Hindawi Publishing Corporation
EURASIP Journal on Image and Video Processing
Volume 2007, Article ID 42761, 15 pages
doi:10.1155/2007/42761

Research Article

Video Coding Using 3D Dual-Tree Wavelet Transform

Beibei Wang,[1] Yao Wang,[1] Ivan Selesnick,[1] and Anthony Vetro[2]

[1] *Electrical and Computer Engineering Department, Polytechnic University, Brooklyn, NY 11201, USA*
[2] *Mitsubishi Electric Research Laboratories, Cambridge, MA 02139, USA*

Received 14 August 2006; Revised 14 December 2006; Accepted 5 January 2007

Recommended by Béatrice Pesquet-Popescu

This work investigates the use of the 3D dual-tree discrete wavelet transform (DDWT) for video coding. The 3D DDWT is an attractive video representation because it isolates image patterns with different spatial orientations and motion directions and speeds in separate subbands. However, it is an overcomplete transform with 4 : 1 redundancy when only real parts are used. We apply the noise-shaping algorithm proposed by Kingsbury to reduce the number of coefficients. To code the remaining significant coefficients, we propose two video codecs. The first one applies separate 3D set partitioning in hierarchical trees (SPIHT) on each subset of the DDWT coefficients (each forming a standard isotropic tree). The second codec exploits the correlation between redundant subbands, and codes the subbands jointly. Both codecs do not require motion compensation and provide better performance than the 3D SPIHT codec using the standard DWT, both objectively and subjectively. Furthermore, both codecs provide full scalability in spatial, temporal, and quality dimensions. Besides the standard isotropic decomposition, we propose an anisotropic DDWT, which extends the superiority of the normal DDWT with more directional subbands without adding to the redundancy. This anisotropic structure requires significantly fewer coefficients to represent a video after noise shaping. Finally, we also explore the benefits of combining the 3D DDWT with the standard DWT to capture a wider set of orientations.

1. INTRODUCTION

Video coding based on 3D wavelet transforms has the potential of providing a scalable representation of a video in spatial resolution, temporal resolution, and quality. For this reason, extensive research efforts have been undertaken to develop efficient wavelet-based scalable video codecs. Most of these studies employ the standard separable discrete wavelet transform. Because directly applying the wavelet transform in the time dimension does not lead to an efficient representation when the underlying video contains objects moving in different directions, motion-compensated temporal filtering is deployed in state-of-the-art wavelet-based video coders [1–4]. Motion compensation can significantly improve the coding efficiency, but it also makes the encoder very complex. Furthermore, the residual signal resulting from block-based motion compensation is very blocky and cannot be represented by a frame-based 2D DWT efficiently. Hence, the newest scalable video coding standard [5] still uses block-based transforms for coding the residual.

An important recent development in wavelet-related research is the design and implementation of 2D multiscale transforms that represent edges more efficiently than does the separable DWT. Kingsbury's dual-tree complex wavelet transform (DT-CWT) [6] and Do's contourlet transform [7] are examples. The DT-CWT is an overcomplete transform with limited redundancy ($2^m : 1$ for m-dimensional signals). This transform has good directional selectivity and its subband responses are approximately shift invariant. The 2D DT-CWT has given superior results for image processing applications compared to the DWT [6, 8]. In [9], the authors developed a subpixel transform domain motion-estimation algorithm based on the 2D DT-CWT, and a maximum phase correlation technique. These techniques were incorporated in a video codec that has achieved a performance comparable to H.263 standard.

Selesnick and Li described a 3D version of the dual-tree wavelet transform and showed that it possesses some motion selectivity [10]. The design and the motion selectivity of dual-tree filters are described in [10, 11]. Although the separable transforms can be efficiently computed, the separable implementations of multidimensional (*MD*) transforms mix edges in different directions which leads to annoying visual artifacts when the coefficients are quantized. The 3D DDWT is implemented by first applying separable transforms and then combining subband signals with simple

linear operations. So even though it is nonseparable and free of some of the limitations of separable transforms, it inherits the computational efficiency of separable transforms.

A core element common to all state-of-the art video coders is motion-compensated temporal prediction, which is the main contributor to the complexity and error sensitivity of a video encoder. Because the subband coefficients associated with the 3D DDWT directly capture moving edges in different directions, it may not be necessary to perform motion estimation explicitly. This is our primary motivation for exploring the use of the 3D DDWT for video coding.

The major challenge in applying the 3D complex DDWT for video coding is that it is an overcomplete transform with 8 : 1 redundancy. In our current study, we choose to retain only the real parts of the wavelet coefficients, which still leads to perfect reconstruction, while retaining the motion selectivity. This reduces the redundancy to 4 : 1 [10].

To reduce the number of coefficients necessary for representing an image, Reeves and Kingsbury proposed an iterative projection-based noise-shaping (NS) scheme [8], which modifies previously chosen large coefficients to compensate for the loss of small coefficients. We have found that noise shaping applied to the 3D DDWT can yield a more compact set of coefficients than from the 3D DWT [12]. The fact that noise shaping can reduce the number of coefficients to below that required by the DWT (for the same video quality) is very encouraging.

To code the retained coefficients, we must specify both the locations and amplitudes (sign and magnitude) of the retained coefficients. 3D SPIHT is a well-known embedded video-coding algorithm [13], which applies the 3D DWT to a video directly, without motion compensation, and offers spatial, temporal, and PSNR-scalable bitstreams. The 3D DDWT coefficients can be organized into four trees, each with the same structure as the standard DWT. Our first DDWT-based video codec (referred as DDWT-SPIHT) applies 3D SPIHT to each DDWT tree. This codec gives better rate-distortion (R-D) performances than the 3D DWT.

With the standard nonredundant DWT, there is very little correlation among coefficients in different subbands, and DWT-based wavelet coders all code different subbands separately. Because the DDWT is a redundant transform, we should exploit the correlation between DDWT subbands in order to achieve high coding efficiency. Through statistical analysis of the DDWT data, we found that there is strong correlation about locations of significant coefficients, but not about the magnitude and signs.

Based on the above findings, we developed another video codec referred to as DDWTVC. It codes the significant bits across subbands jointly by vector arithmetic coding, but codes the sign and magnitude information using context-based arithmetic coding within each subband. Compared to the 3D SPIHT coder on the standard DWT, the DDWTVC also offers better rate-distortion performance, and is superior in terms of visual quality [14]. Compared to the first proposed DDWT-SPIHT, DDWTVC has comparable and slightly better performance.

As with the standard separable DWT, the 3D DDWT applies an isotropic decomposition structure, that is, for each stage, the decomposition only continues in the low-frequency subband LLL, and for each subband the number of decomposition levels is the same for all spatial and temporal directions. However, not only the low-frequency subband LLL, but also subbands LLH, HLL, LHL, and so forth, include important low-frequency information, and may benefit from further decomposition. Typically, more spatial decomposition stages produce noticeable gain for video processing. But additional temporal decomposition does not bring significant gains and incurs additional memory cost and processing delay.

If a transform allows decomposition only in one direction when a subband is further divided, it will generate rectangular frequency tilings, and is thus called anisotropic [15, 16]. Based on these observations, we propose a new anisotropic DDWT, and examine its application to video coding. The experimental results show that the new anisotropic decomposition is more effective for video representation in terms of PSNR versus the number of retained coefficients.

Although the DDWT has wavelet bases in more spatial orientations than the DWT, it does not have bases in the horizontal and vertical directions. Recognizing this deficiency, we propose to combine the 3D DDWT and DWT, to capture directions represented by both the 3D DDWT and the DWT. Combining the 3D DWT and DDWT shows slight gains over using 3D DDWT alone.

To summarize the main contributions, the paper mainly focuses on video processing using a novel edge and motion selective wavelet transform, the 3D DDWT. In this paper, we demonstrate how to select the significant coefficients of the DDWT to represent video. Two iterative algorithms for coefficient selection, noise shaping, and matching pursuit are examined and compared. We propose and validate the hypothesis that only a few bases of 3D DDWT have significant energy for an object feature. Based on these properties, two video codecs using the DDWT are proposed and tested on several standard video sequences. Finally, two extensions of the DDWT are proposed and examined for video representation.

The paper is organized as follows. Section 2 briefly introduces the 3D DDWT and its advantage. Section 3 describes how to select significant coefficients for video coding. Section 4 investigates the correlation between wavelet bases at the same spatial/temporal location for both the significance map and the actual coefficients. Section 5 describes the two proposed video codecs based on the DDWT, and compares the coding performance to 3D SPIHT with the DWT. The scalability of the proposed video codec is discussed in Section 6. Section 7 describes the new anisotropic wavelet decomposition and how to combine 3D DDWT and DWT. The final section summarizes our work and discusses future work for video coding using the 3D DDWT.

2. 3D DUAL-TREE WAVELET TRANSFORM

The design of the 3D dual-tree complex wavelet transform is described in [10]. At the core of the wavelet design is a Hilbert pair of bases, ψ_h and ψ_g, satisfying $\psi_g(t) = \mathcal{H}(\psi_h(t))$. They can be constructed using a Daubechies-like algorithm for constructing Hilbert pairs of short orthonormal (and biorthogonal) wavelet bases. The complex 3D wavelet is defined as $\psi(x,y,z) = \psi(x)\psi(y)\psi(z)$, where $\psi(x) = \psi_h(x) + j\psi_g(x)$. The real part of $\psi(x,y,z)$ can be represented as

$$\begin{aligned}\psi_a &= \text{RealPart}\,\{\psi(x,y,z)\}\\ &= \psi_1(x,y,z) - \psi_2(x,y,z) - \psi_3(x,y,z) - \psi_4(x,y,z),\end{aligned} \tag{1}$$

where

$$\psi_1(x,y,z) = \psi_h(x)\psi_h(y)\psi_h(z), \tag{2}$$

$$\begin{aligned}\psi_2(x,y,z) &= \psi_g(x)\psi_g(y)\psi_h(z),\\ \psi_3(x,y,z) &= \psi_g(x)\psi_h(y)\psi_g(z),\\ \psi_4(x,y,z) &= \psi_h(x)\psi_g(y)\psi_g(z).\end{aligned} \tag{3}$$

Note that $\psi_1(x,y,z)$, $\psi_2(x,y,z)$, $\psi_3(x,y,z)$, $\psi_4(x,y,z)$ are four separable 3D wavelet bases, and each can produce one DWT tree containing 1 low subband and 7 high subbands. Because ψ_a is a linear combination of these four separable bases, the wavelet coefficients corresponding to ψ_a can be obtained by linearly combining the four DWT trees, yielding one DDWT tree containing 1 low subband, and 7 high subbands.

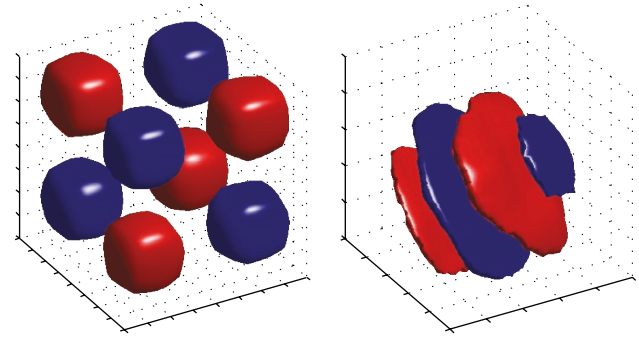
To obtain the remaining DDWT subbands, we take in addition to the real part of $\psi(x)\psi(y)\psi(z)$, the real part of $\psi(x)\psi(y)\overline{\psi(z)}$, $\psi(x)\overline{\psi(y)}\psi(z)$, $\psi(x)\overline{\psi(y)}\overline{\psi(z)}$, where the overline represents complex conjugation. This gives the following orthonormal combination matrix:

$$\begin{bmatrix}\psi_a(x,y,z)\\ \psi_b(x,y,z)\\ \psi_c(x,y,z)\\ \psi_d(x,y,z)\end{bmatrix} = \frac{1}{2}\begin{bmatrix}1 & -1 & -1 & -1\\ 1 & -1 & 1 & 1\\ 1 & 1 & -1 & 1\\ 1 & 1 & 1 & -1\end{bmatrix}\begin{bmatrix}\psi_1(x,y,z)\\ \psi_2(x,y,z)\\ \psi_3(x,y,z)\\ \psi_4(x,y,z)\end{bmatrix}. \tag{4}$$

By applying this combination matrix to the four DWT trees, we obtain four DDWT trees, containing a total of 4 low subbands and 28 high subbands. Each high subband has a unique spatial orientation and motion.

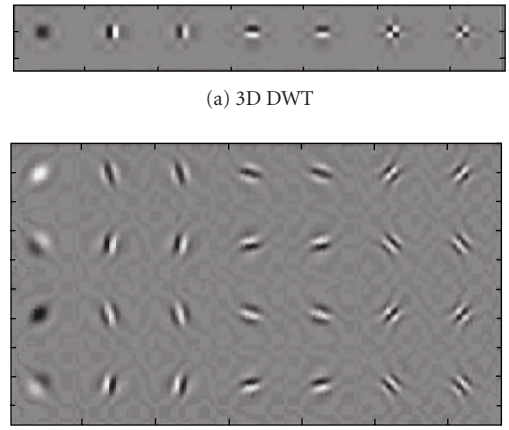
Figure 1 shows the isosurfaces of a selected wavelet from both the DWT Figure 1(a) and the DDWT Figure 1(b). Like a contour plot, the points on the surfaces are points where the function is equal valued. As illustrated in Figure 1, the wavelet associated with the separable 3D transform has the checkerboard phenomenon, a consequence of mixing of orientations. The wavelet associated with the dual-tree 3D transform is free of this effect.

Figure 2 shows all the wavelets in a particular temporal frame for both the DWT and DDWT. In Figure 2(b), the wavelets in each row correspond to 7 high subbands contained in one DDWT tree. For 3D DDWT, each subband



(a) (b)

Figure 1: Isosurfaces of a typical 3D DWT basis (a) and a typical 3D DDWT basis (b).



(a) 3D DWT

(b) 3D DDWT

Figure 2: Typical wavelets associated with (a) the 3D DWT and (b) 3D DDWT in the spatial domain.

corresponds an image pattern with a certain spatial orientation and motion direction and speed. The motion direction of each wavelet is orthogonal to the spacial orientation. Note that the wavelets with the same spatial orientation in Figure 2(b) have different motion directions and/or speeds. For example, the second and third wavelets in the top row move in opposite directions. As can be seen, the 3D DWT can represent the horizontal and vertical features well, but it mixes two diagonal directions in a checkerboard pattern. The 3D DDWT is free of the checkerboard effect, but it does not represent the vertical and horizontal orientations in pursuit of other directions. The 3D DDWT has many more subbands than the 3D DWT (28 high subbands instead of 7, 4 low subbands instead of 1). The 28 high subbands isolate 2D edges with different orientations that are moving in different directions.

Because different wavelet bases of the DDWT represent object features with different spatial orientations and motions, it may not be necessary to perform motion-compensated filtering, which is a major contributor to the

computational load of a block-based hybrid video coder and wavelet-based coders using separable DWT. If a video sequence contains differently oriented edges moving in different directions and speeds, coefficients for the wavelets with the corresponding spatial orientation and motion patterns will be large. By applying the 3D DDWT to a video sequence directly, and coding large wavelet coefficients, we are essentially representing the underlying video as basic image patterns (varying in spatial orientation and frequency) moving in different ways. Such a representation is naturally more efficient than using a separable wavelet transform directly, with which a moving object in arbitrary directions that are not characterized by any specific orientation and/or motion will likely contribute many small coefficients associated with wavelets. Directly applying the 3D DDWT to the video is also more computationally efficient than first performing motion estimation and then applying a separable wavelet transform along the motion trajectory, and finally applying a 2D wavelet transform to the prediction error image. Finally, because no motion information is coded separately, the resulting bitstream can be fully scalable.

For the simulation results presented in the paper, 3-level wavelet decompositions are applied for both the 3D DDWT and 3D DWT. The 3D DWT uses the Daubechies (9, 7)-tap filters. For the DDWT, the Daubechies (9, 7)-tap filters are used at the first level, and Qshift filters in [6] are used beyond level 1.

3. ITERATIVE SELECTION OF COEFFICIENTS

For video coding, the 4 : 1 redundancy of the 3D DDWT (real parts) [10] is a major challenge. However, an overcomplete transform is not necessarily ineffective for coding because a redundant set provides flexibility in choosing which basis functions to use in representing a signal. Even though the transform itself is redundant, the number of the critical coefficients that must be retained to represent a video signal accurately can be substantially smaller than that obtained with standard non-redundant separable transform.

The selection of significant coefficients from nonorthogonal transforms, like DDWT, is very different from the orthogonal transforms, like DWT. Because the bases are not orthogonal, one should not simply keep all the coefficients that are above a certain threshold and delete those that are less than the threshold. In this section, we compare the efficiency of two coefficient selection schemes, matching pursuit and noise shaping.

3.1. Matching-pursuit algorithm

Matching pursuit (MP) is a greedy algorithm to decompose any signal into a linear expansion of waveforms that are selected from a redundant dictionary of functions [17, 18]. These waveforms are selected to best match the signal structures. The matching-pursuit (MP) algorithm is well known for video coding with overcomplete representations [19].

With the matching-pursuit (MP) algorithm, the significant coefficients are chosen iteratively. Starting with all the original coefficients for a given signal, the one with the largest magnitude is chosen. The error between the original signal and the one reconstructed using the chosen coefficient is then transformed (without using the previously chosen basis function). The largest coefficient is then chosen from the resulting coefficients, and a new error image is formed and transformed again. This process repeats until the desired number of coefficients is chosen.

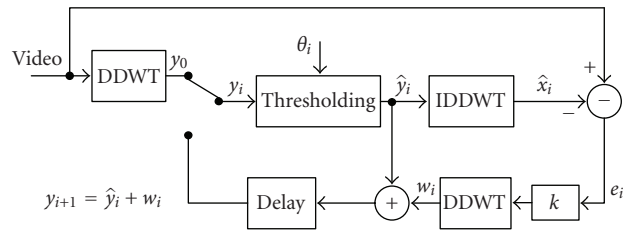


Figure 3: Noise-shaping algorithm.

Because only one coefficient is chosen in each iteration, the computation is very slow. Our simulations (see Section 3.3) show that the matching pursuit only has slight gain over using the N largest original DDWT coefficients directly.

3.2. Noise-shaping algorithm

For nonorthogonal transforms like the DDWT, deleting insignificant coefficients can be modelled as adding noise to the other coefficients. In [20], the effect of additive noise in oversampled filter bank systems is examined. Much of the algebra for the overcomplete DDWT transform analysis is similar with the polyphase domain analysis in [20]. Recognizing this, Reeves and Kingsbury proposed an iterative projection-based noise shaping (NS) scheme [8]. As illustrated in Figure 3, the coefficients are obtained by running the iterative projection algorithm with a preset initial threshold, and gradually reducing it until the number of remaining coefficients reaches N, a target number. In each iteration, the error coefficients are multiplied by a positive real number k and added back to the previously chosen large coefficients, to compensate for the loss of small coefficients due to thresholding.

NS requires substantially fewer computations than MP, to yield the set of coefficients that can yield the same representation accuracy. This is because with NS, many coefficients can be chosen in one iteration (those that are larger than a threshold), whereas with MP, only one coefficient is chosen in each iteration.

Reeves and Kingsbury have shown that noise shaping applied to 2D DT-CWT can yield a more compact set of coefficients than from the 2D DWT [8]. Our research [12] verifies that NS has the similar effect on video data transformed with the 3D DDWT. Our simulation results in Section 3.3 show that the NS algorithm leads to significantly more accurate representation of the original signal than the MP algorithm with the same number of coefficients, while requiring significantly less computation.

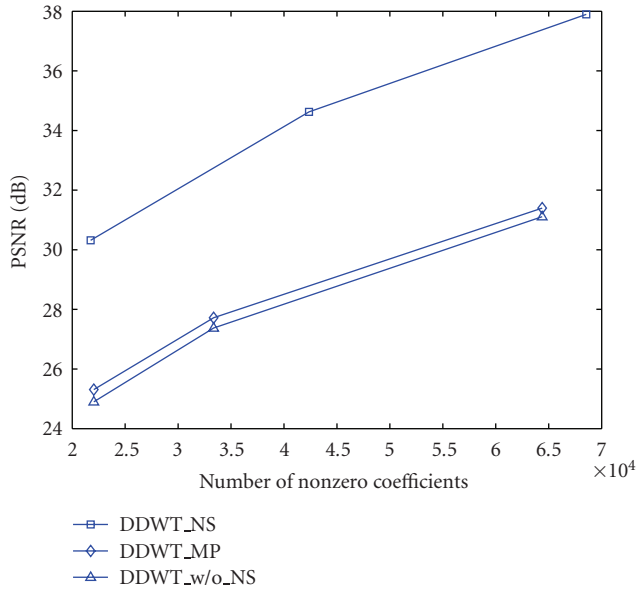
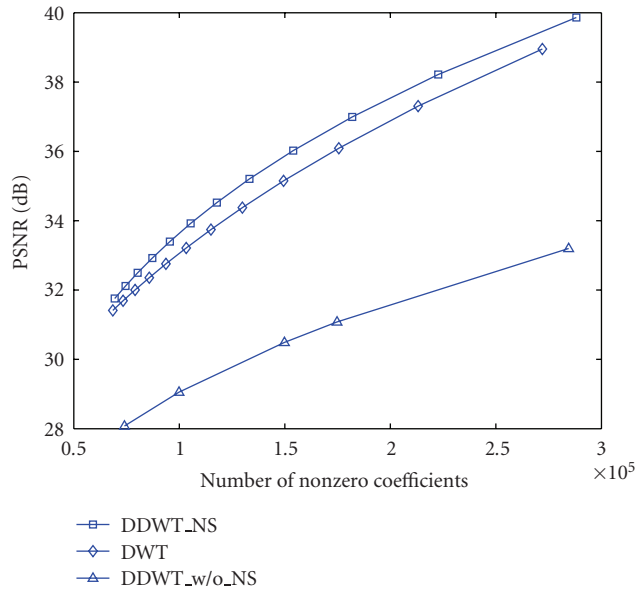


Figure 4: PSNR (dB) versus number of nonzero coefficients for the DDWT using noise shaping (DDWT_NS, top curve), using matching pursuit (DDWT_MP, middle curve), without noise shaping (DDWT_w/o_NS, lower curve) for a small size test sequence.

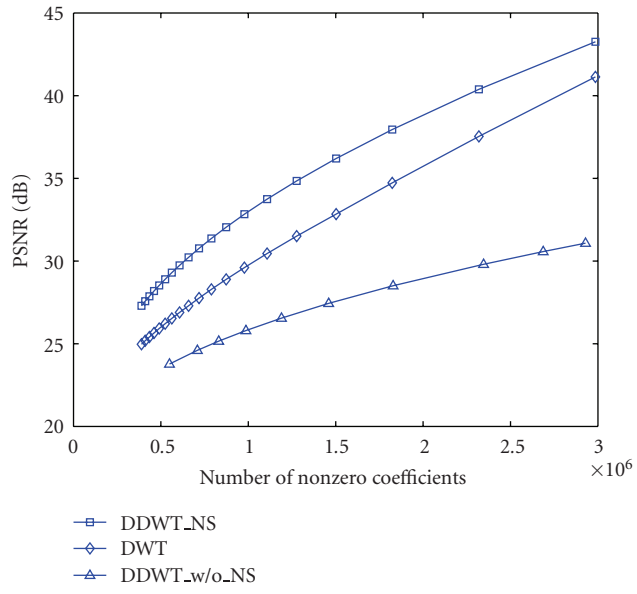
3.3. Simulation results

For a given number of coefficients to retain, N, the results designated below as DWT and DDWT_w/o_NS are obtained by simply choosing the N largest ones from the original coefficients. DDWT_MP is obtained by selecting coefficients with MP. With DDWT_NS, the coefficients are obtained by running the iterative projection noise-shaping algorithm with a preset initial threshold 256.0, and gradually reducing it until the number of remaining coefficients reaches N. The reducing step is set as 1. The energy compensation parameter k is set as 1.8, which gives the best performance for all tested video sequences experimentally. Figure 4 compares the reconstruction quality (in terms of PSNR) using the same number of retained coefficients (original values without quantization) using different methods. Because the MP algorithm takes tremendous computation to deduce a large set of coefficients, this comparison is done using a small size ($80\times80\times80$ pixels) video sequence. The DDWT MP provides only marginal gain over simply choosing the largest N coefficients (DDWT_w/o_NS). On the other hand, DDWT_NS yielded much better image quality (5-6 dB higher) than DDWT_w/o_NS with the same number of coefficients.

Figure 5 compares the reconstruction quality (in terms of PSNR) using the same number of retained coefficients using different methods (except for DDWT_MP) for two standard test sequences. The testing sequence "Foreman" is QCIF and "Mobile Calendar" is CIF. Both sequences have the same frame rate 30 fps and 80 frames are used for simulations. Figure 5 shows that although the raw number of coefficients with 3D DDWT is 4 times more than DWT, this number



(a) Foreman (QCIF)



(b) Mobile Calendar (CIF)

Figure 5: PSNR (dB) versus number of nonzero coefficients for the DDWT using noise shaping (DDWT_NS, upper curve), the DWT (middle curve), and the DDWT without noise shaping (DDWT_w/o_NS, lower curve).

can be reduced substantially by noise shaping. In fact, with the same number of retained coefficients, DDWT_NS yields higher PSNR than DWT. For "Foreman," 3D DDWT_NS has a slightly higher PSNR than the DWT (0.3–0.7 dB), and is 4–6 dB better than DDWT_w/o_NS. For "Mobile Calendar," the DDWT_NS is 1.5–3.4 dB better than the DWT. The superiority of DDWT for "Mobile Calendar" sequence can be attributed to the many directional features with different orientations and consistent small motions in the sequence.

Figure 5 shows that with DDWT_NS, we can use fewer coefficients to reach a desired reconstruction quality than DWT. However, this does not necessarily mean that DDWT_NS will require fewer bits for video coding. This is because we need to specify both the location as well as the value of each retained coefficient. Because DDWT has 4 times more coefficients, specifying the location of a DDWT coefficient requires more bits than specifying that of a DWT coefficient. The success of a wavelet-based coder critically depends on whether the location information can be coded efficiently. As shown in Section 4.1, there are strong correlations among the locations of significant coefficients in different subbands. The DDWTVC codec to be presented in Section 5.2 exploits this correlation in coding the location information.

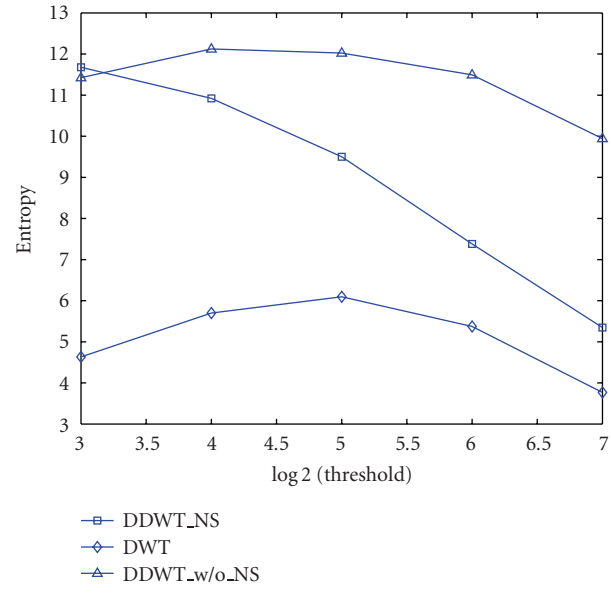
4. THE CORRELATION BETWEEN SUBBANDS

Because the DDWT is a redundant transform, the subbands produced by it are expected to have nonnegligible correlations. Since wavelet coders code the location and magnitude information separately, we examine the correlation in the location and magnitude separately.

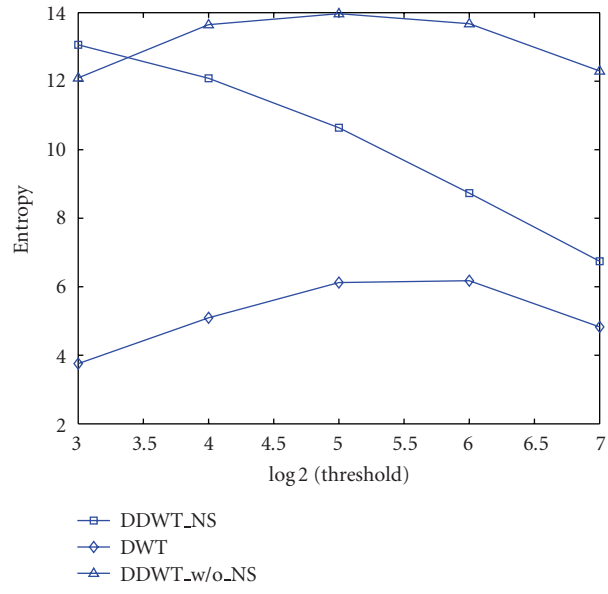
4.1. Correlation in significant maps

We hypothesize that although the 3D DDWT has many more subbands, only a few subbands have significant energy for an object feature. Specifically, an oriented edge moving with a particular velocity is likely to generate significant coefficients only in the subbands with the same or adjacent spatial orientation and motion pattern. On the other hand, with the 3D DWT, a moving object in arbitrary directions that are not characterized by any specific wavelet basis will likely contribute to many small coefficients in all subbands. To validate this hypothesis, we compute the entropy of the vector consisting of the significance bits at the same spatial/temporal location across 28 high subbands. The significance bit in a particular subband is either 0 or 1 depending on whether the corresponding coefficient is below or above a chosen threshold. The entropy of the significance vector will be close to 28 if there is not much correlation between the 28 subbands. On the other hand, if the pattern that describes which bases are simultaneously significant is highly predictable, the entropy should be much lower than 28. Similarly, we calculate the entropy of the significance bits across the 7 high subbands of DWT, and compare it to the maximum value of 7.

Figure 6 compares the vector entropy for significant maps among the DWT, DDWT_NS, and DDWT_w/o_NS, for varying thresholds from 128 to 8. The results shown here are for the top scale only—other scales follow the same trend. We see that, with DDWT, even without noise shaping, the vector entropy is much lower than 28. Moreover, noise shaping helps reduce the entropy further. In contrast, with DWT, the vector entropy is close to 7 at some threshold values. This study validates our hypothesis that the significance maps across the 28 subbands of DDWT are highly correlated.



(a) Foreman (QCIF)



(b) Mobile Calendar (CIF)

Figure 6: The vector entropy of significant maps using the 3D DWT, the DDWT_NS, and the DDWT_w/o_NS, for the top scale.

4.2. Correlation in coefficient values

In addition to the correlation among the significance maps of all subbands, we also investigate the correlation between the actual coefficient values. Strong correlation would suggest vector quantization or predictive quantization among the subbands. Towards this goal, we compute the correlation matrix and variances of the 28 high subbands. Figure 7 illustrates the correlation matrices for the finest scale, for both the DDWT_w/o_NS and DDWT_NS. We note that the correlation patterns in other scales are similar to this top scale.

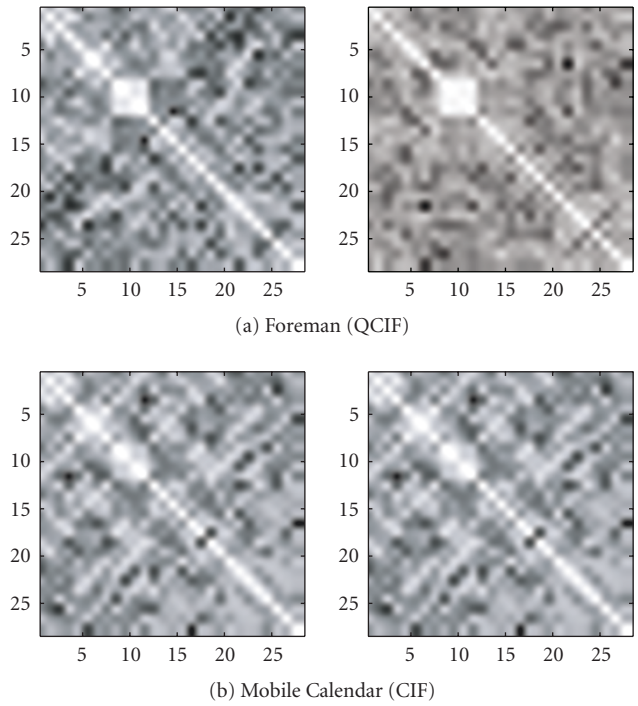


Figure 7: The correlation matrices of the 28 subbands of 3D DDWT_w/o_NS (left) and DDWT_NS (right). The grayscale is logarithmically related to the absolute value of the correlation. The brighter colors represent higher correlation.

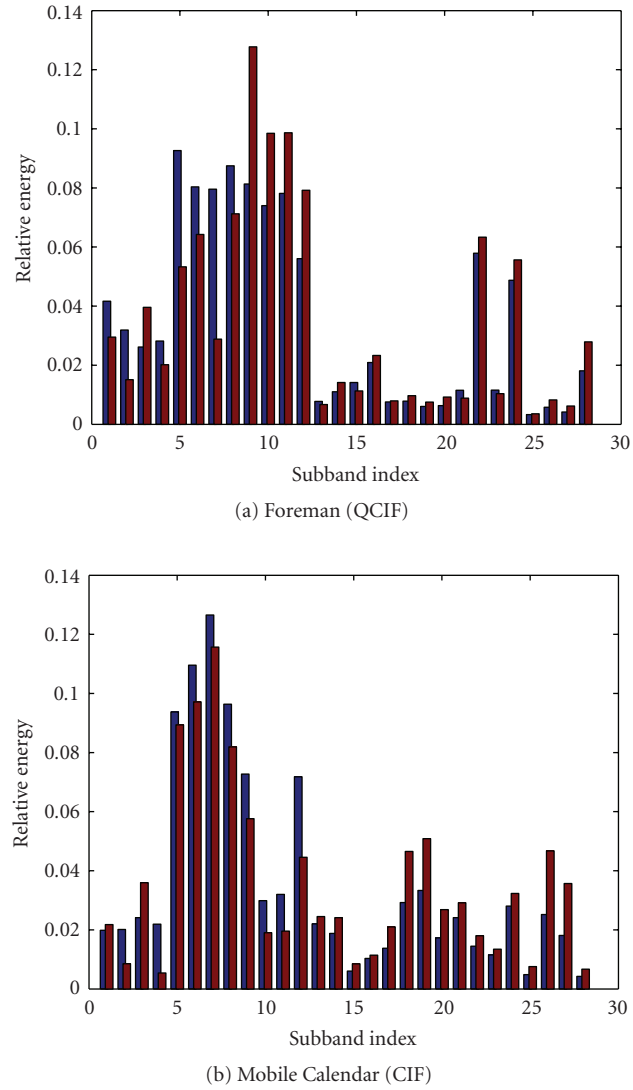


Figure 8: The relative energy of 3D DDWT 28 subbands with (the right column in each subband) and without noise shaping (the left column).

From these correlation matrices, we find that only a few subbands have strong correlation, and most other subbands are almost independent. After noise shaping, the correlation between subbands is reduced significantly. A greater number of subbands are almost independent from each other. It is interesting to note that, for the "Foreman" sequence (which has predominantly vertical edges and horizontal motion), bands 9–12 (the four subbands in the third column of Figure 2) are highly correlated before and after noise shaping. These four bands have edges close to vertical orientations but all moving in the horizontal direction. For "Mobile Calendar," these four bands also have relatively stronger correlations before noise shaping, but this correlation is reduced after noise shaping. Figure 8 illustrates the energy distribution among the 28 subbands for the top scale with and without noise shaping. The energy distribution pattern depends on the edge and motion patterns in the underlying sequence. For example, the energy is more evenly distributed between different subbands with "Mobile Calendar." Further more, noise shaping helps to concentrate the energy into fewer subbands.

5. DDWT-BASED VIDEO CODING

In this section, we present two codecs: DDWT-SPIHT and DDWTVC. Both codecs do not perform motion estimation. Rather, the 3D DDWT is first applied to the original video directly and the noise-shaping method is then used to deduce the significant coefficients. The two codecs differ in their ways to code the retained DDWT coefficients. The DDWT-SPIHT codec directly applies the well-known 3D SPIHT codec on each of the four DDWT trees. Hence it does not exploit the correlation crosssubbands in different trees. The second codec, DDWTVC, exploits the intersubband correlation in the significance maps, but code the sign and magnitude information within each subband separately.

5.1. DDWT-SPIHT codec

Recall that the DDWT coefficients are arranged in four subband trees, each with a similar structure as the standard DWT. So it is interesting to find out how an existing DWT-based codec works on each of the four 3D DDWT trees. The 3D SPIHT [13] is a well-known wavelet codec, which utilizes the 3D DWT property that an insignificant parent does not

have significant descendants with high probability (parent-children probability). To examine such correlation across different scales in 3D DDWT, we evaluated the parent-children probability shown in Figure 9. We can see that there is strong correlation across scales with 3D DDWT, but compared to 3D DWT, the correlation is weeker. After noise shaping, this correlation is further reduced.

Based on the similar structure and properties with DWT, our first DDWT-based video codec applies 3D SPIHT [13] to each DDWT tree after noise shaping. As will be seen in the simulation results presented in Section 5.3, this simple method, not optimized for DDWT statistics, already outperforms the 3D SPIHT codec based on the DWT. This shows that DDWT has the potential to significantly outperform DWT for video coding.

5.2. DDWTVC codec

In DDWTVC, the noise-shaping method is applied to determine the 3D DDWT coefficients to be retained, and then a bitplane coder is applied to code the retained coefficients. The low subbands and high subbands are coded separately, each with three parts: significance-map coding, sign coding, and magnitude refinement.

5.2.1. Coding of significance map

As has been shown in Section 4.1, there are significant correlations between the significance maps across 28 high subbands, and the entropy of the significance vector is much smaller than 28. This low entropy prompted us to apply adaptive arithmetic coding for the significance vector. To utilize the different statistics of the high subbands in each bitplane, individual adaptive arithmetic codec is applied for each bitplane separately. Though the vector dimension is 28, for each bitplane, only a few patterns appear with high probabilities. So only patterns appearing with sufficiently high probabilities (determined based on training sequences) are coded using vector arithmetic coding. Other patterns are coded with an escape code followed by the actual binary pattern.

For the four low subbands, vector coding is used to exploit the correlation among the spatial neighbors (2×2 regions) and four low subbands. The vector dimension in the first bitplane is 16. If a coefficient is already significant in a previous bitplane, the corresponding component of the vector is deleted in the current bitplane. After the first several bitplanes, the largest dimension is reduced to below 10. As with the high subbands, only symbols occurring with a sufficiently high probability are coded using arithmetic coding. Different bitplanes are coded using separate arithmetic coders and different vector sizes.

The proposed video coder codes 3D DDWT coefficients in each scale separately. As illustrated in Figure 9, 3D DDWT does not have strict parent-children relationship as does the 3D DWT [13]. Noise shaping destroys such a relationship further. So the spatial-temporal orientation trees used in 3D SPIHT [13] are only applied in the finest stage, which has a lot of zero coefficients.

5.2.2. Coding of sign information

This part is used to code the sign of the significant coefficients. Our experiments show that four low subbands have very predicable signs. This predictability is due to the particular way the 3D DDWT coefficients are generated. Recall the orthonormal combination matrix for producing the 3D DDWT given in (2). Because the original DWT low-subbands are always positive (because they are lowpass filtered values of the original image pixels) and the coefficients in different low subbands have similar values at the same location, based on the combination matrix, the low subband in the first DDWT tree is almost always negative, and the other three low subbands in the other three DDWT trees are almost all positive. We predict the signs of significant coefficients in low subbands according to the above observation, and code the prediction errors using arithmetic coding.

For high subbands, we have found that the current coefficient tends to have the same sign as its neighbor in the lowpass direction, but have the opposite sign to its highpass neighbor. (In a subband which is horizontally lowpass and vertically high-pass, the lowpass neighbors are those to the left and right, and highpass neighbors are those above and below.) The prediction from the lowpass neighbor is more accurate than that from the highpass neighbor. The coded binary valued symbol is the product of the predicted and real sign bit. To exploit the statistical dependencies among adjacent coefficients in the same subband, we apply the similar sign context models of 3D embedded wavelet video (EWV) [21].

5.2.3. Magnitude refinement

This part is used to code the magnitudes (0 or 1) of significant coefficients in the current bitplane. Because only a few subbands have strong correlation as demonstrated in Section 4.2, the magnitude refinement is done in each subband individually. The context modelling is used to explore the dependence among the neighboring coefficients. Context models similar to the EWV method [21] are applied to 3D DDWT here.

5.3. Experimental results of DDWT video coding

In this section, we evaluate the coding performance of the two proposed codecs, DDWT-SPIHT and DDWTVC. The comparisons are made to 3D SPIHT [13] using DWT (to be referred as DWT-SPIHT). None of these codecs use motion compensation. Only the comparisons of luminance component Y are presented here. Two CIF sequences "Stefan" and "Mobile Calendar" and a QCIF sequence "Foreman" are used for testing. All sequences have 80 frames with a frame rate of 30 fps. Figure 10 compares the RD performances of DWT-SPIHT and the two proposed video codecs, DDWTVC and DDWT-SPIHT.

Figure 10 illustrates that both DDWT-SPIHT and DDWT-VC outperform DWT-SPIHT for all video sequences. Compared to DDWT-SPIHT, DDWTVC gives comparable or better performance for tested sequences. For a

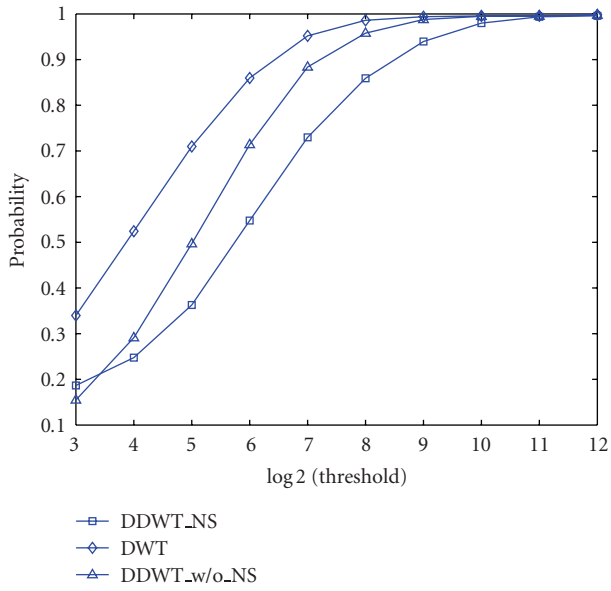


Figure 9: Probability that an insignificant parent does not have significant descendants for "Forman."

video sequence which has many edges and motions, like "Mobile Calendar," DDWTVC outperforms DWT-SPIHT more than 1.5 dB. DDW-TVC improves up to 0.8 dB for the "Foreman" and 0.5 dB better PSNR for "Stefan." Considering that DDWT has four times raw data than 3D DWT, these results are very promising.

Subjectively, both DDWTVC and DDWT-SPIHT have better quality than DWT-SPIHT for all tested sequences. Coded frames by DDWTVC and DWT-SPIHT for a frame from the "Stefan" sequence are shown in Figure 11. We can see that DDWTVC preserves edge and motion information better than DWT-SPIHT; DWT-SPIHT exhibits blurs in some regions and when there are a lot of motions. The visual differences here are consistent with those in other sequences.

When displayed as a video at real time (30 fps), the DWT-SPIHT coded video was found to exhibit annoying flickering artifacts. To investigate the reason behind this, we show the $x - t$ frames of decoded video, where an $x - t$ frame is a horizontal cut of the video through time (as illustrated in Figure 12). Figure 13 (a) illustrates the original $x - t$ frame, and (b) is the decoded $x - t$ frame from DDWTVC, and (c) is the DWT-SPIHT $x - t$ frame. Figure 13 illustrates that the motion trajectory in the original and DDWTVC decoded $x - t$ frames is much more smoother, but the DWT-SPIHT $x - t$ frame has more zigzag characteristics, which might be the reason why the DWT-SPIHT coded video has flickering artifacts.

Recall that the DDWT-SPIHT codec exploits the spatial and temporal correlation within each subband while coding significance, sign, and magnitude information. The DDWTVC codec also exploits within subband correlation when coding the sign and magnitude information. But for the significance information, it exploits the intersubband correlation, at the expense of the intrasubband correlation.

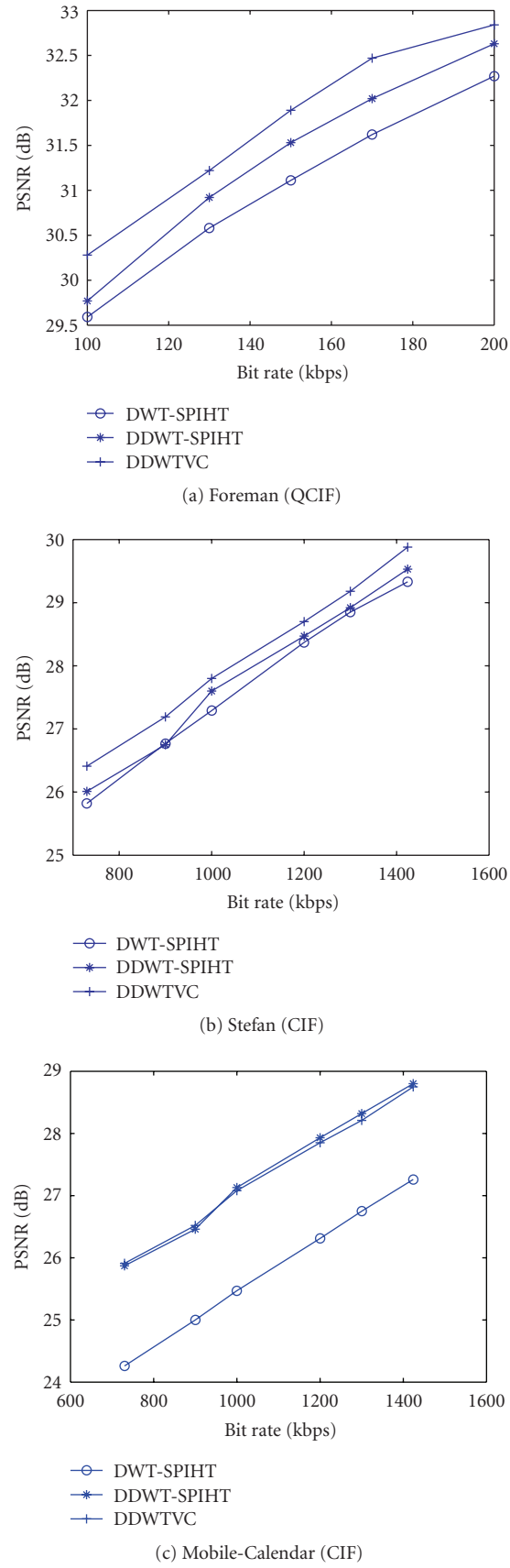


Figure 10: The R-D performance comparison of DDWT-SPIHT, DDWTVC, and DWT-SPIHT.

(a) The 16th frame in "Stefan" reconstructed from DDWTVC

(b) The 16th frame in "Stefan" reconstructed from DWT-SPIHT

Figure 11: The subjective performance comparison of DDWTVC and DWT-SPIHT for "Stefan."

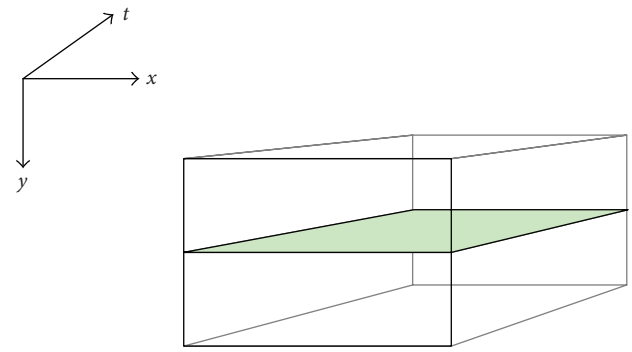


Figure 12: The illustration of $x - t$ frame: horizontal contents (x) along the temporal direction (t).

Our simulation results suggest that the exploiting interband correlation is equally, if not more, important as exploiting the intraband correlation. The benefit from exploiting the interband correlation is sequence dependent. A codec that can exploit both interband and intraband correlations is expected to yield further improvement. This is a topic of our future research.

6. SCALABILITY OF DDWTVC

Scalable coding refers to the generation of a scalable (or embedded) bit stream, which can be truncated at any point to yield a lower-quality representation of the signal. Such rate scalability is especially desirable for video streaming applications, in which many clients may access the server through access links with vastly different bandwidths.

The main challenge in designing scalable coders is how to achieve scalability without sacrificing the coding efficiency. Ideally, we would like to achieve rate-distortion (R-D) optimized scalable coding, that is, at any rate R, the truncated stream yields the minimal possible distortion for that R.

One primary motivation for using 3D wavelets for video coding is that wavelet representations lend themselves to both spatial and temporal scalability, obtainable by ordering the wavelet coefficients from coarse to fine scales in both space and time. It is also easy to achieve quality scalability by representing the wavelet coefficients in bitplanes and coding the bitplanes in order of significance. Because the 3D DWT is an orthogonal transform, the R-D optimality is easier to approach by simply coding the largest coefficients first.

To generate an R-D-optimized scalable bit stream using an overcomplete transform like 3D DDWT, it will be necessary to generate a scalable set of coefficients so that each additional coefficient offers a maximum reduction in distortion without modifying the previous coefficients. However, with the iterative noise-shaping algorithm, the selected coefficients do not enjoy this desired property, because the noise-shaping algorithm modifies previously chosen large coefficients to compensate for the loss of small coefficients. With the coefficients derived from a chosen threshold, the DDWTVC produces a fully scalable bit stream, offering spatial, temporal, and quality scalability over a large range. But the R-D performance is optimal only for the highest bit rate associated with this threshold.

Results in Figure 10 are obtained by choosing the best noise-shaping threshold among a chosen set, for each target bit rate. Specifically, the candidate thresholds are 128, 64, 32 for different bit rates, respectively. Our experiments demonstrate that at low bit rate (less than 1 Mbps for CIF), the coefficients set retained by noise shaping threshold 128 offers best results, and threshold 64 works best when the bit rate is between 1 and 2 Mbps. If the bit rate is above 2 Mbps, the codec uses coefficients obtained by threshold 32.

Figure 14 illustrates the reconstruction quality (in terms of PSNR) at different bit rates for different final noise-shaping thresholds. In this simulation, the encoded bitstreams, which are obtained by choosing different final noise-shaping thresholds, are truncated at different decoding bit rates. The truncation is such that the decoded sequence

(a) The original $x - t$ frame in "Mobile Calendar"

(b) The DDWTVC $x - t$ frame in "Mobile Calendar"

(c) The DWT-SPIHT $x - t$ frame in "Mobile Calendar"

FIGURE 13: The subjective comparison of the $x-t$ frames in "Mobile Calendar."

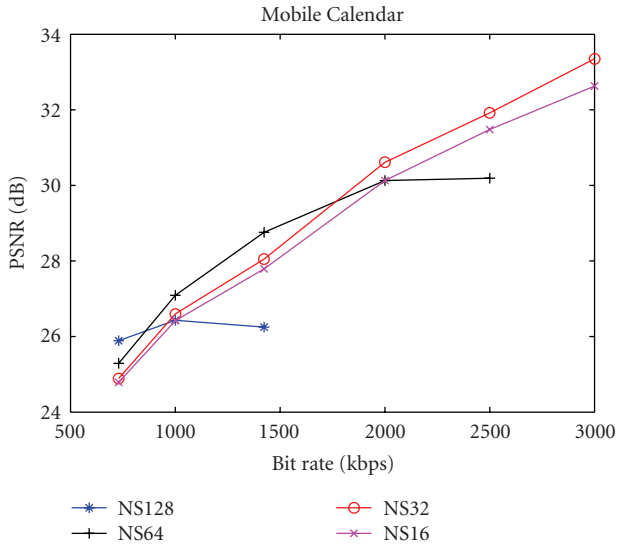


FIGURE 14: Comparison of the reconstruction quality (in terms of PSNR) at different bit rates for different final noise-shaping thresholds.

has the same temporal and spatial resolution but different PSNR (SNR scalability). As expected, each noise-shaping final threshold is optimal only for a limited bit rates range. For example, the final threshold 32 gives highest PSNR between 2500–3000 kbps, and threshold 64 outperforms other thresholds from 1000 kbps to 2000 kbps. If we choose one low final threshold, for example, threshold 32 (the *o* curve), the maximum degradation from best achievable quality at different rate is about 1 dB or so. Considering that it is fully scalable, the 1 dB is a coding efficiency penalty for full scalability. Note that the coding results for all the thresholds are obtained by using the statistics collected for coefficients obtained with the threshold of 64. Had we used the statistics collected for the actual threshold used, the performance for thresholds other than 64 would have been better.

7. EXTENSIONS OF THE 3D DDWT

7.1. The 3D anisotropic dual-tree wavelet transform

In the previous codec designs, the 3D DDWT utilizes an isotropic decomposition structure in the same way as the conventional 3D DWT. It is worth pointing out that not only the low frequency subband LLL, but also subbands LLH, HLL, LHL, and so forth, include important low frequency information. In addition, more spatial decomposition stages normally produce noticeable gain for video processing. On the other hand, less-temporal decomposition stages can save memory and processing delay. Based on these observations, we propose a new anisotropic wavelet transform for 3D DDWT. The proposed anisotropic DDWT extends the superiority of normal isotropic DDWT with more directional subbands without adding to the redundancy.

The anisotropic DDWT we introduce here follows a particular rule in dividing the frequency space: when a subband is in the low-frequency end in any one direction, it will be further divided in this direction, until no more decomposition can be done.

In the DDWT, different frequency tilings lead to different orientation of wavelets [10]. Figure 15 illustrates the orientation of three subbands of isotropic 2D DDWT. The wavelets that have the subband indexed as 1, 2, and 3 have orientations of approximately -45, -75, and -15 degrees as shown in Figure 15 (for clarity, only 1 decomposition level is shown below).

Figure 16 demonstrates the 2D frequency tiling of the isotropic and anisotropic wavelet transforms, respectively, for 2 levels of decomposition in each direction. In Figure 16(b), the original LH and HL subbands are further divided into two corresponding rectangular subbands. In both Figures 16(a) and 16(b), wavelets that have the subbands indexed as 1, 2, and 3 have orientation of approximately -45, -75, and -15 degrees. But in Figure 16(b), anisotropic wavelets corresponding to subbands 4 to 7 have some additional orientations of -81, -63, -9, -27 degrees.

In 3D, the number of subbands and orientations increases more dramatically. With the original 3D DDWT, the frequency space is always partitioned as cubes. For each level of decomposition, the LLL subband is further divided into eight cubes. The number of subbands increases by 7. For an N level decomposition, the total number of subbands is $7N + 1$. On the other hand, for the anisotropic DDWT, the frequency space is partitioned into cuboids. The subbands of DDWT are further divided. The total number of subbands is

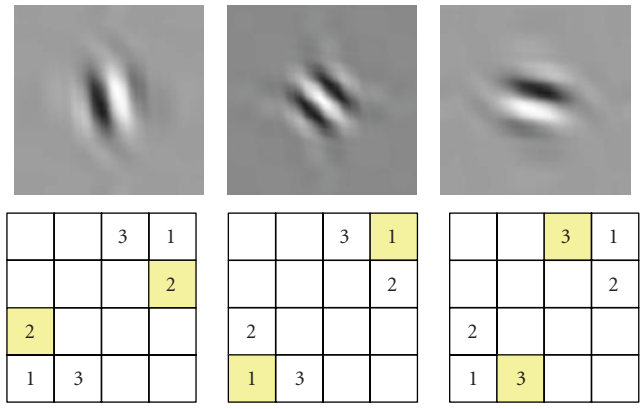


Figure 15: Typical wavelets associated with the isotropic 2D DDWT. The top row illustrates the wavelets in the spatial domain, the second row illustrates the (idealized) support of the spectrum of each wavelet in the 2D frequency plane.

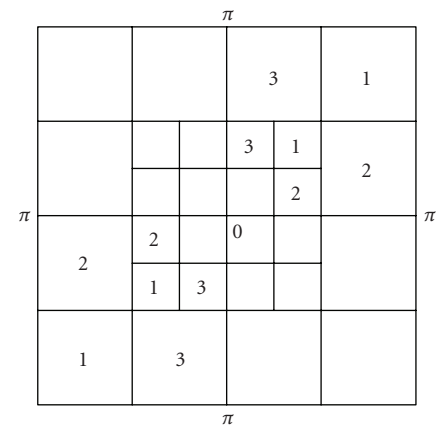
$(N_r + 1)(N_c + 1)(N_t + 1)$, where N_r, N_c, and N_t are decomposition levels for row, column, and temporal direction, respectively. Usually we use the same number of decomposition levels for the two special directions and allow different levels for the temporal direction. The additional subbands with different orientations add to the flexibility of the original isotropic DDWT. Asides from the different decompositions within a single tree, all other implementations are just the same as DDWT [10], and the redundancy is not increased.

Figure 17 demonstrates the structure of 3D isotropic and proposed anisotropic transforms in spacial and temporal domains. Both structures applied two wavelet decomposition levels. In the isotropic structure, only the low subband LLL is decomposed each time. But the anisotropic structure decomposes all subbands except the highest-frequency subband HHH into new subbands. We applied noise shaping on the new anisotropic structure of 3D DDWT, and compared it to the original isotropic 3D DDWT and the standard 3D DWT. In this experiment, three wavelet decomposition levels are applied in each direction for both video sequences. The 3D DDWT and DWT filters are the same as in Section 2.

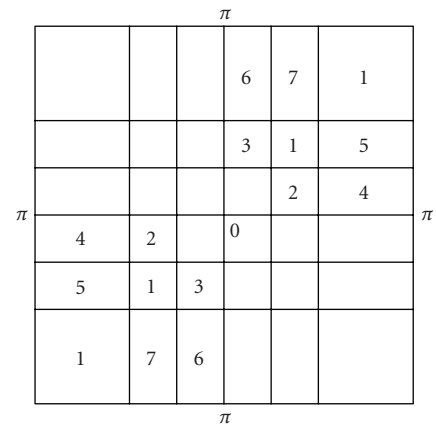
Figure 18 shows that the 3D DDWT, both isotropic and anisotropic structures, achieve better quality (in terms of PSNR) than the standard 3D DWT, with the same number of retained coefficients. To achieve the same PSNR, the anisotropic structure needs about 20% fewer coefficients on average than the isotropic structure. With the same number of retained coefficients, *DDWT_NS* yields higher PSNR than DWT. The anisotropic structure (*DDWT_anisotropic_NS*) outperforms the isotropic structure (*DDWT_isotropic_NS*) by 1-2 dB.

7.2. Combining 3D DDWT and DWT

The 3D DDWT isolates different spatial orientations and motion directions in each subband, which is desirable for video representations. In terms of 2D orientation, the DDWT is oriented along six directions: ±75°, ±45°, and



(a) Isotropic tiling



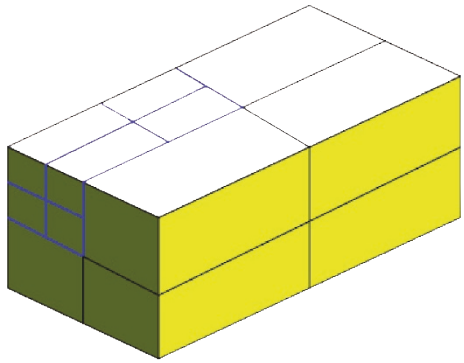
(b) Anisotropic tiling

Figure 16: 2D anisotropic dual-tree wavelets for 2 levels of decomposition in each direction (from one tree).

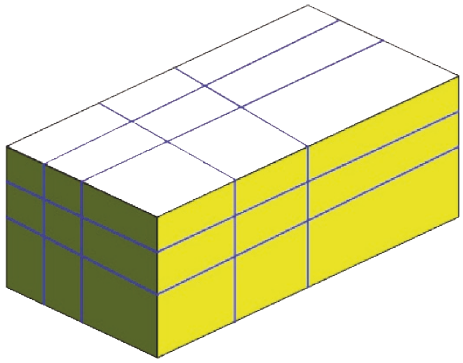
±15°. Unfortunately, the DDWT does not represent the vertical and horizontal orientations (±0° and ±90°) in pursuit of other directions. Recognizing this deficiency, we propose to combine the 3D DDWT and DWT, to capture directions represented by both.

Considering that the horizontal and vertical orientations are usually dominant in natural video sequences, we gave the DWT priority to represent the video sequences. We assume that the horizontal and vertical features in the video sequences will be represented by large DWT coefficients. So we apply DWT at first and only keep the coefficients over a certain threshold. Then we apply the DDWT on the residual video, and the noise shaping is used to select the significant DDWT coefficients of the residual video. This is illustrated in Figure 19. Recognizing that the DWT subband has a normalized energy that is four times that of the DDWT subband, we use 1000 as the threshold for chosing significant 3D DWT coefficients, and use 256 as the initial noise-shapping threshold for DDWT coefficients.

The simulation results of combining DWT and DDWT are shown in Figure 20. The isotropic structure is used in the simulations. Figure 20 illustrates that the combined DDWT and DWT achieve slightly better quality (in terms of PSNR)

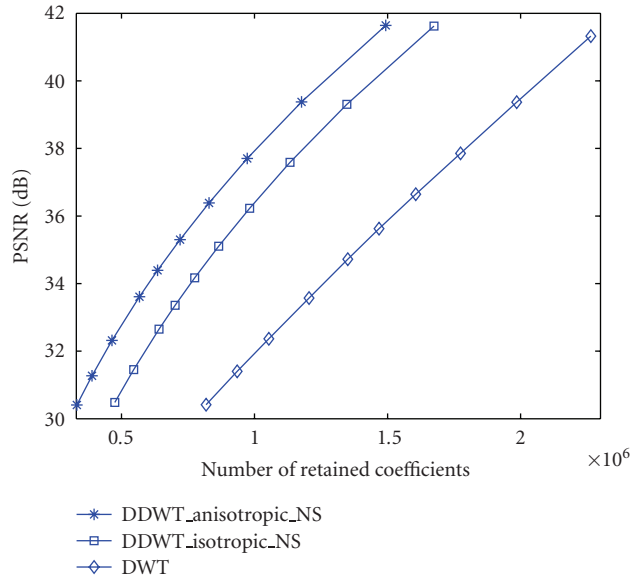


(a) Isotropic decomposition

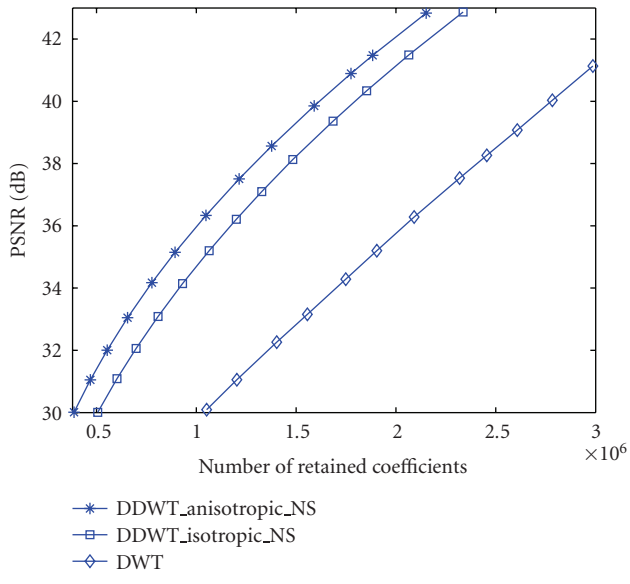


(b) Anisotropic decomposition

Figure 17: Comparison of isotropic decomposition and anisotropic decomposition.



(a) Stefan (CIF)



(b) Mobile-Calendar (CIF)

Figure 18: Comparison of the reconstruction quality (in terms of PSNR) using the same number of retained coefficients with the isotropic DDWT with noise shaping (DDWT_isotropic_NS, upper curve) and the anisotropic DDWT with noise shaping (DDWT_anisotropic_NS, middle curve) and DWT (lower curve).

than the 3D DDWT alone, with the same number of retained coefficients. To achieve the same PSNR, the combined transform needs up to 8% fewer coefficients than the 3D DDWT alone.

8. CONCLUSION

We demonstrated that the 3D DDWT has attractive properties for video representation. Although the 3D DDWT is an overcomplete transform, the raw number of coefficients can be reduced substantially by applying noise shaping. The fact that noise shaping can reduce the number of coefficients to below that required by the DWT (for the same video quality) is very encouraging. The vector entropy study validates our hypothesis that only a few basis functions have significant energy for an object feature. The relatively low vector entropy suggests that the whereabouts of significant coefficients may be coded efficiently by applying vector arithmetic coding to the significance bits across subbands. The fact that coefficient values do not have strong correlation among the subbands, on the other hand, indicates that the benefit from vector coding the magnitude bits across the subbands may be limited.

Based on our investigation, two new video codecs, namely, DDWT-SPIHT and DDWTVC, using the 3D dual-tree wavelet transform are proposed and tested on standard video sequences. The DDWT-SPIHT applies 3D SPIHT on each DDWT tree to exploit the correlation within each subband. The 3D DDWT video codec (DDWTVC) applies adaptive vector arithmetic coding across subbands to efficiently code the significance bits jointly. This vector coding successively exploits the cross-band correlation in significance bits. But the spatial dependence of significance bits in each

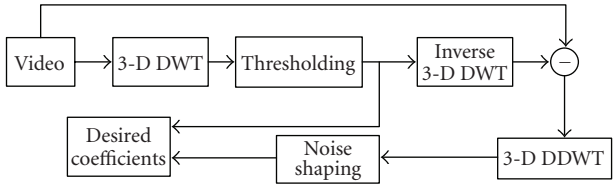
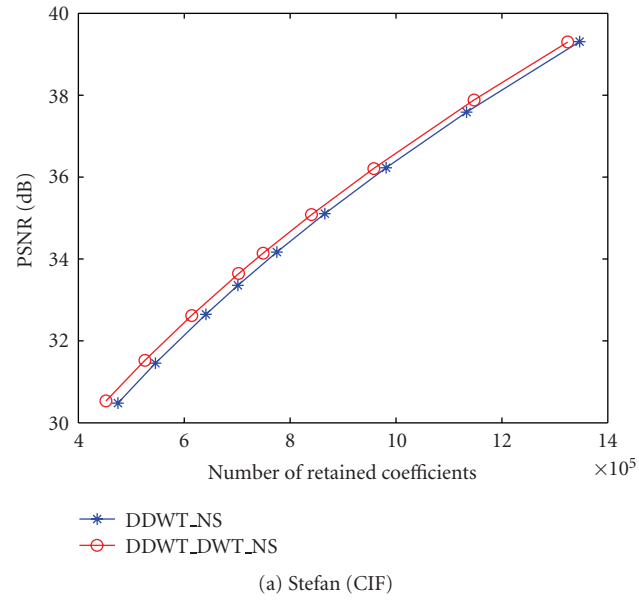


Figure 19: The structure of combining 3D DDWT and DWT.

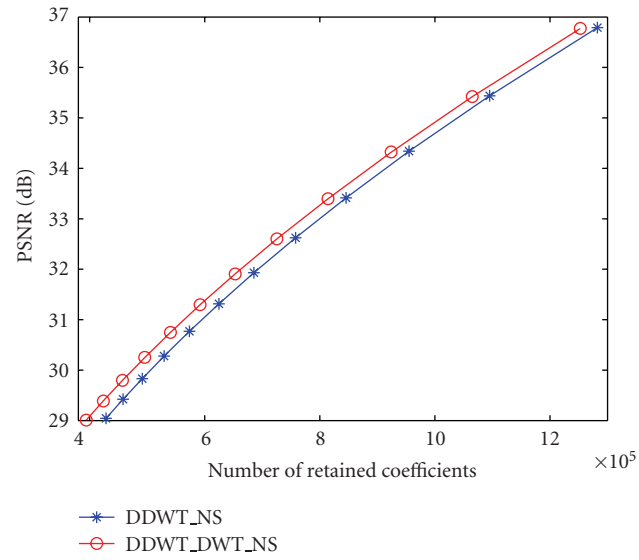
subband has not been explored in the current DDWTVC. Recognizing that context-based coding is an effective mean to explore such dependence, we have explored the use of context-based arithmetic vector coding. A main difficulty in applying context models is that the complexity grows exponentially with the number of pixels included in the context. We have tested the efficiency of various contexts, which differ in the chosen subbands and spatial neighbors. However the study so far has not yielded significant gain over direct vector coding.

Besides the standard isotropic decomposition structure, a new anisotropic structure of the novel 3D dual-tree wavelet transform is also proposed and tested on video coding. The anisotropic structure of the 3D DDWT decomposes not only the lowest subband LLL, but also all other subbands except the highest subband HHH. The number of the decomposition stages can be different along temporal, horizontal, and vertical directions. This structure is more effective than the traditional isotropic structure. The anisotropic structure can yield better reconstruction quality (in terms of PSNR) for the same number of coefficients. We also propose to combine 3D DWT and DDWT to capture more directions and edges in video sequences. The combined structure, however, leads to only slight gains in terms of reconstruction quality versus number of coefficients.

In terms of future work, more properties of the 3D dual-tree transform need to be exploited. First of all, a codec that can exploit both interbands and intraband correlation in coding the significance bits is expected to provide significant improvement. Secondly, how to incorporate the proposed anisotropic DDWT in video coding is still open, because the number of subbands and orientations increases more dramatically in anisotropic structure. Based on the gain in terms of the reconstruction quality versus number of (unquantized) coefficients, we expect that a codec using the anisotropic DDWT can lead to additional significant gains. The codec in [22] exploits both inter- and intraband correlations. It also compares the performance obtainable with isotropic and anisotropic decomposition. With isotropic DDWT, their codec has, however, similar performance as DDWTVC. The anisotropic DDWT achieved on average a gain of 1 dB over isotropic. Finally, with noise shaping, the optimal set of coefficients to be retained changes with the target bit rate. To design a scalable video coder, we would like to have a scalable set of coefficients so that each additional coefficient offers a maximum reduction in distortion without modifying the previous coefficients. How to deduce such coefficient sets is a challenging open research problem.



(a) Stefan (CIF)



(b) Mobile-Calendar (CIF)

Figure 20: Comparison of the reconstruction quality (in terms of PSNR) using the same number of retained coefficients with DDWT (DDWT_NS, upper curve) and the combined DDWT and DWT(DDWT_DWT_NS, lower curve). The DDWT coefficients are obtained by noise shaping in both cases.

ACKNOWLEDGMENTS

This work was supported in part by the National Science Foundation under Grant no. CCF-0431051 and is partially supported by the Joint Research Fund for Overseas Chinese Young Scholars of NSFC under Grant no. 60528004. Parts of this work have been presented at International Conference on Acoustics, Speech, and Signal Processing (ICASSP) 2005, and Picture Coding Symposium (PCS) 2006.

REFERENCES

[1] S.-T. Hsiang and J. W. Woods, "Embedded video coding using invertible motion compensated 3-D subband/wavelet filter bank," *Signal Processing: Image Communication*, vol. 16, no. 8, pp. 705–724, 2001.

[2] J. Xu, Z. Xiong, S. Li, and Y.-Q. Zhang, "Memory-constrained 3-D wavelet transform for video coding without boundary effects," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 12, no. 9, pp. 812–818, 2002.

[3] Y. Andreopoulos, M. van der Schaar, A. Munteanu, J. Barbarien, P. Schelkens, and J. Cornelis, "Fully-scalable wavelet video coding using in-band motion compensated temporal filtering," in *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '03)*, vol. 3, pp. 417–420, Hong Kong, April 2003.

[4] A. Secker and D. Taubman, "Lifting-based invertible motion adaptive transform (LIMAT) framework for highly scalable video compression," *IEEE Transactions on Image Processing*, vol. 12, no. 12, pp. 1530–1542, 2003.

[5] "Joint Scalable Video Model 2.0 Reference Encoding Algorithm Description," ISO/IEC JTC1/SC29/WG11/N7084. Buzan, Korea, April 2005.

[6] N. Kingsbury, "A dual-tree complex wavelet transform with improved orthogonality and symmetry properties," in *Proceedings of IEEE International Conference on Image Processing (ICIP '00)*, vol. 2, pp. 375–378, Vancouver, BC, Canada, September 2000.

[7] M. N. Do and M. Vetterli, "The contourlet transform: an efficient directional multiresolution image representation," *IEEE Transactions on Image Processing*, vol. 14, no. 12, pp. 2091–2106, 2005.

[8] T. H. Reeves and N. G. Kingsbury, "Overcomplete image coding using iterative projection-based noise shaping," in *Proceedings of IEEE International Conference on Image Processing (ICIP '02)*, vol. 3, pp. 597–600, Rochester, NY, USA, September 2002.

[9] K. Sivaramakrishnan and T. Nguyen, "A uniform transform domain video codec based on dual tree complex wavelet transform," in *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP '01)*, vol. 3, pp. 1821–1824, Salt Lake City, Utah, USA, May 2001.

[10] I. Selesnick and K. Y. Li, "Video denoising using 2D and 3D dual-tree complex wavelet transforms," in *Wavelets: Applications in Signal and Image Processing X*, vol. 5207 of *Proceedings of SPIE*, pp. 607–618, San Diego, Calif, USA, August 2003.

[11] I. Selesnick, R. G. Baraniuk, and N. C. Kingsbury, "The dual-tree complex wavelet transform," *IEEE Signal Processing Magazine*, vol. 22, no. 6, pp. 123–151, 2005.

[12] B. Wang, Y. Wang, I. Selesnick, and A. Vetro, "An investigation of 3D dual-tree wavelet transform for video coding," in *Proceedings of International Conference on Image Processing (ICIP '04)*, vol. 2, pp. 1317–1320, Singapore, October 2004.

[13] B.-J. Kim, Z. Xiong, and W. A. Pearlman, "Low bit-rate scalable video coding with 3-D set partitioning in hierarchical trees (3-D SPIHT)," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 10, no. 8, pp. 1374–1387, 2000.

[14] B. Wang, Y. Wang, I. Selesnick, and A. Vetro, "Video coding using 3-D dual-tree discrete wavelet transforms," in *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP '05)*, vol. 2, pp. 61–64, Philadelphia, Pa, USA, March 2005.

[15] D. Xu and M. N. Do, "Anisotropic 2D wavelet packets and rectangular tiling: theory and algorithms," in *Wavelets: Applications in Signal and Image Processing X*, vol. 5207 of *Proceedings of SPIE*, pp. 619–630, San Diego, Calif, USA, August 2003.

[16] D. Xu and M. N. Do, "On the number of rectangular tilings," *IEEE Transactions on Image Processing*, vol. 15, no. 10, pp. 3225–3230, 2006.

[17] S. G. Mallat and Z. Zhang, "Matching pursuits with time-frequency dictionaries," *IEEE Transactions on Signal Processing*, vol. 41, no. 12, pp. 3397–3415, 1993.

[18] R. Gribonval and P. Vandergheynst, "On the exponential convergence of matching pursuits in quasi-incoherent dictionaries," *IEEE Transactions on Information Theory*, vol. 52, no. 1, pp. 255–261, 2006.

[19] R. Neff and A. Zakhor, "Very low bit-rate video coding based on matching pursuits," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 7, no. 1, pp. 158–171, 1997.

[20] H. Bolcskei and F. Hlawatsch, "Oversampled filter banks: optimal noise shaping, design freedom, and noise analysis," in *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP '97)*, vol. 3, pp. 2453–2456, Munich, Germany, April 1997.

[21] J. Hua, Z. Xiong, and X. Wu, "High-performance 3-D embedded wavelet video (EWV) coding," in *Proceedings of 4th IEEE Workshop on Multimedia Signal Processing (MMSP '01)*, pp. 569–574, Cannes, France, October 2001.

[22] J. B. Boettcher and J. E. Fowler, "Video coding using a complex wavelet transform and set partitioning," to appear in *IEEE Signal Processing Letters,* September 2007.