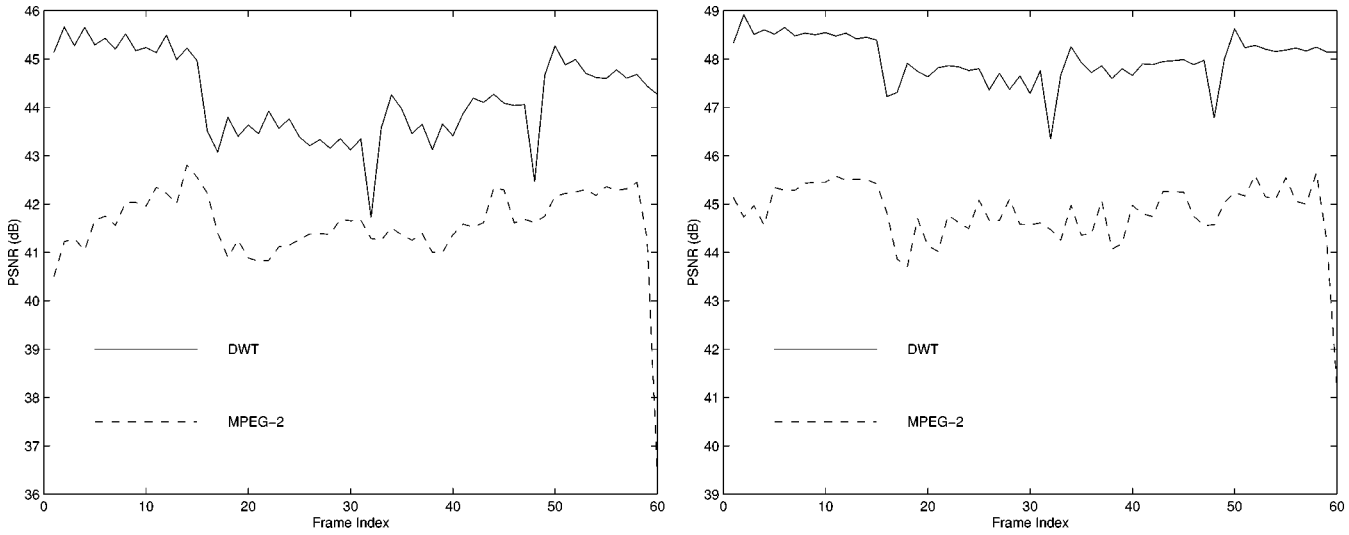

Fig. 11. PSNR performance of the 3-D subband video coder DWT and the MPEG-2 coder for "Akiyo" sequence in noisy channels with channel SNR $E_b/N_0 = 6.77$ or BER $= 10^{-3}$. (a) 0.25 bpp or 760320 bps. (b) 0.50 bpp or 1520640 bps.

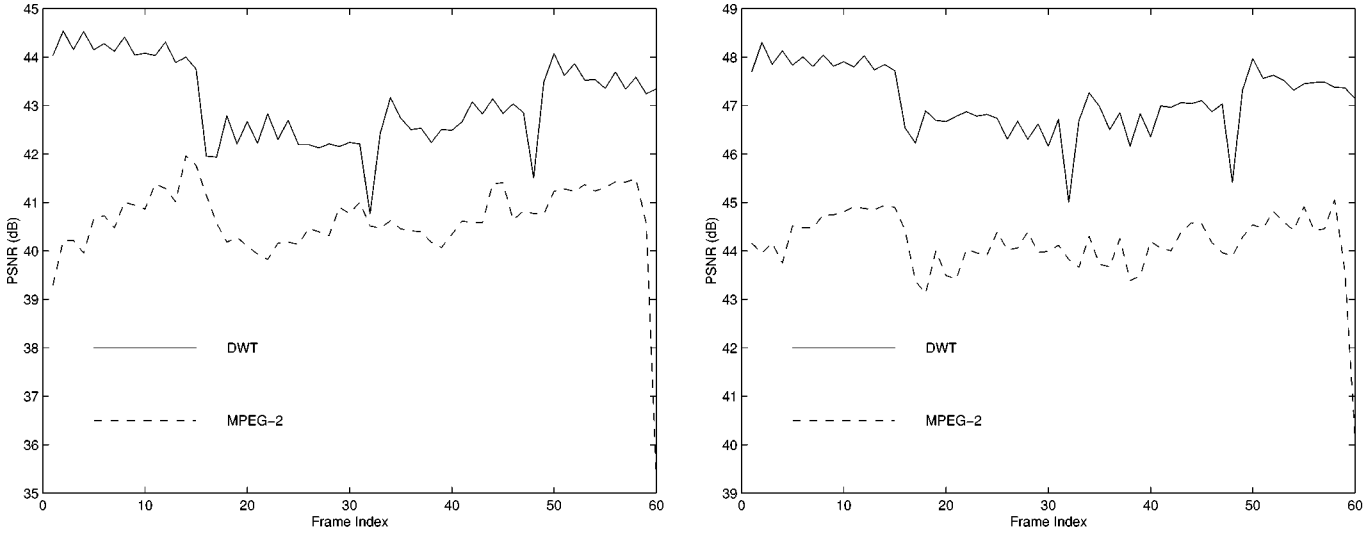

Fig. 12. PSNR performance of the 3-D subband video coder DWT and the MPEG-2 coder for "Akiyo" sequence in noisy channels with channel SNR $E_b/N_0 = 4.33$ or BER $= 10^{-2}$. (a) 0.25 bpp or 760320 bps. (b) 0.50 bpp or 1520640 bps.

the testing sequence in each simulation. These results clearly demonstrate the high compression performance of the 3-D subband video coder in noisy channels. Moreover, with the UEP scheme, the performance gain of the subband coder is further increased in noisy channels.

VI. Conclusion

In this paper, we have presented a new 3-D subband video coding structure. It consists of a simple 3-D data transform and an adaptive quantization procedure. We have implemented several video coders based on the DCT, LOT, and LBT uniform subband and DWT methods. All of these video coders are able to provide superior performance in both noiseless and noisy environments. A major novelty of the coding scheme is classifying the compressed data stream into sub-streams with different noise sensitivity levels for better channel adaptation. Quadtree coding and lattice vector quantization techniques are used to reduce the size of the VLC data portion and achieve high compression performance even without entropy coding. Simulation results have shown the viability of this approach in application areas such as wireless video communications. In particular, we believe that a video coder with a simple 3-D LBT/LOT or even DCT transform and a few levels of quadtree coding followed by a look-up table based LVQ can be a very attractive choice for portable computing devices.

We recognize that there are some common problems with the 3-D subband approach that we did not address in this paper, which include difficulties with fast motion scenes and scene cuts. The major intention of this paper is to investigate the effectiveness of 3-D data transform in certain applications. We understand that, to some extend, temporal filtering has clear advantages over MCP approach in the sense of compression and computation. However, the optimal range of motion activity for temporal filtering is still under study. At the same time, we are investigating adaptive 3-D data structures which can handle scene cuts.

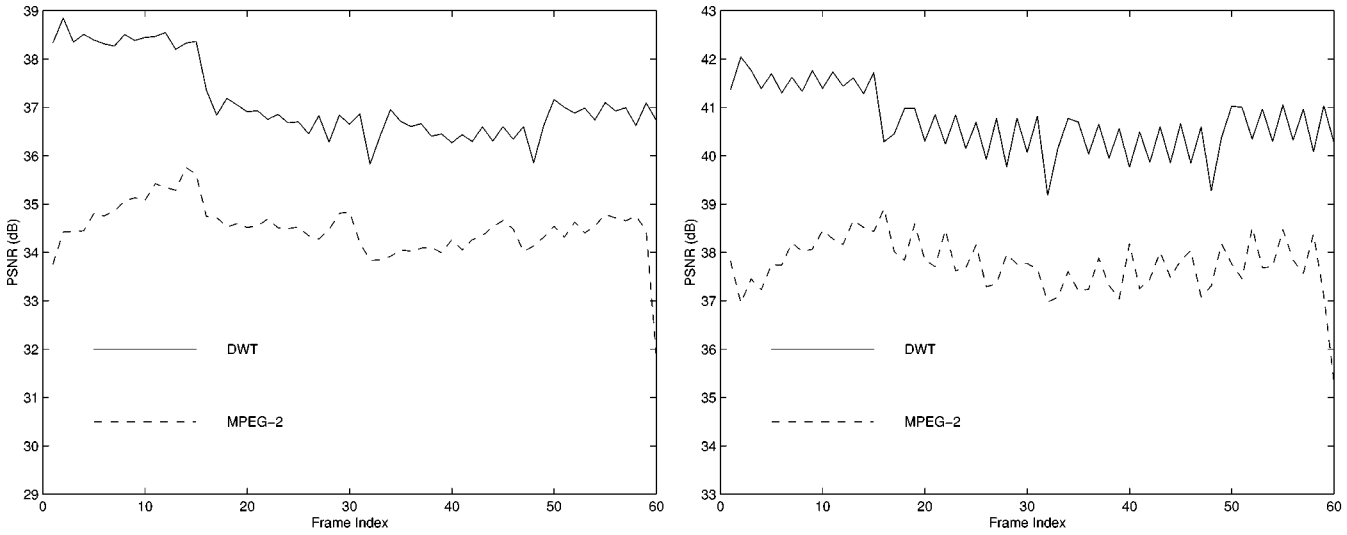

Fig. 13. PSNR performance of the 3-D subband video coder DWT and the MPEG-2 coder for "Hall Monitor" sequence in noisy channels with channel SNR $E_b/N_0 = 6.77$ or BER $= 10^{-3}$. (a) 0.25 bpp or 760320 bps. (b) 0.50 bpp or 1520640 bps.

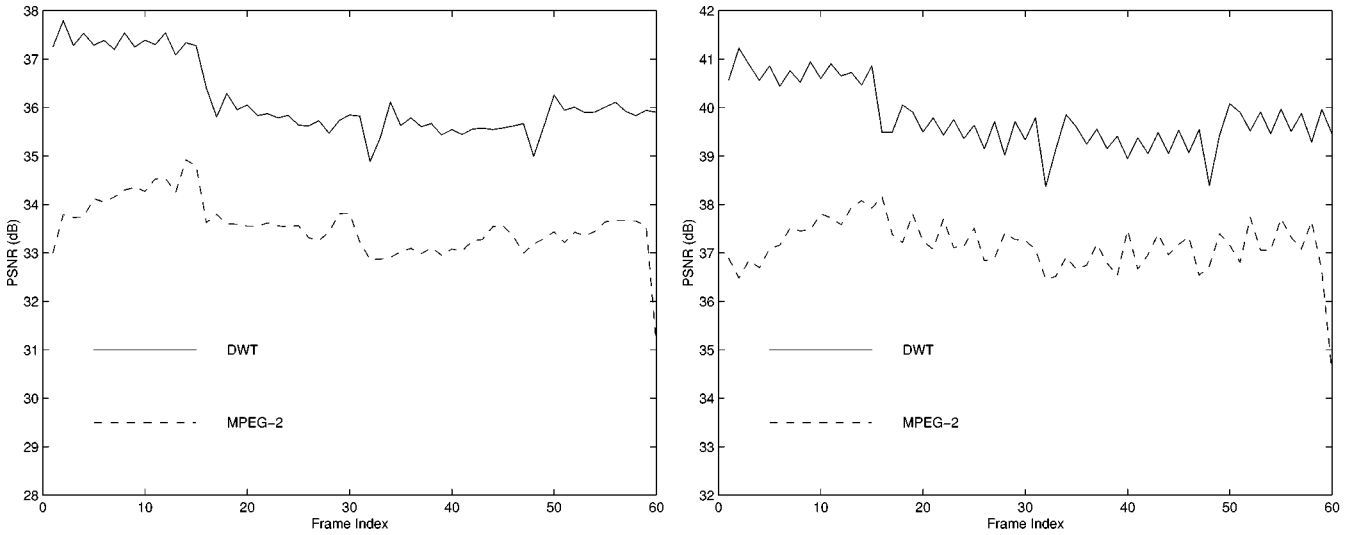

Fig. 14. PSNR performance of the 3-D subband video coder DWT and the MPEG-2 coder for "Akiyo" sequence in noisy channels with channel SNR $E_b/N_0 = 4.33$ or BER $= 10^{-2}$. (a) 0.25 bpp or 760320 bps. (b) 0.50 bpp or 1520640 bps.

The LT bases we exploited in this paper were designed for maximizing the coding gain. Recently, researchers have begun to explore several different LT bases [28], [29], which intentionally introduce some limited amount of transform redundancy aimed at minimizing the channel error effects at the decoder. These LT bases are designed to achieve an effective balance between efficiency and robustness. We are currently studying such an approach in which error-resilient data transform methods, quantization/coding methods, and channel error-protection methods can be integrated to improve the quality of service for wireless video communications.

REFERENCES

[1] V. Bhaskaran and K. Konstantinides, *Image and Video Compression Standards, Algorithms and Architectures*. Norwell, MA: Kluwer, 1997.

[2] A. Bovik, Ed., *Handbook of Image and Video Processing*. New York: Academic , 2000.

[3] K. R. Rao and P. Yip, *Discrete Cosine Transform: Algorithms, Advantages, Applications*. San Diego, CA: Academic, 1990.

[4] D. Taubman and A. Zakhor, "Multirate 3-d subband coding of video," *IEEE Trans. Image Processing*, vol. 3, pp. 572–588, Sept. 1994.

[5] B.-J. Kim and W. A. Pearlman, "An embedded wavelet video coder using three-dimensional set partitioning in hierarchical trees," in *IEEE Data Compression Conf.*, Snowbird, UT, Mar. 1997, pp. 251–260.

[6] S. A. Martucci, I. Sodager, T. Chiang, and Y.-Q. Zhang, "A zerotree wavelet video coder," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 7, pp. 109–118, Feb. 1997.

[7] H. Man and M. Smith, "A new three-dimensional subband coding technique for digital video broadcasting," in *Proc. 2nd Annu. UCSD/IEEE Conf. Wireless Communications*, San Diego, CA, Feb.–Mar. 1999.

[8] S. J. Choi and J. W. Woods, "Motion compensated 3-d subband coding of video," *IEEE Trans. Image Processing*, vol. 8, pp. 155–167, Feb. 1999.

[9] H. S. Malvar and D. H. Staelin, "The LOT: Transform coding without blocking effects," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 37, pp. 553–559, Apr. 1989.

[10] H. S. Malvar, *Signal Processing with Lapped Transforms*. Norwood, MA: Artech House, 1992.

[11] ——, "Biorthogonal and nonuniform lapped transforms for transform coding with reduced blocking and ringing artifacts," *IEEE Trans. Signal Processing*, vol. 46, pp. 1043–1053, Apr. 1998.

[12] T. D. Tran, R. de Queiroz, and T. Q. Nguyen, "The generalized lapped biorthogonal transform," in *Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Processing*, Seattle, WA, May 1998, pp. 1441–1444.

[13] R. L. de Queiroz and T. D. Tran, "Lapped transforms for image compression," in *The Handbook on Transforms and Data Compression*, K. R. Rao and P. Yip, Eds. Boca Raton, FL: CRC Press, 2001, ch. 5.
[14] T. D. Tran, R. de Queiroz, and T. Q. Nguyen, "Linear phase perfect reconstruction filter bank: Lattice structure, design, and application in image coding," *IEEE Trans. Signal Processing*, vol. 148, pp. 1333–147, Jan. 2000.
[15] M. Antonini, M. Barlaud, P. Mathieu, and I. Daubechies, "Image coding using wavelet transform," *IEEE Trans. Image Processing*, vol. 1, pp. 205–220, Feb. 1992.
[16] S. K. Mitra, D. C. Huey, and R. J. Sherwood, "New methods of digital ladder realization," *IEEE Trans. Audio Electroacoust.*, vol. AU-21, pp. 485–500, Dec. 1973.
[17] S. K. Mitra and R. J. Sherwood, "Digital ladder networks," *IEEE Trans. Audio Electroacoust.*, vol. AU-21, pp. 30–36, Feb. 1973.
[18] W. Sweldens, "The lifting scheme: A construction of second generation wavelets," *SIAM J. Math. Anal.*, vol. 29, no. 2, pp. 511–546, 1997.
[19] P. Vaidyanathan, *Multirate Systems and Filter Banks*. Englewood Cliffs, NJ: Prentice-Hall, 1993.
[20] H. Man, F. Kossentini, and M. Smith, "A family of efficient and channel error resilient wavelet/subband image coders," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 9, pp. 95–108, Feb. 1999.
[21] H. Man, M. Smith, and F. Kossentini, "On robustness of adaptive quantization for subband coding of images," in *SPIE Proc. Visual Communications and Image Processing*, San Jose, CA, Jan. 1999.
[22] J. M. Shapiro, "Embedded image coding using zerotrees of wavelet coefficients," *IEEE Trans. Signal Processing*, vol. 41, pp. 3445–3462, Dec. 1993.
[23] A. Said and W. A. Pearlman, "A new fast and efficient image codec based on set partitioning in hierarchical trees," *IEEE Trans. Circuits Systems Video Technol.*, vol. 6, pp. 243–250, June 1996.
[24] Z. Xiong, K. Ramchandran, and M. T. Orchard, "Space-frequency quantization for wavelet image coding," *IEEE Trans. Image Processing*, vol. 6, pp. 677–693, May 1997.
[25] R. L. Joshi, H. Jafarkhani, J. H. Kasner, T. R. Fischer, N. Farvardin, M. W. Marcellin, and R. H. Bamberger, "Comparison of different methods of classification in subband coding of images," *IEEE Trans. Image Processing*, vol. 6, pp. 1473–1486, Nov. 1997.
[26] H. Man, F. Kossentini, and M. Smith, "A class of EZW image coders for noisy channels," in *Proc. IEEE Int. Conf. Image Processing*, vol. 3, Santa Barbara, CA, Oct. 1997, pp. 90–93.
[27] I. H. Witten, R. M. Neal, and J. G. Cleary, "Arithmetic coding for data compression," *Commun. ACM*, vol. 30, pp. 520–540, June 1987.
[28] S. Hemami, "Reconstruction-optimized lapped orthogonal transforms for robust image transmission," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 6, pp. 168–181, Apr. 1996.
[29] D.-M. Chung and Y. Wang, "Multiple description image coding using signal decomposition and reconstruction based on lapped orthogonal transforms," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 9, pp. 895–908, Sept. 1999.

Hong Man (M'00) received the B.S. degree from Soochow University, China, in 1988, the M.S. degree from Gonzaga University, Spokane, WA, in 1994, and the Ph.D. degree from Georgia Institute of Technology, Atlanta, in 1999, all in electrical engineering.

He is currently an Assistant Professor in the Department of Electrical and Computer Engineering, Stevens Institute of Technology, Hoboken, NJ. He is also the Director of the Visual Information Environment Laboratory at Stevens Institute of Technology. His research interests are in the areas of image and video processing, multimedia communications and networking, wireless *ad hoc* networks, and image analysis.

Dr. Man has served as a member of organizing committee for the IEEE International Workshop on Multimedia and Signal Processing (MMSP'02). He is a member of the IEICE.

Ricardo L. de Queiroz (SM'99) received the B.S. degree from Universidade de Brasilia, Brazil, in 1987, the M.S. degree from Universidade Estadual de Campinas, Brazil, in 1990, and the Ph.D. degree from University of Texas at Arlington in 1994, all in electrical engineering.

During 1990–1991, he was a Research Associate with the DSP Research Group, Universidade de Brasilia. In 1994, he was a Teaching Assistant at the University of Texas at Arlington. He joined Xerox Corporation, Webster, NY, in August 1994, where he is currently a Member of the Research Staff at the Color and Digital Imaging Systems Laboratory. He is also an Adjunct Faculty Member at the Rochester Institute of Technology, Rochester, NY. He has contributed chapters to books and published extensively in journals and conferences, and holds more than 20 issued patents, with several still pending. His research interests include multirate signal processing, image and signal compression, color imaging, processing compressed images, multimedia and image understanding. He has received several grants and scholarships, including some from the Brazilian Government.

Dr. de Queiroz is an Associate Editor for the IEEE SIGNAL PROCESSING LETTERS. He has been actively involved with the Rochester chapter of the IEEE Signal Processing Society, where he served as Chair and has organized the annual Western New York Image Processing Workshop since its inception. For these activities, he received an Outstanding Activity Award from the local section. He also received an Academic Excellence Award from the University of Texas at Arlington. He is a member of IS&T.

Mark J. T. Smith (F'95) received the B.S. degree from the Massachusetts Institute of Technology, Cambridge, and the M.S. and Ph.D. degrees from the Georgia Institute of Technology, Atlanta, all in electrical engineering.

He is an electrical engineering Professor in the College of Engineering, Georgia Institute of Technology. Until September 2001, he served as the Executive Assistant to the President of the School, and also as an IEEE Distinguished Lecturer in Signal Processing during 1997–1998. He has authored many papers in the areas of speech and image processing, filter banks, and wavelets, and is the co-author of *Introduction to Digital Signal Processing* (New York: Wiley, 1991) and *Digital Filtering* (New York: Wiley, 1993). He is also co-editor of *Wavelets and Subband Transforms: Design and Applications* (Norwell, MA: Kluwer, 1996) and co-author of *A Study Guide for Digital Image Processing* (San Francisco, CA: Scientific, 1998).

Dr. Smith is a past Chairman of the IEEE Signal Processing Digital Signal Processing Technical Committee in the IEEE Signal Processing Society and a former member of the Board of Governors. He has served as an Associate Editor for the IEEE TRANSACTIONS ON ACOUSTICS, SPEECH, AND SIGNAL PROCESSING and as a member of the MIP's Advisory Board of the National Science Foundation. He has been active as a member of the Organizing Committees for the IEEE Digital Signal Processing Workshops since 1988, serving as General Chairman in 1992 and 1994, and as General Co-Chair in 2002.