Wavelet-based Video Coding for Real-Time Transmission

Chang-Mo Yang*, Myung-Ok Lee** and Yo-Sung Ho*
*Kwangju Institute of Science and Technology (K-JIST)
Tel : +82-62-970-2263 Fax : +82-62-970-2204 E-mail: {changmo, hoyo}@kjist.ac.kr
**Research Center for Multimedia & Communication, HiChips Inc.
Tel : +82-62-511-3450 Fax : +82-62-511-3453 E-mail: myklee@hichips.com

Abstract: In this paper, we propose new wavelet-based video coding schemes for real-time video transmission. The proposed video coding schemes include multi-level dyadic wavelet decomposition, motion estimation and motion compensation, raster scanning within each subband, formation of block trees, partitioning of block trees, and adaptive arithmetic entropy coding. Although the proposed video coding schemes are simple, they produce bitstreams with good features, including SNR scalability from the embedded nature. Experimental results demonstrate that the proposed video coding schemes are quite competitive to other wavelet-based video coders in the literature.

Keywords: Video Compression, Wavelet, Embedded Coding, Scalability, Block Partitioning

1. INTRODUCTION

In recent years, the discrete wavelet transform (DWT) has been popularly employed in image and video coding applications. Using a discrete wavelet transform, we divide an input signal into a number of segments, each corresponding to a different frequency subband. Therefore, processing of each subband can be easier than processing the whole image frame. In addition, wavelet-based coders have no blocking artifacts and mosquito noises, which are main problems of DCT-based coding algorithms at low bit rates.

Depending on the way how temporal redundancy is exploited, we can classify wavelet-based video coding schemes into two categories: three-dimensional (3-D) subband coding and motion-compensated predictive coding schemes.

In 3-D subband coding [1,2], 2-D image coding techniques are extended to 3-D video coding. The embedded zerotree wavelet (EZW) [3] coding scheme can be utilized to encode error frames obtained by motion compensation. Tham and Ranganath proposed a 3-D subband video coding scheme for very low bit-rate applications [1]. The video coder first performs a motion compensated 3-D wavelet decomposition of a group of video frames, and then encodes important wavelet coefficients using a data structure called tri-zerotrees (TRI-ZTR). In a 3-D SPIHT [2] algorithm, they extend the set partitioning in hierarchical trees (SPIHT) algorithm [4] that has proved successful for still image coding into 3-D video coding. In 3-D SPIHT, 3-D spatial-temporal orientation trees are coupled with SPIHT sorting and refinement. They also extend the scheme to color-embedded video coding without explicit bit allocation, which can be used for some color plane representation.

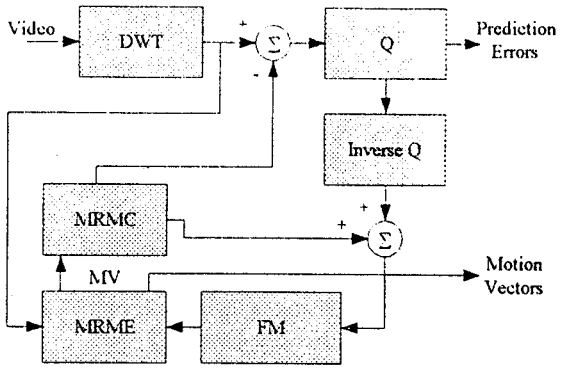
However, 3-D subband coding schemes have some drawbacks. They generally blur moving objects in temporally low resolution pictures. It is caused by the averaging effect of low-pass filtering performed in temporal decomposition. Motion-compensated 3-D subband video coding techniques have been proposed to alleviate this problem. Besides, they improve compression efficiency by concentrating signal energy in the temporal low subband. However, they require large frame memory for 3-D subband decomposition because these techniques operate on several consecutive frames. As the target number of temporal layers increases, the number of required frames increases exponentially and unavoidable excessive encoding and decoding delay is introduced.

The motion-compensated predictive coding involves DCT-based video coding schemes. Recently, Lee and Oh have proposed a new wavelet-based motion-compensated video coder that affords temporal and spatial resolution scalabilities [5]. A new motion prediction structure with a temporal hierarchy of frames is adopted to afford temporal resolution scalability. It can provide a higher compression ratio than replenishment schemes, since motion estimation (ME) further reduces the temporal redundancy. For spatial scalability, wavelet decomposition with multiresolution motion estimation (MRME) is employed. The prediction error is encoded by SPIHT.

In this paper, we propose wavelet-based video coding schemes for real-time transmission. After we apply three-step block matching ME and MRME operations, we encode prediction errors by a new quantization method exploiting the relationship among wavelet coefficients and block based partitioning. We can also consider a bounding box of moving objects and encode only the region of moving objects in the image. This strategy is very useful for surveillance systems. For entropy coding, we use an adaptive arithmetic coder as in JPEG. For rate control, we exploit the embedded property of residual quantization.

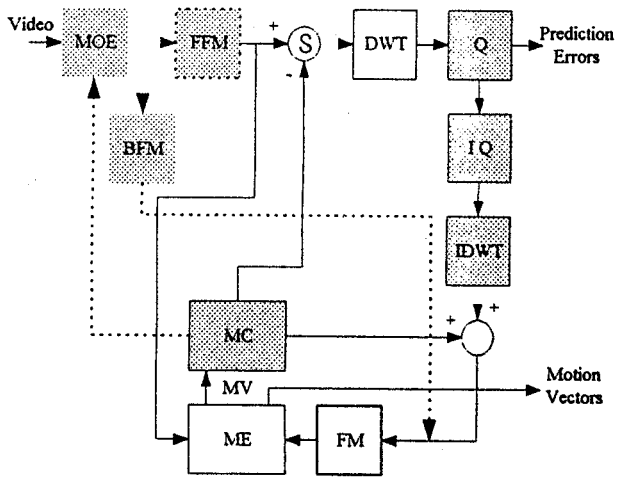
2. SYSTEM OVERVIEW

In video coding, motion estimation for removing temporal redundancy of video sequences is very crucial and generally determines performance and complexity of the coder. In this paper, we propose two video coding schemes using different motion estimation algorithms.



DWT: Discrete Wavelet Transform
Q: Quantization
FM: Frame Memory
MRME: Multiresolution Motion Estimation
MRMC: Multiresolution Motion Compensation
MV: Motion Vector

Fig. 1. Multiresolution Motion-Compensated Video Coder



MOE: Moving Object Extraction
FFM: Foreground Frame Memory
BFM: Background Frame Memory
DWT: Discrete Wavelet Transform
IDWT: Inverse Discrete Wavelet Transform
Q: Quantization
IQ: Inverse Quantization
FM: Frame Memory
ME: Motion Estimation
MC: Motion Compensation
MV: Motion Vector

Fig. 2. Wavelet-based Video Coder using Bounding Box and Spatial-Domain Motion Compensation

Fig. 1 shows an MRME-based video coding scheme. Once, ME is performed in the discrete wavelet transform domain, the inverse discrete wavelet transform for ME of the next frame is not required. Therefore, its computational complexity is relatively lower than conventional DCT-based video coding schemes. After the input frame is decomposed into subbands by the discrete wavelet transform, ME and MC is performed. Then, prediction errors are quantized and entropy coded. Finally, the motion compensated frame is added to the reconstructed error frame to make the reference frame for the next frame.

Fig. 2 shows a spatial-domain ME-based video coding scheme. In this case, ME is performed in the spatial domain using the three-step block matching algorithm for real-time video applications. Prediction errors are quantized and entropy coded. In this scheme, we need to perform the inverse discrete wavelet transform and the inverse quantization to make the reference frame for the next frame. In this scheme, we can optionally use the bounding box extraction.

Once we separate the region of moving objects from the background, we encode only the region of moving objects. The background is not coded, but just copied from the previous frame. This strategy is quite useful for various video applications, such as surveillance systems.

3. THE PROPOSED VIDEO CODER

3.1. Discrete Wavelet Transform

The wavelet transform of a signal captures the localized time-frequency information of the signal. The property of time-frequency localization greatly enhances the ability to study behaviors of the signal as well as to change these features locally without significantly affecting the state of signal characteristics in other regions of frequency or time. In addition, the continuous wavelet transform is closely linked to the discrete wavelet transform. This relationship allows us to speak of a wavelet series of any finite energy signal, obtained by discretizing the continuous wavelet transform. The coefficients of such a wavelet series of a signal completely capture the time-frequency characteristics, with each coefficient corresponding to a discrete time-and-frequency window. This feature provides the discrete wavelet transform with a multi-resolution property, which makes possible the study of a signal at varying resolutions [6].

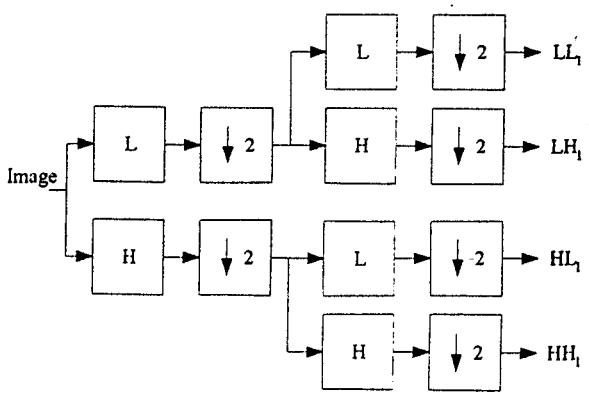


Fig. 3. 2-D Subband Decomposition

The discrete wavelet transform used in this paper is identical to one in a hierarchical subband system, where

subbands are logarithmically spaced in frequency and represent the octave-band decomposition. At first, an input image is divided into four subbands and critically subsampled. Each coefficient represents a spatial area corresponding to approximately a 2×2 area of the original image. Four subbands are generated by separable applications of vertical and horizontal filters. Subbands labeled LL_1, LH_1, HL_1 and HH_1 represent the finest scale wavelet coefficients, as shown in Fig. 3, the filters L and H are one-dimensional low and high pass filters, respectively.

In order to obtain the next coarser scale of wavelet coefficients, we decompose the subband LL_1 further and critically subsample its subbands again. This process continues until the final scale is reached. Fig. 4 represents the result of two-level wavelet decomposition of an image.

Fig. 4. Two-Level Wavelet Decomposition

3.2. Bounding Box Extraction

Although the MPEG-4 video coding algorithm is efficient, its computational complexity is high for real-time applications. In this paper, we propose a different coding scheme, where we use frame difference, median filtering, and thresholding operations. When an input image is entered to the coder, frame difference between the previous reconstructed frame and the current frame is computed. After we remove noises existing in the frame difference by median filtering, we can obtain a bounding box by a thresholding operation.

3.3. Motion Estimation and Motion Compensation

For the spatial-domain ME, we use one of the fast motion search algorithms, a three-step method. Although the full search method is more effective than the three-step search method, its computational complexity is not suitable for real-time video applications.

The three-step motion search method is a fine-coarse search mechanism. The first step involves motion estimation based on 4-pixel/4-line resolution at nine locations, with the center point corresponding to zero motion vector. The second step involves motion estimation based on 2-pixel/2-line resolution around the location selected by the first step. This is repeated in the third step with 1-pixel/1-line resolution [7]. The last step yields the motion vector. For a distortion measure, we use the mean absolute difference (MAD) for simplicity.

For MRME, we use a variable block-size ME method. The block size for ME is 2×2 in the LL. In the LL subband, motion vectors are estimated using the three-step motion search method. The search range of ±2 is utilized both in the horizontal and vertical directions in the LL subband. The mean absolute difference (MAD) criterion is used as a distortion measure.

For searching motion vectors in higher subbands, the parent-children relationship [3,4] is utilized, as shown in Fig. 5. In the proposed ME scheme, we do not perform a refinement procedure [5,8] of motion vector in higher subbands for simplicity. Instead, motion vectors of the LL subband are directly utilized for higher subbands. As increasing the scale of subbands, the block size for ME is doubled and motion vectors of the LL subband are also scaled by two. Fig. 6 illustrates the procedure.

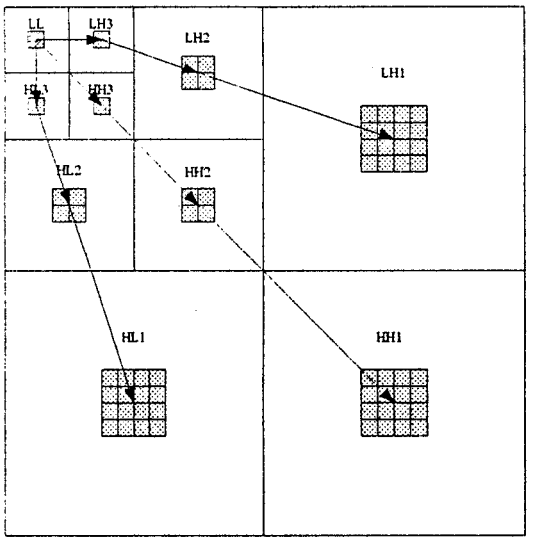


Fig. 5. Parent-Children Relationship

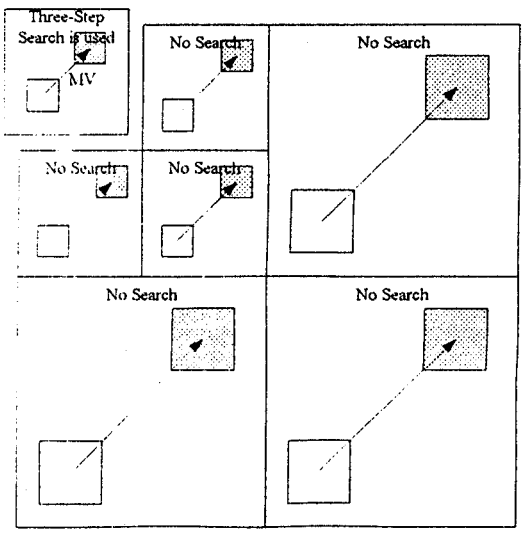


Fig. 6. Multiresolution ME/MC

3.4. Residual Quantization

When we decompose the error frame into wavelet coefficients, most coefficients in high frequency bands have very small magnitudes and can be quantized to zero without any noticeable distortion. Since, the portion of quantized zero coefficients is high, we only need to send positions of nonzero coefficients in high frequency subbands. The most simple and straightforward way is to send a map of nonzero coefficients for each subband to the decoder. However, this method cannot exploit inter-band correlation and some redundancy still exists in the data.

Generally, a zerotree structure of wavelet coefficients is employed to improve the compression efficiency for the significance map [3,4]. However, the memory usage of zerotree coding schemes is relatively high and they are not suitable for coding of large images. In order to reduce the memory usage of the zerotree coding scheme, we can exploit a new inter-band magnitude relationship existing in the wavelet coefficients.

After the input image is decomposed into subbands, one wavelet coefficient at a given scale is related to other coefficients corresponding to the same spatial location at the same scale of different orientation. We call the other coefficients as cousins of the given coefficient. Therefore, except for the lowest frequency subband, every coefficient has two cousins. For the lowest frequency subband, no coefficient has cousins. We define this relationship as the coefficient-cousin relationship. Fig. 7 illustrates this inter-band magnitude relationship.

In order to construct block trees using the coefficient-cousin relationship, we divide each subband into small blocks, nominally with the dimension of 64×64 pixels. After each subband is divided into small blocks, we can construct block trees using the relationship among cousins, as shown in Fig. 8.

Once we construct block trees, we identify the significance of a tree T using $c_{i,j}$, the coefficient in the position (i,j), and quantization step n by

$$S_n(T) = \begin{cases} 1, & if\ 2^n \le \max_{(i,j)\in T} |c_{i,j}| < 2^{n+1} \\ 0, & else \end{cases} \quad (1)$$

where "1" means significant and "0" means insignificant with respect to n. Similarly, we identify the significance of a coefficient $c_{i,j}$ by

$$S_n(c_{i,j}) = \begin{cases} 1, & if\ 2^n \le |c_{i,j}| < 2^{n+1} \\ 0, & else \end{cases} \quad (2)$$

When a tree T is identified to be significant with respect to n, it is partitioned into four small trees of the same size by the partitioning operation, as shown in Fig. 9. The significant tree is recursively split until the size of blocks $B_i(T)$, $i \in \{1,2,3\}$, is 4×4. When the size of blocks $B_i(T)$, $i \in \{1,2,3\}$, is 4×4, all coefficients in T are encoded. The main motivation for the proposed partitioning is to find high-energy areas quickly and encode them first.

In the proposed quantization algorithm, the significance information is stored in three ordered lists: list of insignificant pixels (LIP), list of insignificant blocks (LIB), and list of significant pixels (LSP). For all the lists, each entry is identified by a coordinate (i,j) that represents an individual pixel in LIP and LSP, and represents the starting position of the tree in LIB. The proposed algorithm consists of four coding passes: initialization, sorting pass, refinement pass, and update of the quantization step.

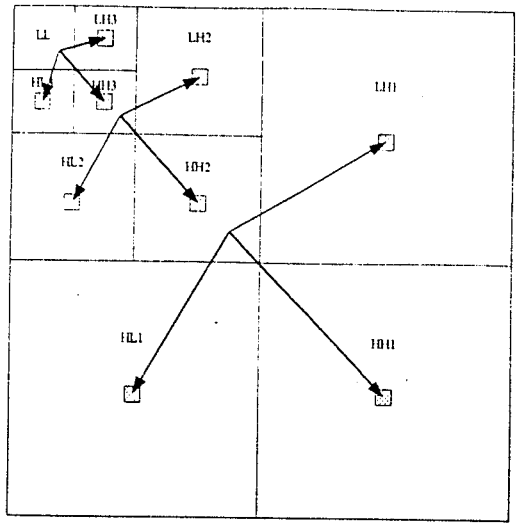


Fig. 7. Relationship among Cousins

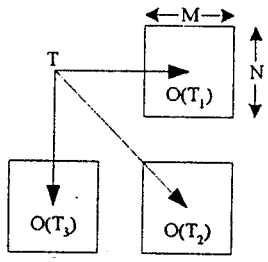


Fig. 8. Formation of Block Trees

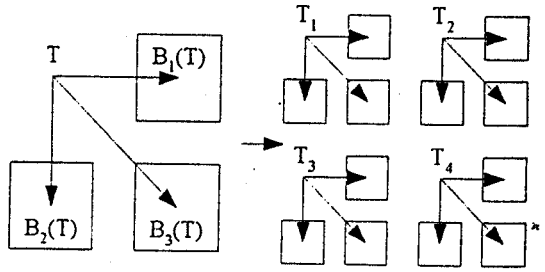


Fig. 9. Set Partitioning in Block Trees

In the initialization pass, we divide each subband into smaller blocks and construct block trees, as mentioned before. Then, we calculate the initial quantization step n in the transform image X.

$$n = \left\lfloor \log_2 \left(\max_{\forall (i,j)\in X} |c_{i,j}| \right) \right\rfloor \quad (3)$$

After we determine entries of LIP, LIB, and LSP, all coefficients in the lowest subband are used as initial LIP entries, and all block trees are used as initial LIB entries. LSP are set to null in this initialization pass.

In the sorting pass, we identify the significance of pixels and trees in LIP and LIB. When a pixel in LIP is

identified to be significant, the sign of the pixel is also identified, and the pixel is moved to LSP. Similarly, we evaluate the significance of trees in LIB sequentially. When a tree T in LIB is identified to be significant, it is partitioned according to the partitioning rule, and removed from LIB. All partitioned small trees T_i, $i \in \{1,2,3,4\}$, are added back to LIB. When the size of blocks $B_i(T)$, $i \in \{1,2,3\}$, is 4×4, significance of all pixels in the tree T is identified and the tree is removed from LIB. If a pixel is identified to be significant, the sign of the pixel is identified and the pixel is added to LSP. Otherwise, the pixel is added to LIP.

In the refinement pass, the n-th most significant bit (MSB) of entries in LSP is identified with respect to n. In this pass, we do not consider entries included in the last sorting pass with the same n.

In the update of quantization step, we decrease the quantization step n by one, and repeat the coding step from the sorting pass.

3.5. Entropy Coding

We use an adaptive arithmetic coder [9] to encode motion vectors and significance identification in the residual quantization. It is well known that the adaptive arithmetic coder is computationally efficient. The adaptive arithmetic coder estimates the probability of significance of coefficients with a state machine and then arithmetically encodes it. The minimum code-length l of the sequence in bits is given by

$$l = -\log_2 \prod_{i=1}^{n} p(x_i \mid x_{i-1}, x_{i-2}, \ldots, x_1) \qquad (4)$$

where $p(x_i \mid x_{i-1}, x_{i-2}, \ldots, x_1)$ is a conditional probability of x_i given $x_{i-1}, x_{i-2}, \ldots, x_1$. However, $p(x_i \mid x_{i-1}, x_{i-2}, \ldots, x_1)$ is generally unknown in practice. Therefore, we have to estimate $p(x_i \mid x_{i-1}, x_{i-2}, \ldots, x_1)$ based on the past observations in the coding process. A set of past observations on which the probability of the current symbol is conditioned is called as the modeling context [10].

We use seven different contexts for encoding the significance identification of pixels and four contexts for coding of signs. We use one parent coefficient and neighborhoods located to the north, west, south, east, northwest, and northeast directions of the current coefficient to encode the significance identification of pixels. For coding of signs, we use neighborhoods located to the north, west, south, and east directions of the current coefficient. We encode the significance identification of each tree, the refinement bit and motion vectors with one fixed context. The bounding box information is encoded with no probability model. All of the above contexts are not shared among different wavelet scales. For unavailable neighboring or parent coefficients, the corresponding context bits are set to zero.

4. EXPERIMENTAL RESULTS

Experiments are performed with a Pentium 800 MHz personal computer (PC) with 128M RAM. Each frame of test sequences is decomposed by dyadic 9/7-tap biorthogonal wavelet filters. As a performance measure, we use the peak signal-to-noise ratio (PSNR) that is defined by

$$PSNR = 10\log_{10}\left(\frac{255^2}{MSE}\right) \quad dB \qquad (4)$$

4.1. Performance of Intra-Frame Coding

Fig. 10 compares performance of intra-frame coding by the proposed and SPIHT schemes. For performance comparison, two popular monochrome images, BOATS and BARBARA, of size 576×720 pixels are used. Our experimental results demonstrate that the proposed scheme provides good results and often outperforms SPIHT. For BOAT, the proposed scheme provides slightly better objective quality than SPIHT. However, the proposed scheme significantly outperforms SPIHT for BARBARA that has a lot of high-frequency components.

4.2. Performance of Video Coding

Experiments of video coding are performed on a popular 4:2:0, 30-fps color sequence, MOTHER AND DAUGHTER, of size 176×144 pixels. We encode the video sequence at 30 and 60 kbps (kbits per second) with 10 fps (frame per second). Since the test sequence has 30 fps, every third frame is encoded and reconstructed. Fig. 11 and Fig. 12 show performances of the proposed video coding schemes at 30 and 60 kbps, respectively. As shown in Fig. 11 and Fig. 12, our experimental results demonstrate that proposed coding schemes provide good performance. Especially, the multiresolution motion-compensated video coding scheme provides the best performance.

Table 1 lists average PSNR values of the proposed multiresolution motion-compensated and 3-D SPIHT [2] video coding schemes at 30 kbps and 60 kbps respectively. As shown in Table 1, the proposed video coding scheme provides slightly higher PSNR values than 3-D SPIHT [2].

Table 2 shows the encoding time of multiresolution motion compensated video coding scheme for 100 frames. Experimental results demonstrate that the proposed video coding scheme is simple and suitable for real-time video applications.

5. CONCLUSIONS

In this paper, we have proposed new video coding schemes for real-time transmission. In order to exploit temporal redundancies in video sequences, we use the three-step block matching ME and MRME algorithms. The prediction error frame is quantized by a new method exploiting the relationship among wavelet coefficients. The embedded property of the quantization method enables an accurate rate control. The proposed video coding schemes have a low computational complexity. Although they are simple, the rate-distortion performance of proposed video coding schemes is competitive to other wavelet-based video coders. Proposed video coding schemes can provide SNR scalability from the embedded nature of residual quantization. In addition, memory usage of the proposed schemes is lower than that of 3-D subband video coders.

ACKNOWLEDEGMENTS

This work was supported in part by the Korea Science and Engineering Foundation (KOSEF) through the Ultra-Fast Fiber-Optic Networks (UFON) Research Center at Kwangju Institute of Science and Technology (K-JIST), and in part by the Ministry of Education (MOE) through the Brain Korea 21 (BK21) project.

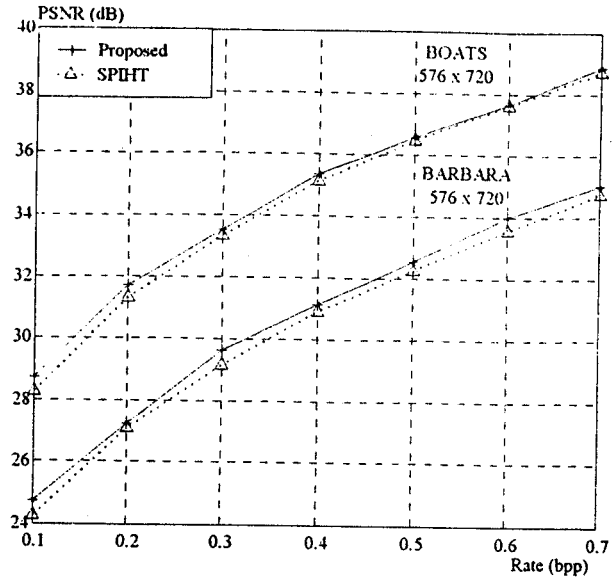


Fig. 10. Performance Comparisons for Intra-Frame

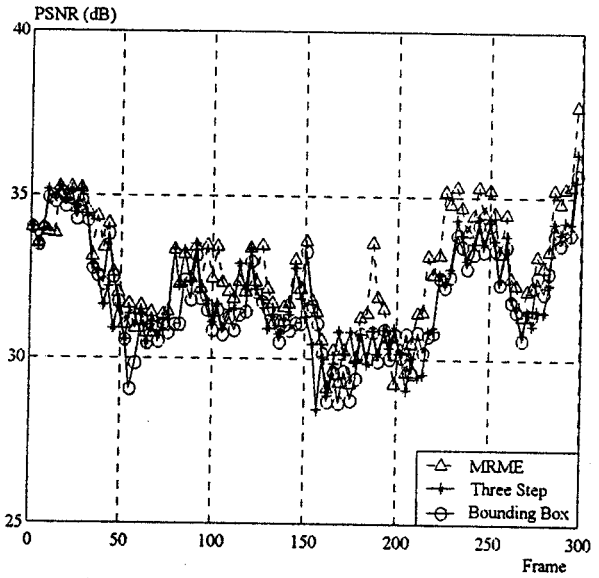


Fig. 11. Performance Comparisons at 30 kbps

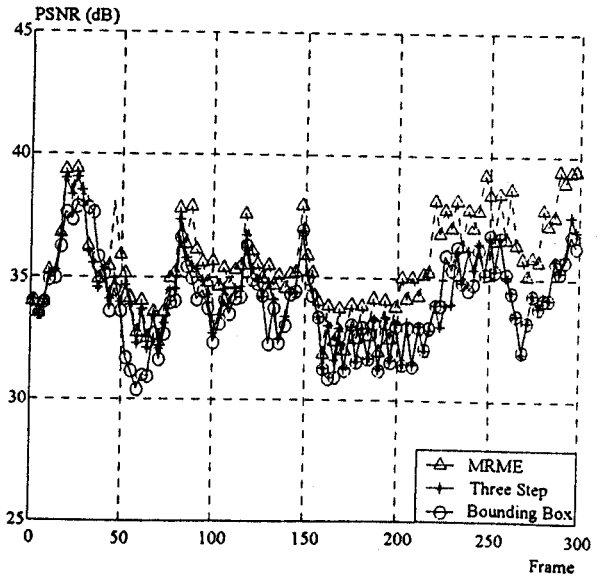


Fig. 12. Performance Comparisons at 60 kbps

Table 1. Performance Comparisons of Average PSNR

Coding Scheme	Compression Ratio	Average PSNR
MRME	30 kbps	32.8 dB
	60 kbps	35.6 dB
3-D SPIHT	30 kbps	32.7 dB
	60 kbps	35.5 dB
MC 3-D SPIHT	30 kbps	32.7 dB
	60 kbps	35.6 dB

Table 2. Encoding Time

Compression Ratio	30 kbps	60 kbps
Encoding Time	7.36 sec	8.46 sec

References

[1] J. Y. Tham, S. Ranganath and A. A. Kassim, "Highly Scalable Wavelet-based Video Codec for Very Low Bit-Rate Environment", *IEEE Journal on Selected Areas in Comm.*, vol. 16, no. 1, pp. 12-27, Jan. 1998.

[2] B. J. Kim, Z. Xiong and W. A. Pearlman, "Low Bit-Rate Scalable Video Coding with 3-D Set Partitioning in Hierarchical Trees", *IEEE Trans. on Circuits and Systems for Video Tech.*, vol. 10, no. 8, pp. 1374-1387, Dec. 2000.

[3] J. Shapiro, "Embedded Image Coding using Zerotrees of Wavelet Coefficients", *IEEE Trans. on Signal Proc.*, vol. 41, no. 12, pp. 3445-3462, Dec.1993.

[4] A. Said and W. Pearlman, "A New, Fast and Efficient Image Codec based on Set Partitioning in Hierarchical Trees", *IEEE Trans. on Circuits and Systems for Video Tech.*, vol. 6, no. 3, pp. 243-250, June 1996.

[5] J. Y. Lee, H. S. Oh and S. J. Ko, "Motion-Compensated Layered Video Coding for Playback Scalability", *IEEE Trans. on Circuits and Systems for Video Tech.*, vol. 11, no. 5, pp. 619-628, May 2001.

[6] S. Mallat, "A Theory for Multiresolution Signal Decomposition: The Wavelet Representation", *IEEE Trans. on Pattern Anal. Mach. Intell.*, vol. 11, no. 7, pp. 674-693, July 1989.

[7] K. R. Rao and J. J. Hwang, *Techniques and Standards for Image, Video, and Audio Coding*, Prentice Hall, 1996.

[8] S. Zafer, Y. Zhang, and B. Jabbari, "Multiscale Video Representation using Multiresolution Motion Compensation and Wavelet Decomposition", *IEEE Journal on Selected Areas in Comm.*, vol. 11, no. 1, pp. 24-35, Jan. 1993.

[9] W. B. Pennebaker and J. L. Mitchell, *JPEG Still Image Data Compression Standards*, Van Nostrand Reinhold Publishers, New York, 1993.

[10] X. Wu and J. Chen, "Context Modeling and Entropy Coding of Wavelet Coefficients for Image Compression", *IEEE Int'l Conf. on Acoustics, Speech and Signal Proc.*, vol 4, pp. 3097-3100, April 1997.