

# Video Coding Using a Complex Wavelet Transform and Set Partitioning

Joseph B. Boettcher, *Student Member, IEEE*, and James E. Fowler, *Senior Member, IEEE*

**Abstract**—A video-coding system that exploits the motion-selective characteristics of the 3D complex dual-tree discrete wavelet transform is presented. The proposed system does not perform explicit motion compensation but instead relies on the dual-tree transform to isolate moving features. Although the dual-tree transform is redundant, a noise-shaping process increases the sparsity of the transform coefficients, resulting in a high degree of spatiotemporally coherent regions of insignificant coefficients. The transform coefficients are coded with binary set-partitioning using  $k$ -d trees in an algorithm that exploits both within-subband spatiotemporal coherency as well as cross-subband correlation to achieve efficient coding. Experimental results demonstrate that the proposed system outperforms other coders that also do not perform explicit motion estimation or compensation.

**Index Terms**—dual-tree wavelet transform, set partitioning, video coding

## I. INTRODUCTION

Traditionally, video coders have employed a block-based motion-compensation feedback loop to exploit the temporal redundancy in a video signal. Although this approach has been effective, it has several drawbacks. The feedback loop impedes scalability, while block-based motion compensation can produce visual artifacts in the reconstructed frames at block boundaries. Furthermore, motion-estimation procedures are usually the heaviest computational burden on the encoder. Recent systems have avoided the motion-compensation feedback loop by applying a discrete wavelet transform (DWT) in the temporal direction. However, for the resulting temporal subbands to be of beneficial quality, motion estimation is still necessary in these systems to guide the temporal transform in the direction of predicted motion; i.e., motion-compensated temporal filtering (MCTF).

An alternative to MCTF has arisen recently in the form of the complex dual-tree discrete wavelet transform (DDWT) [1–3]. The DDWT is a redundant transform that, in the 3D case [3], produces four times as many subbands as the DWT, with each subband oriented in a different spatiotemporal direction. When applied to a video signal, these orientations help isolate image features moving in different directions, providing inherent motion selectivity. The ability of the transform to describe motion without explicit motion estimation or compensation has motivated the use of the DDWT in video-coding systems [4–6] looking to avoid the computational complexity associated with motion estimation. However, since

the 3D DDWT is four times redundant, efficient coding of the transform coefficients is a challenging task.

The coder proposed in [5, 6]—the DDWT video coder (DDWTVC)—exploits the fact that, although the DDWT is greatly redundant, there is a significant degree of correlation between coefficients residing at the same spatiotemporal locations in different subbands. The DDWTVC uses arithmetic coding of cross-subband vectors of coefficients to exploit this cross-subband correlation. However, large-magnitude DDWT coefficients typically occur rather sparsely in any given DDWT subband, with most of the coefficients being small or zero (i.e., insignificant coefficients). In fact, it has proven beneficial to apply a “noise-shaping” procedure [2] to deliberately increase the sparsity of the transform coefficients [4, 6]. Yet, DDWTVC does not explicitly exploit the fact that the insignificant coefficients tend to form spatiotemporally coherent regions within each subband.

In this paper, we propose a video-coding system using the DDWT in conjunction with an embedded wavelet-based coder called binary set-splitting with  $k$ -d trees (BISK) [7–9]. Our DDWT-BISK algorithm uses set partitioning to exploit not only cross-subband correlation but also spatiotemporal coherency within subbands to effectively represent the sparse coefficient volume. In the following, we first overview complex wavelet transforms and their prior use in video coding before describing the DDWT-BISK coder in detail. We then present experimental results that demonstrate that the proposed DDWT-BISK coder outperforms other coders, including DDWTVC, that do not perform explicit motion estimation or compensation.

## II. VIDEO CODING WITH COMPLEX WAVELETS

### A. Complex Wavelet Transforms

In order to overcome some of the shortcomings of the traditional critically-sampled DWT, Kingsbury [1] introduced the DDWT consisting of two trees of real wavelet filters operating on the same data in parallel, with the filters designed such that the two trees produce the real and imaginary parts of the complex-valued coefficients. While the DWT lacks shift invariance, the DDWT is approximately shift invariant and offers higher directional selectivity. However,  $2^m$ :1 redundancy is added for an  $m$ -dimensional signal.

Selesnick and Li [3] developed a 3D version of the DDWT to provide a useful representation for video. It turns out that the degree of redundancy can be reduced without sacrificing perfect reconstruction by simply discarding the complex parts of the coefficients, resulting in 4:1 redundancy. For this real-valued transform, four separable 3D DWTs based on Hilbert

The authors with the Department of Electrical & Computer Engineering and the GeoResources Institute, Mississippi State University, Starkville, MS.

This work was funded in part by the US National Science Foundation under Grant No. CCR-0310864.

pairs are applied to the original signal, and only the real parts of the coefficients are retained. The four sets of transform data are then combined with linear operations to produce subbands that isolate features in a variety of orientations. The resulting DDWT subbands are arranged in four separate *transform combinations* with each combination having the same subband organization as would a 3D DWT of the original data but with each combination containing subbands of different orientation. For example, a 3D dyadic DWT consists of 7 highpass subbands at each resolution level, plus a single baseband at the lowest resolution. Consequently, at each resolution level, the corresponding 3D DDWT (illustrated in Fig. 1) consists of  $4 \times 7 = 28$  highpass subbands, except at the lowest resolution, which contains 4 baseband subbands.

Although the 3D DDWT produces four times the data that the 3D DWT does, the DDWT requires fewer critical coefficients to efficiently represent the underlying signal [4]. To wit, Reeves and Kingsbury [2] proposed deliberately reducing the number of DDWT coefficients by discarding small magnitude coefficients and refining the remaining coefficients to compensate. This “noise-shaping” procedure is an iterative projection of signals between the original-signal domain and the DDWT domain. On each iteration, the signal is thresholded in the DDWT domain to remove small coefficients, and the remaining coefficients are compensated by the original-signal-domain error induced by the thresholding. This noise-shaping procedure increases the sparsity of the representation to the point that the 3D DDWT typically requires fewer non-zero coefficients than the 3D DWT to achieve the same level of reconstruction quality for a video signal [4]. The DDWTVC coder [5, 6] extends this observation to actual coding results that include quantization and entropy coding.

### B. DDWTVC

Because the 3D DDWT provides inherent motion-selectivity, Wang *et al.* [5, 6] developed the DDWTVC coder to avoid explicit motion estimation and compensation. DDWTVC is a bitplane coder that exploits cross-subband redundancy in the highpass bands of the 3D DDWT coefficients. While the 3D dyadic DWT results in seven highpass subbands per level of decomposition, the corresponding 3D DDWT produces 28 highpass bands per level, due to the 4:1 redundancy of the transform. After the noise shaping of [2] is applied to the original video sequence, the significance states of the co-located coefficients in each of the 28 highpass bands are coded as a 28-bit vector using adaptive arithmetic coding in each bitplane separately. For the four lowpass bands, a 16-bit significance vector comprising  $2 \times 2$  blocks of coefficients co-located in each of the bands is coded at the first bitplane; however, with each successive bitplane, previously significant coefficients are removed from the vector so that the dimensionality decreases. The sign information for the coefficients is predicted, and the prediction error is coded. Arithmetic coding of both the sign-prediction error and the magnitude-refinement information is performed with context models in each subband individually. Wang *et al.* showed that the DDWTVC system exhibits rate-distortion performance superior to that of 3D-

SPIHT [10] applied directly to the video sequence with no motion estimation or compensation.

### III. DDWT-BISK

In the DDWTVC coder, correlation between transform coefficients is exploited *across* subbands in the form of 28-bit significance vectors; however, the spatiotemporal coherency of insignificant-coefficient regions *within* a given subband is not exploited despite the fact that this coherency must necessarily be substantial due to the sparsity ensured by the noise-shaping process. In order to efficiently code spatiotemporally coherent regions of DDWT coefficients, we propose a modified version of the BISK algorithm [7–9]. BISK performs bitplane coding in which significant coefficients are located by recursive spatiotemporal partitioning. Specifically, *k*-d trees are used to split sets of coefficients into two subsets of roughly equal size. Once a significant coefficient is located, its sign information is coded, and its magnitude is refined on successive passes. Significance, sign, and magnitude-refinement information are all coded with adaptive arithmetic coding.

In the proposed DDWT-BISK coder, the noise shaping of [2] is applied to produce sparse DDWT coefficients. Then, a modified version of the 3D-BISK algorithm [8, 9] operates on the transform coefficients to produce the final coded bitstream. First, coefficients are grouped into 4-dimensional vectors, where each vector consists of the four coefficients at the same spatiotemporal location in the same subband from each of the four DDWT transform combinations, as illustrated in Fig. 1. These coefficient vectors are then assembled into sets spanning the entire spatiotemporal subbands, producing 7 sets of vectors at each resolution level (4 sets of vectors at the baseband level), assuming a dyadic decomposition structure (other decompositions are discussed below). All sets are initially placed in the list of insignificant sets (LIS).

The algorithm then performs bitplane coding with *sorting* and *refinement* passes. In the sorting pass, sets in the LIS are tested against the current threshold to determine the significance of the set as a whole—if the magnitude of any coefficient in the set is above the threshold, the set is significant. Significance sets are split in two along the longest dimension of the set. The resulting subsets are added back to the LIS as two new sets to be recursively tested and split if necessary. Eventually, a significant set will be reduced to a single four-coefficient vector in which at least one of the four coefficients will be significant. At this point, the vector is removed from the LIS, and a significance symbol is output to denote which coefficients in the vector are significant and which are not. The significant coefficients from the vector are then added to the list of significant pixels (LSP), while the insignificant coefficients are added to the list of insignificant pixels (LIP). After each sorting pass, the LIP is processed by comparing each coefficient to the current threshold and outputting the significance state. If a coefficient in the LIP becomes significant, it is transferred to the LSP. The refinement pass then processes each coefficient in the LSP and outputs the current bitplane value of the coefficient magnitude. Sorting and refinement passes continue until the target bitstream length

has been reached. This set-partitioning procedure is illustrated in Fig. 2.

For the DDWT-BISK system, we consider two types of wavelet decomposition structures for the 3D DDWT as illustrated in Figs. 3(a) and (b). Fig. 3(a) gives a traditional dyadic decomposition wherein the wavelet transform is applied to only the lowpass band at each successive level of decomposition. Alternatively, we also consider a wavelet-packet decomposition, shown in Fig. 3(b) as the “anisotropic” structure, in which a full  $J$ -scale 1D wavelet transform is applied to each dimension of the 3D dataset separately. This anisotropic transform structure generates a greater number of subbands than does a dyadic structure for the same number of decomposition levels; in the context of the 3D DDWT, these additional subbands can provide additional directional orientations and can thus increase the degree of motion selectivity. We note that a dyadic decomposition was used in the original 3D DDWT development [3] and in the DDWTVC coder [5, 6]. Although the anisotropic DDWT was discussed in [6] wherein it was demonstrated that the anisotropic structure provided significantly better reconstruction quality after noise-shaping than the dyadic DDWT for the same number of retained coefficients, the actual DDWTVC system is tied to the dyadic DDWT due to its use of 28-bit vectors for arithmetic coding.

#### IV. RESULTS

In our experiments, we code the grayscale sequences shown in Table I with DDWT-BISK using both the dyadic and anisotropic transform structures. The iterative noise-shaping procedure employed is identical to that used in [6]—the initial and final thresholds driving the iterations are selected from a small set of possible values, with the selection optimized for each given bitrate. We compare our results against those provided in [6] for the DDWTVC coder. All coders apply three levels of wavelet decomposition in each dimension.

We also compare to JPEG2000 as a state-of-the-art coder using a traditional real-valued critically sampled DWT and no motion estimation or compensation. JPEG2000 results use extensions in Part 2 of the JPEG2000 standard to produce either the anisotropic decomposition of Fig. 3(b) or a wavelet-packet transform consisting of a 1D temporal transform followed by a 2D dyadic spatial transform such as illustrated in Fig. 3(c) (we call this latter decomposition the “packet” decomposition after terminology in [9, 10]).

Table I provides average PSNR results for all the test sequences at a fixed rate. The results show that DDWT-BISK with the dyadic transform is generally competitive with JPEG2000 using the packet transform. However, when the anisotropic transform structure is used, DDWT-BISK consistently shows substantial gains, while JPEG2000 shows little change in PSNR. Figs. 4 and 5 include the DDWTVC coder along with DDWT-BISK and JPEG2000 for rate-distortion comparison. As both plots indicate, the DDWTVC and DDWT-BISK systems perform similarly when using the dyadic transform structure. However, DDWT-BISK with the anisotropic transform achieves significantly higher PSNR levels than those of the other methods.

#### V. CONCLUSIONS

Prior work has indicated that the 3D DDWT can provide a video-signal representation with properties useful for video compression. Those promising findings led us to adapt the BISK algorithm for the coding of DDWT coefficients. Our proposed coding scheme is applied to video data after it undergoes an iterative projection-based noise-shaping procedure to reduce the number of non-zero DDWT coefficients. Whereas the existing DDWTVC coder exploits correlation as it exists across subbands in the DDWT, our DDWT-BISK coder additionally exploits coherent regions of insignificant coefficients that occur within subbands, a coherence that must necessarily be substantial due to the sparsity imposed by the noise-shaping process. When the DDWT uses a dyadic wavelet decomposition, our DDWT-BISK coder provides rate-distortion performance similar to that of both the DDWTVC coder as well as JPEG2000 applied using a temporal DWT and no motion estimation or compensation. However, the real advantage of the DDWT-BISK system is revealed when a DDWT with an anisotropic wavelet decomposition is used. While JPEG2000, when using this anisotropic transform, yields more or less unchanged performance as compared to the dyadic transform, DDWT-BISK consistently achieves substantial gains, with PSNR levels often 1 dB or more over both JPEG2000 as well as DDWTVC.

The anisotropic decomposition has many more subbands than does the dyadic transform, increasing the directionality of the decomposition, while at the same time decreasing the size of the subbands. The increased directionality of the decomposition appears to increase the sparsity of the significant coefficients. On the other hand, the anisotropic subbands are much smaller than their dyadic counterparts, reducing the capability of set-partitioning algorithms like DDWT-BISK to group large spatiotemporally contiguous regions of insignificant coefficients together into sets. The experimental results here suggest that the benefits of the first effect (increased directionality) must substantially outweigh the detriments of the second (decreased subband size).

While an anisotropic decomposition would appear to increase directionality within a critically sampled DWT as well, this additional directionality apparently has little impact on DWT-based coders since the DWT has only limited directionality to start with. Furthermore, while the anisotropic transform proves to be beneficial for our DDWT-BISK coder, it is unclear whether similar gains would be seen for other DDWT-based coders. DDWTVC, for example, is confined to using the dyadic DDWT; on the other hand, the DDWT-BISK coder can be applied effectively to any subband tiling.

#### ACKNOWLEDGMENT

The authors thank Y. Wang and B. Wang for helpful source code and discussions.

#### REFERENCES

- [1] N. G. Kingsbury, “Complex wavelets for shift invariant analysis and filtering of signals,” *Journal of Applied Computational Harmonic Analysis*, vol. 10, pp. 234–253, May 2001.

TABLE I

DISTORTION AVERAGED OVER ALL FRAMES OF THE SEQUENCE FOR RATE OF 0.5 BPP (1520 KBPS).

	PSNR (dB)			
	DDWT-BISK dyadic	DDWT-BISK anisotropic	JPEG2000 packet	JPEG2000 anisotropic
Stefan	30.4	31.5	30.5	30.7
Mobile	29.4	30.5	28.0	27.7
Foreman	38.2	38.8	38.1	37.9
Coastguard	32.1	33.9	32.8	33.1
Table Tennis	33.7	35.6	35.1	35.2

Sequences are CIF (352 × 288) with 80 frames at 30 Hz.

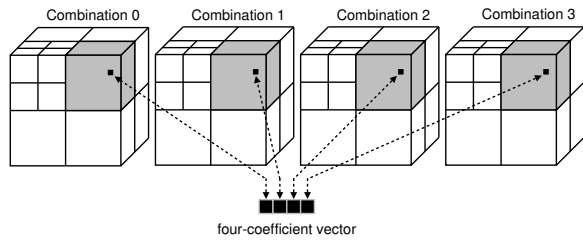


Fig. 1. A DDWT formed from four transform combinations produced from dyadic DWTs. Co-located coefficients in each of the four transform combinations form a four-coefficient vector.

[2] T. H. Reeves and N. G. Kingsbury, "Overcomplete image coding using iterative projection-based noise shaping," in *Proceedings of the International Conference on Image Processing*, vol. 3, Rochester, NY, June 2002, pp. 597–600.

[3] I. W. 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*, M. A. Unser, A. Aldroubi, and A. F. Laine, Eds. San Diego, CA: Proc. SPIE 5207, August 2003, pp. 607–618.

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

[5] —, "Video coding using 3-D dual-tree discrete wavelet transforms," in *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing*, vol. 2, Philadelphia, PA, March 2005, pp. 61–64.

[6] —, "Video coding using 3-D dual-tree wavelet transform," *EURASIP Journal on Image and Video Processing*, vol. 1, January 2007.

[7] J. E. Fowler, "Shape-adaptive coding using binary set splitting with  $k$ -d trees," in *Proceedings of the International Conference on Image Processing*, vol. 2, Singapore, October 2004, pp. 1301–1304.

[8] J. T. Rucker and J. E. Fowler, "Coding of ocean-temperature volumes using binary set splitting with  $k$ -d trees," in *Proceedings of the International Geoscience and Remote Sensing Symposium*, vol. 1, Anchorage, AK, September 2004, pp. 289–292.

[9] —, "Shape-adaptive embedded coding of ocean-temperature imagery," in *Proceedings of the 40<sup>th</sup> Asilomar Conference on Signals, Systems, and Computers*, Pacific Grove, CA, October 2006.

[10] 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, December 2000.

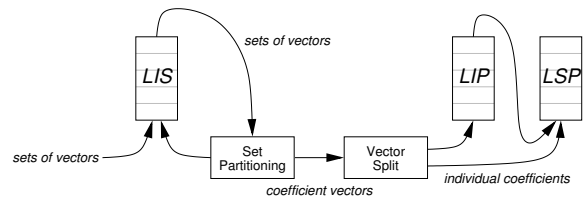


Fig. 2. The set-partitioning process of the DDWT-BISK coder. The LIS processes sets of coefficient vectors. Once vectors leave the LIS, they are split into individual coefficients—significant coefficients go to the LSP, insignificant coefficients go to the LIP.

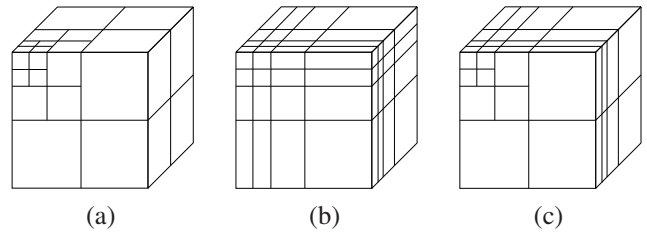


Fig. 3. Three-level wavelet decomposition for (a) "dyadic," (b) "anisotropic," and (c) "packet" decompositions.

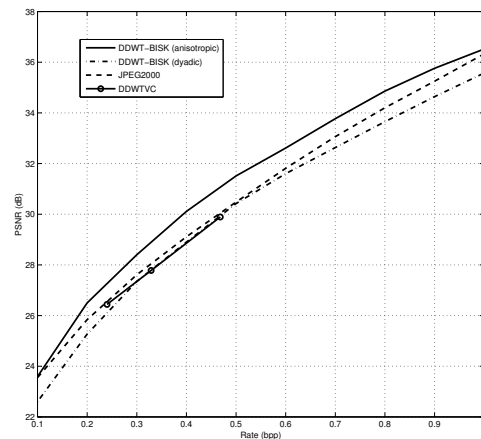


Fig. 4. Rate-distortion performance for "Stefan."

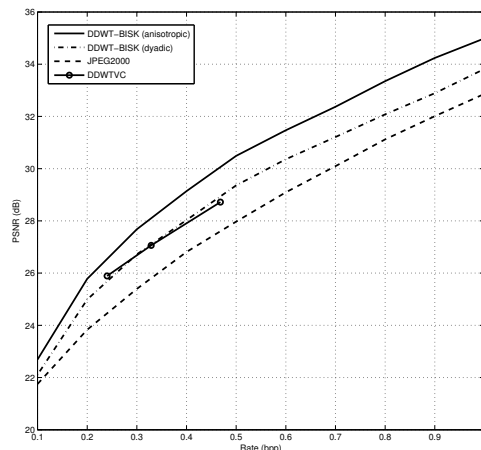


Fig. 5. Rate-distortion performance for "Mobile-Calendar."