Motion Differential SPIHT for Image Sequence and Video Coding

CIPR Technical Report TR-2010-4

Yang Hu and William A. Pearlman

November 2010

Center for
Image Processing Research

Rensselaer Polytechnic Institute
Troy, New York 12180-3590
http://www.cipr.rpi.edu

Motion Differential SPIHT for Image Sequence and Video Coding

Yang Hu and William A. Pearlman
Department of Electrical, Computer, and Systems Engineering
Rensselaer Polytechnic Institute, Troy, NY 12180, U.S.A.

Abstract

Efficient image sequence coding exploits both intra- and inter-frame correlations. SPIHT is efficient in intra-frame decorrelation for still images. Based on SPIHT, differential-SPIHT removes inter-frame redundancy by re-using the significance map of a SPIHT coded frame. The motion differential SPIHT (MD-SPIHT) automatically decides the coding methods for each frame, according to the inter-frame correlations in terms of PSNR. It can be seen as coding correlated sources (frames) into common and private messages, where the common message is the re-used significance map. From the simulation results of hyperspectral images, 3D/4D medical images, and video, MD-SPIHT is more efficient, compared with conventional 2D SPIHT and motion JPEG2000, while retaining the desirable features of random frame access, flexible frame rates, scalable bit rates, parallel coding, and computational simplicity.

Index Terms

Video coding, image sequence coding, motion coding, SPIHT, significance map.

I. Introduction

Image sequence coding achieves high efficiency by exploiting both spatial (intra-frame) and temporal (inter-frame) correlations. To exploit temporal correlation, there are two main approaches: inter-frame prediction and three-dimensional (3D) wavelet transform. Inter-frame prediction is the more computationally intensive and is widely used in the MPEGx and H.26x

We gratefully acknowledge the support of the Office of Naval Research under Contract No. N00014-05-1-0507.

video coding standards. In the 3D wavelet transform approaches, both 2D and 3D zerotree coding methods can be used. For example, 3D JPEG2000 coding can be realized with a 3D wavelet transform followed either by coding cubic subblocks [18] or by coding planar (x-y) subblocks in every transformed frame [8]. To remove the intra-frame correlation, many methods have been developed, such as the transfromation, intra-frame prediction, and so on. SPIHT [14], as the state-of-the-art image coding algorithm, is well-known for its efficiency in intra-frame de-correlation. However, to expand SPIHT to image sequence coding, some effective inter-frame de-correlation method is required to achieve high coding efficiency.

In this paper, we present an image sequence coding system based on SPIHT called *motion differential SPIHT* (MD-SPIHT), employing the inter-frame de-correlation approach *differential-SPIHT*. This approach was initially proposed by us in [7], and, to the best of our knowledge, it was the first such approach designed for SPIHT or other set partition coding methods to remove inter-frame correlation. In MD-SPIHT, a frame is coded either by conventional 2D SPIHT or by differential-SPIHT, where the inter-frame correlation is exploited by re-using the significance map of a previously SPIHT coded frame.

This system gives excellent results in compressing hyperspectral images, 3D and 4D medical images, and video with low to moderate motion. One desirable advantage of this method is that, unlike 3D wavelet transform or predictive and some other differential methods, random access to any frame is assured, because each one is encoded and decoded individually. It thus exhibits the desirable features of easy editing and robustness to error prone environments. The MD-SPIHT also retains the SPIHT properties of a rate-scalable bit-stream and low computational complexity.

A brief description of SPIHT is given in section II. In section III, MD-SPIHT is presented. Simulation results are given in section IV, and section V is the conclusion.

II. Conventional 2D SPIHT

SPIHT [14] is a state-of-the-art image coding method. It is well-known for both its efficiency and simplicity. SPIHT applies progressive bit-plane coding along the spatial-orientation trees on the wavelet coded coefficients, and achieves an embedded bit-stream. The basic coding process of SPIHT is introduced in this section.

SPIHT iteratively tests wavelet coefficients by comparing them with the threshold of each iteration, called *significance tests*. Let $C(i, j)$ be the wavelet coefficient at location (i, j). The thresholds are usually chosen as $T = 2^n$, starting from integer $n = n_{max} = \lfloor \log_2(\max_{(i,j)} |C(i,j)|) \rfloor$, and decreases by one at each following iteration. There is a flexible number of iterations and each iteration includes a sorting pass and a refinement pass.

The 2D wavelet transformed structure is viewed as a collection of non-overlapping trees rooted in coefficients of the lowest frequency subband. When the magnitudes of all coefficients in a set (a tree or sub-tree) are less than the current threshold, the set is deemed *insignificant* and a '0' is sent to the codestream; otherwise, the set is deemed *significant* and a '1' is sent. When a set tests as significant, it is partitioned into four individual coefficients (tree offspring) and the set of all descendants of these offspring. Individual coefficients are compared to the current threshold and signified as either significant or insignificant by sending the appropriate bits to the codestream. Initially, all coefficients and all sets rooted at the lowest subband are insignificant. When an individual coefficient tests as significant, its sign bit is also sent and its coordinates are moved to an ordered list, called the LSP (List of Significant Points). These tests proceed until all the previously insignificant sets and coefficients have been tested at the operative threshold. This procedure is called *sorting pass*.

After a sorting pass at threshold $T = 2^n$ (bit plane n), the bits in the n-th bitplane of LSP coefficients, which are previously found significant at higher thresholds, are sent to the codestream. This procedure is called *refinement pass*. Then the threshold is lowered by a factor of 2, or equivalently the n is lowered by 1, for the next iteration.

The output of significance tests from the sorting passes is called the *significance map*, which conveys the execution path of the encoder to the decoder. The sign bits constitute another class of data from the sorting passes. A third class contains the refinement bits from the refinement passes. The SPIHT codec is shown in Figure 1, where *map*(*), *sign*(*), and *ref*(*) indicate the significance map, sign bits, and refinement bits, respectively. Among them, the significance map consumes most of the codestream. For 512×512 grayscale Lena, the portions of the three kinds of data are provided in Figure 2 for different recovery PSNR, and the corresponding bit rates are given in Table I. For commonly used recovery PSNR (from 30 to 40 dB), the significance map consumes about 70% of the codestream, Figure 2(a); after applying Arithmetic Coding (AC) to the significance map, it still consumes about 60% of the codestream, Figure 2(b).

If 2D-SPIHT is used to code every frame of an image sequence, the significance map will take a large percentage of the codestream. MD-SPIHT reduces the bit consumption of the significance

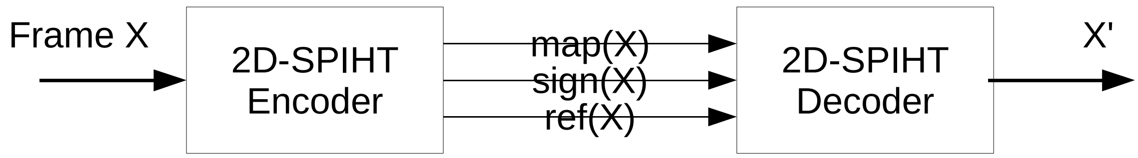
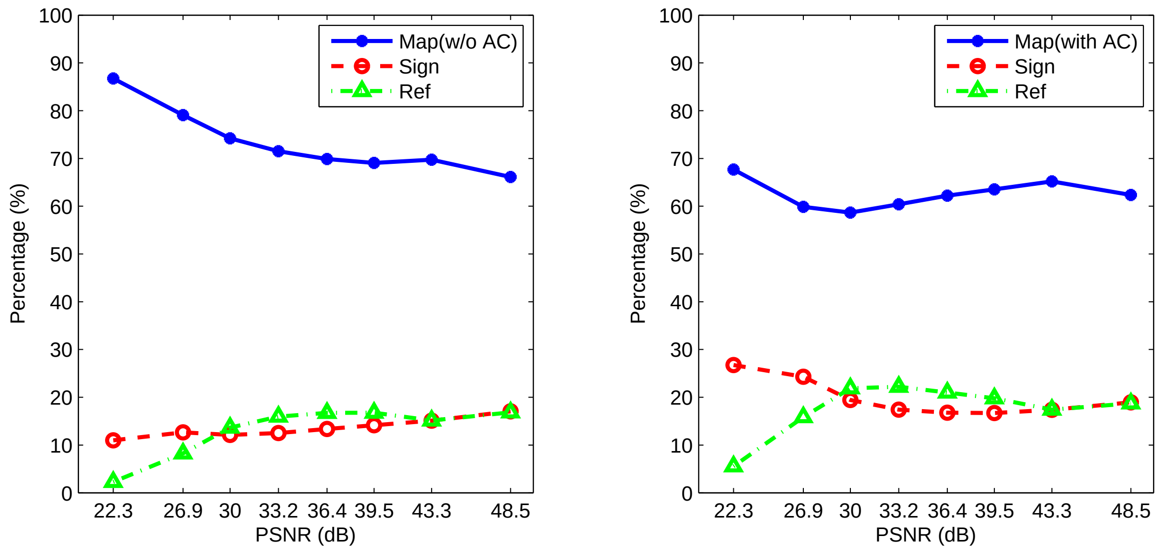


Fig. 1. Conventional 2D SPIHT.



(a) Without arithmetic coding. (b) Significance map with arithmetic coding

Fig. 2. Percentages of the three kinds of SPIHT output data.

map by means of re-using one map among several highly correlated (similar) frames, and thus reduces the overall bit rates. The details of this method is given in the following section.

III. Motion Differential SPIHT

The Motion Differential SPIHT (MD-SPIHT) has a frame-by-frame coding structure, similar to the motion JPEG [13] and motion JPEG2000 [9]. In MD-SPIHT, the frames of an image sequence are compressed either as *reference-frames* or as *differential-frames*. The former are coded with the conventional 2D SPIHT. The latter are coded with differential-SPIHT, which utilizes the significance map and the sign data of a previously coded reference-frame to predict the locations and signs of the significant coefficients of the current frame.

	Bit Rate (bpp)	
PSNR (dB)	Without AC	Map with AC
22.3	0.062	0.025
26.9	0.099	0.052
30.0	0.152	0.095
33.2	0.245	0.176
36.4	0.417	0.332
39.5	0.748	0.634
43.3	1.527	1.328
48.5	2.744	2.470

TABLE I
THE BIT RATES FOR DIFFERENT RECOVERY PSNR.

A. Frame Structure

In an image sequence, after each reference-frame, there could be N ($N = 0, 1, 2, ...$) differential-frames. In this "1-reference-N-differential" frame structure, there are two methods to decide the N. First, N can be fixed to some value for the whole image sequence. For example, if $N = 1$ for the whole sequence, then every other frame is coded as a reference-frame, and the frame after every reference-frame is a differential-frame re-using the significance map of its previous reference-frame. This fixed N method is simple, but it requires the inter-frame correlation to be stationary for the whole sequence to achieve high efficiency. However, most image sequences do not satisfy this requirement, and thus it is hard to make every pair of reference- and differential-frames to be highly correlated with a fixed N.

The second method is to determine the N adaptively during the coding procedure. To be concrete, some correlation metric, such as PSNR (peak signal-to-noise ratio), is adopted, and a threshold T_{PSNR} is selected in terms of this metric. If the correlation between the reference-frame and the current frame is greater than the threshold T_{PSNR}, the current frame will be coded as a differential-frame; otherwise, the current frame will be coded with 2D SPIHT and serves as a new reference-frame, until a newer reference-frame appears.

The codec dealing with such a frame structure is illustrated in Figure 3, where the frames X and Y represent the reference- and differential-frames, respectively. This coding scheme is called *Motion Differential SPIHT* (MD-SPIHT), since it has a similar framework with other so-called *motion-coding* methods, such as the motion JPEG and motion JPEG2000, in that there are no

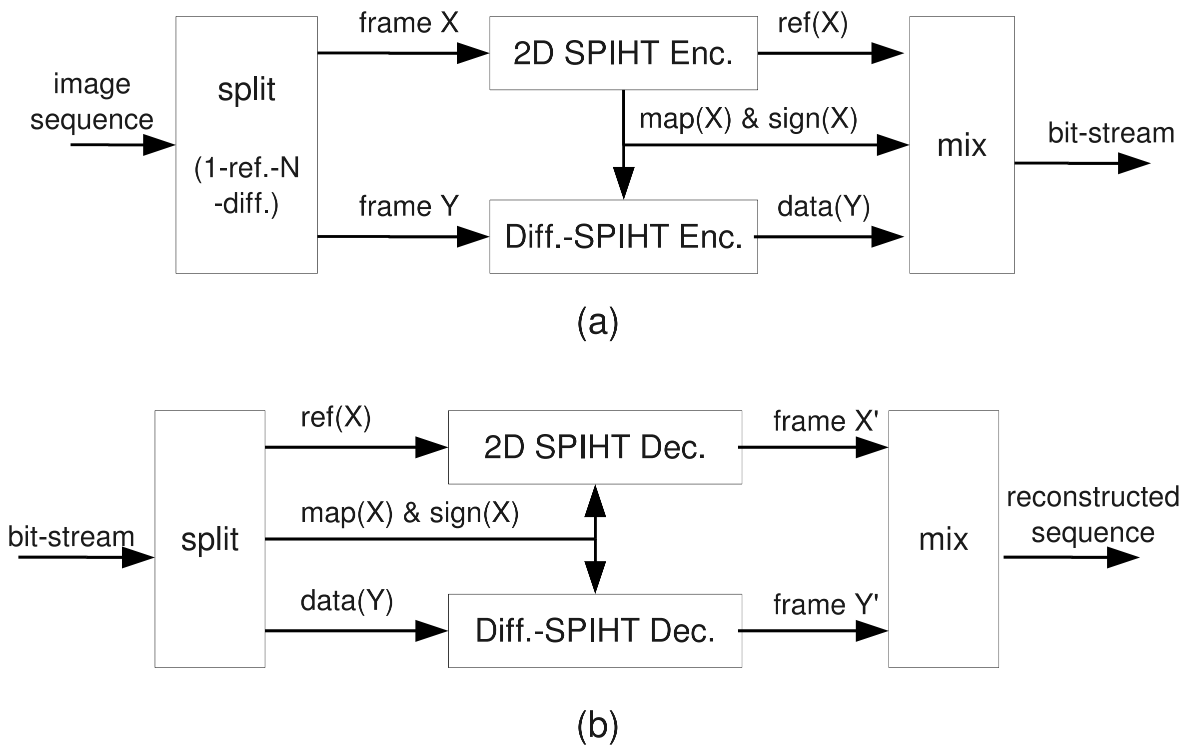


Fig. 3. Motion Differential SPIHT (a) encoder (b) decoder.

inter-frame interactions.

B. Correlation Analysis

For most image sequences, the neighboring frames are correlated. The wavelet coefficients, $C_X(i,j)$ and $C_Y(i,j)$, at the same position, (i,j), in adjacent frames, X and Y, maintains the correlation. Given some threshold 2^n, four relationships between the wavelet coefficients $C_X(i,j)$ and $C_Y(i,j)$ are important to us.

1) Both $C_X(i,j)$ and $C_Y(i,j)$ are significant.
2) Both $C_X(i,j)$ and $C_Y(i,j)$ are insignificant.
3) $C_X(i,j)$ is significant and $C_Y(i,j)$ is insignificant.
4) $C_X(i,j)$ is insignificant and $C_Y(i,j)$ is significant.

If X and Y are highly correlated, the first two relationships are much more likely to happen.

C. Differential-SPIHT

Differential-SPIHT is the coding scheme applied to the differential-frames, Y's. Let the threshold in the last iteration be T_{last}. The significance map, *map*(X), conveys the information that identifies the positions of the significant coefficients at each iteration for frame-X, and such positions are called *significant locations*. The remaining locations with insignificant coefficients with respect to T_{last} are called *insignificant locations*. Therefore, the significance map distinguishes the significant and insignificant locations with respect to T_{last}.

For frame X, all the coefficients on the significant locations are significant, and on the insignificant locations are insignificant. However, applying *map*(X) to frame Y, there are four possible cases, which are the alternative expressions of the four relationships in Section III-B.

- I Significant location with significant coefficient
- II Insignificant location with insignificant coefficient
- III Significant location with insignificant coefficient
- IV Insignificant location with significant coefficient

To make sure no reconstruction fidelity loss compared to the conventional SPIHT with T_{last}, all significant coefficients with respect to T_{last} should be coded, that is, all the significant coefficients in the significant locations (Case I) and in the insignificant locations (Case IV) should be coded. The stages *SPIHT** and *differential-sorting* in differential-SPIHT cover the Case I and IV, respectively. The two stages are represented in Figure 4.

1) SPIHT:* The SPIHT* is an important step in MD-SPIHT. It re-uses the significance map of a previously coded map-frame to exploit the inter-frame correlation and codes the significant coefficients at those locations which are indicated by the significance map. Different from conventional SPIHT, there is no significance test to generate a new significance map. Instead, SPIHT* reads in the significance map of X, *map*(X), to iteratively identify the possibly significant positions for Y. At each iteration, the selected significant positions of Y are exactly the same as those of X. The Y coefficients in these significant positions are usually significant with a high probability, due to the correlation between X and Y. However, there are other possible cases, because of the difference of X and Y.

Let (i, j) represent the significant location at the iteration with threshold T. The coefficient $C_Y(i, j)$ involves the sign and the magnitude. For the sign of $C_Y(i, j)$,

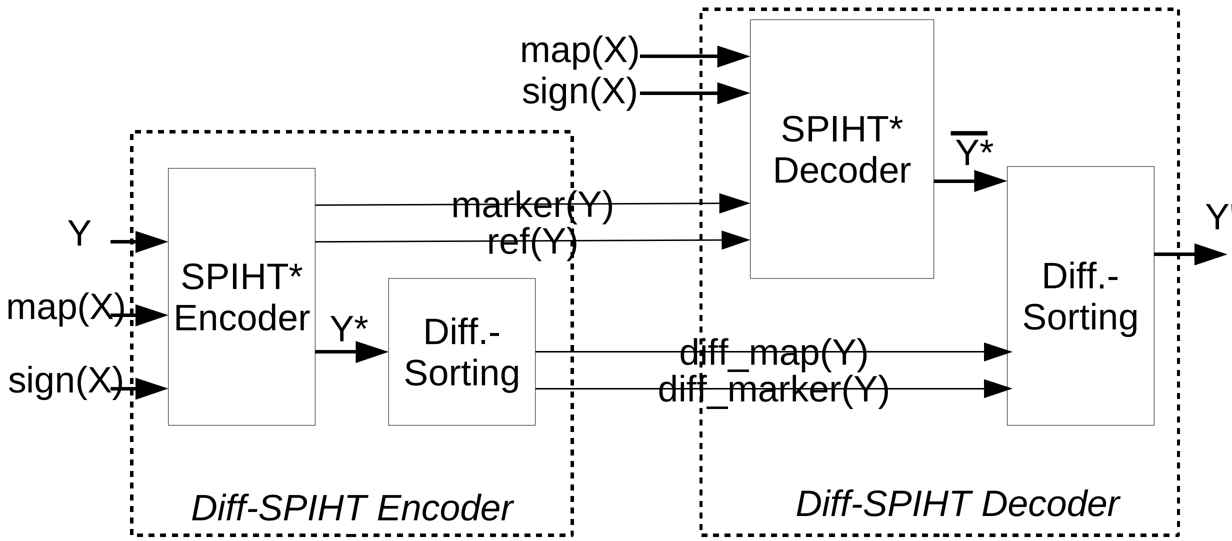


Fig. 4. Differential-SPIHT

- it may be the same as the sign of $C_X(i,j)$;
- it may be different from the sign of $C_X(i,j)$.

The sign of $C_X(i,j)$ is available for SPIHT* from the read-in *sign*(X). For the magnitude of $C_Y(i,j)$,

- it may be insignificant ($|C_Y(i,j)| < T$);
- it may be *just* significant ($T \leq |C_Y(i,j)| < 2T$);
- it may be *super* significant ($|C_Y(i,j)| \geq 2T$).

Instead of a sign bit output in the conventional 2D SPIHT, a marker symbol is sent to indicate the specific case to which $C_Y(i,j)$ belongs. The marker symbols can be entropy coded with Huffman codes based on the statistics gathered from simulations, or with the more effective adaptive arithmetic coding [17]. These marker data are denoted by *marker*(Y) in Figure 4. Then the coefficient is updated to be $|C_Y(i,j)|$ or $(|C_Y(i,j)| - T \times m)$ for the insignificant and (just/super) significant cases, respectively, where $m = \lfloor |C_Y(i,j)|/T \rfloor$. In our simulation, small m's are represented by the marker symbols; large m's are coded with Exponential-Golomb code of order-0 (EG-0 code) [16], which is a prefix-free code used in H.264 texture coding for the large coefficient values.

The refinement pass of SPIHT* is similar to that of conventional SPIHT. Given $T = 2^n$, the bits in bit-plane n of the coefficients that are previously found significant are outputted as *ref*(Y) in Figure 4. Then, update the just refined coefficients by setting the bits in bit-plane n to be 0's. Restricted by the input significance map, the SPIHT* has the same number of iterations as the SPIHT procedure, which produces the significance map. The Y^* in Figure 4 denotes the frame with updated coefficients after all SPIHT* iterations. So far, all the significant coefficients in the significant locations (Case I) are coded.

2) Differential-Sorting: Differential-sorting in Figure 4 is adopted to code the significant coefficients in the insignificant locations (Case IV). In Differential-sorting, the threshold is set to be T_{last}, and the sorting is enacted on the initial spatial orientation trees. The significance tests are performed as in conventional SPIHT. The significance map bits convey the positions of significant coefficients of Y^* and are denoted by *diff_map*(Y). For each significant coefficient, instead of the sign bit output for conventional SPIHT, a marker symbol is outputted to indicate the sign and the magnitude. Unlike *marker*(Y) in SPIHT*, these markers indicate the signs ($+$ or $-$) directly, and there is no such case that the magnitude is insignificant. Similarly, Huffman code can be used to code the markers according to the statistics gathered from simulations, or more effectively adaptive arithmetic coding can be adopted. The large m's in the super significance case are also EG-0 coded as in the SPIHT*. These marker data are denoted by *diff_marker*(Y) in Figure 4.

In Differential-SPIHT decoding, differential-sorting follows SPIHT* decoding to make sure all the significant coefficients are reconstructed.

D. Performance Analysis

Given a T_{last}, MD-SPIHT guarantees no PSNR loss, compared to 2D SPIHT, because it encodes and reconstructs all the coefficients with magnitude larger than or equal to T_{last}, just as 2D-SPIHT does. Combined with reversible integer transforms, MD-SPIHT is able to compress losslessly, which is required by some medical image applications.

Although MD-SPIHT exploits inter-frame correlation, it has essential differences from the 3D DWT or predictive or other differential methods. First, without inter-frame interactions, multiple differential-frames can be coded in parallel, random frame access is supported, and there is no latency. Secondly, it maintains low computational complexity, because there is no special temporal decorrelation process, unlike the computation-consuming inter-frame prediction and

3D DWT. Furthermore, unlike predictive and other differential methods, the encoder need not simulate decoding to create reference frames. Finally, it reduces the memory requirements by buffering only the reference frame and its significance map and sign data, instead of groups of frames.

IV. Simulation Results

MD-SPIHT is simulated to compare with other coding schemes having the frame-by-frame coding structure. One is the conventional 2D SPIHT, which is applied consecutively to every frame of the image sequence. The other is motion JPEG2000 (MJ2K). It codes each frame as a JPEG2000 [8] image, and is the digital cinema distributed standard. The MD-SPIHT uses the adaptive ”1-reference-N-differential” frame structure. Specifically, the PSNR (peak signal-to-noise ratio) serves as the correlation metric. We calculate the PSNR between the reference-frame and the current frame, and compare it with the threshold T_{PSNR} (in dB) to decide the current frame to be coded with differential SPIHT or conventional 2D SPIHT. If the current frame is coded with conventional 2D SPIHT, it will serve as the reference-frame for the following frames, until a newer reference-frame appears.

The PSNR of two images X and Y is based on the mean squared error (MSE) of these two images. Assuming images of size $m \times n$, the MSE is

$$MSE = \frac{1}{mn} \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} [X(i,j) - Y(i,j)]^2 \tag{1}$$

Then the PSNR is defined as

$$PSNR = 10 \log_{10} \left(\frac{MAX^2}{MSE} \right) = 20 \log_{10} \left(\frac{MAX}{\sqrt{MSE}} \right) \tag{2}$$

In the previous equation, MAX is the maximum fluctuation in the image pixels. When the pixels are represented using 8 bits per sample, such as in the tested video and medical images, $MAX = 2^8 - 1 = 255$. When the pixels are represented using 16 bits per sample, such as in the tested hyperspectral images, $MAX = 2^{16} - 1 = 65535$.

In simulation, all schemes adopt 3-level 2D DWT and arithmetic coding. As in the JPEG2000, the 5/3 [4] and 9/7 [3] filters are adopted for lossless and lossy coding, respectively. The comparisons are in terms of the compression efficiency in bpp (bits per pixel).

A. Simulation Results of Hyperspectral Images

Hyperspectral images have been used in geology, atmosphere, ecology, etc. domains as a mature technology. The Airborne Visible/Infrared Imaging Spectrometer (AVIRIS) is a state-of-the-art hyperspectral sensor. According to [1], its image consists of 224 contiguous spectral bands covering wavelengths from 400 to 2500 nanometers (nm) with approximately 10 nm spectral resolution. AVIRIS data include raw, radiance, and reflectance images. The raw images are converted to radiance images by means of the radiometric calibration, and the radiance images are further converted to reflectance images through correcting the atmospheric effects. The radiance or reflectance data are used most, depending on the applications. For most analyses, they give equivalent results.

Hyperspectral images are typically intended for automatic analysis by computers. The distortion caused by lossy compression, which is typically designed based on human visual system, is not acceptable for most applications [11]. In our tests, we apply lossless compression to AVIRIS radiance data. Prior to compression, we resize the tested AVIRIS scene from 224 bands of size 614×512 to 224 bands of size 512×512, by discarding the 102 columns on the right of each band. The tested volume images are the resized scenes ($512 \times 512 \times 224$, 16-bit depth), and each spectral band is viewed as a frame. MD-SPIHT exploits the spectral (inter-band) correlation.

Figure 5 presents the lossless compressed bit rates at different T_{PSNR} for *cuprite* scene 01. We can see that the MD-SPIHT is most efficient at $T_{PSNR} = 45$ dB, at which there are 37 reference-frames and 187 differential-frames. Here, $T_{PSNR} = 45$ dB is used to compress the other hyperspectral images and the results are given in Table II, where the 3D SPIHT [5] results are also provided as a reference.

From the testing results, MD-SPIHT outperforms the 2D SPIHT and MJ2K by saving about 9% and 11% bit rates, respectively. Although MD-SPIHT costs about 15% more bits compared with 3D SPIHT, it is worthwhile to indicate that the lower bit rates achieved by 3D SPIHT is at the price of higher system complexity and the loss of temporal (band-crossing) flexibilities. The 3D SPIHT uses 3D coding units (spatial-temporal orientation trees) on the hierarchical structure of 3D wavelet transform, where the 3rd dimensional transform increases computational complexity. In addition, the group of frames (GOF) of 3D SPIHT requires much larger memory, and makes the random frame access at only the GOF level. If applied to video, 3D SPIHT may

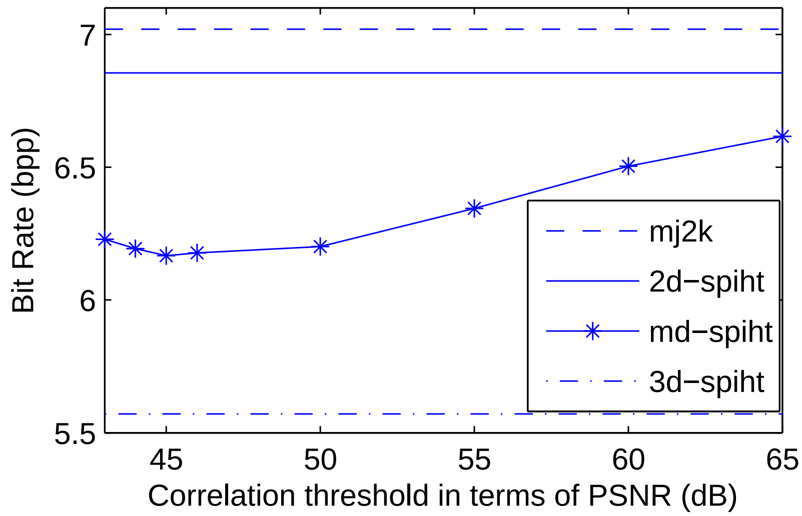


Fig. 5. Lossless bit rates at different correlation thresholds

Sequence	SPIHT MD	SPIHT 2D	SPIHT 3D	MJ2K
cuprite sc04	6.27	6.93	5.63	7.10
moffet field sc01	6.16	6.85	5.71	7.57
moffet field sc03	6.27	6.93	5.08	6.29
jasper ridge sc01	7.17	7.70	5.94	7.89
jasper ridge sc03	7.15	7.68	5.92	7.86
average (normalization)	**6.60** (100 %)	**7.22** (109.4 %)	**5.66** (85.8 %)	**7.34** (111.2 %)

TABLE II
HYPERSPECTRAL IMAGE LOSSLESS BIT RATES (BPP)

cause latency for large GOF size.

B. Simulation Results of 3D/4D Medical Images

Medical image sets, such as images generated by computer tomography (CT) and functional Magnetic Resonance (MR), are increasingly used in diagnosis. The 3D volumetric images are two-dimensional image slices that represent cross sections of a subject. Four-dimensional (4D) medical images, which can be seen as a time series of 3D images, represent the live action of human anatomy and consume even larger amounts of resources in transmission and storage than

Sequence	SPIHT MD (T_{PSNR})	2D	3D	MJ2K
CT_aperts	1.16 (33 dB)	1.28	0.92	1.22
CT_carotid	1.88 (33 dB)	2.04	1.56	1.96
CT_skull	3.01 (27 dB)	3.37	2.34	3.32
CT_wrist	1.97 (35 dB)	2.21	1.48	2.07
MR_liver_t1	3.05 (33 dB)	3.32	2.34	3.25
MR_ped_chest	2.63 (40 dB)	3.05	1.93	3.00
MR_sag_head	2.65 (31 dB)	2.88	2.19	2.91
average (normalization)	**2.34** (100 %)	**2.59** (110.7 %)	**1.82** (77.8 %)	**2.53** (108.1 %)

TABLE III
3D MEDICAL IMAGE LOSSLESS BIT RATES (BPP)

3D image data. In medical applications, lossless compression is extremely important, because any distortion may lead to an faulty diagnosis. In simulations, lossless compression are performed on both 3D and 4D medical images. MD-SPIHT has specific T_{PSNR} for every subject. All the tested medical images are of 8-bit depth.

For 3D medical images, every 2D slice is seen as a frame, and the MD-SPIHT exploits the dependency between slices. The tested 3D CT images are slices 1-64, and 3D MR images are slices 1-48, all of size 512×512. Simulation results are provided in Table III, where MD-SPIHT outperforms 2D SPIHT and MJ2K in terms of saving compressed bit rates by about 11% and 8%, respectively.

4D medical images are time series of 3D volume images. Since they are typically temporally smooth, MD-SPIHT exploits the inter-volume (temporal) dependency. Assume that a 2D slice F_n in volume n is a reference-frame coded with 2D SPIHT. If the corresponding 2D slice F_{n+1} in volume $n+1$ is highly correlated with F_n, then F_{n+1} will be coded with differential SPIHT referring to the map of F_n; otherwise, F_{n+1} will be coded with 2D SPIHT and serve as a new reference-frame. Then code the corresponding slice F_{n+2} in volume $n+2$, with differential SPIHT referring the map of F_n or F_{n+1}, or with 2D SPIHT and serve as a new reference-frame. Simulations are performed on 4D images given in Table IV, and the compressed bit rates are provided in Table V. On average, MD-SPIHT saves compressed bit rates 2% and 4%, respectively, compared with 2D SPIHT and MJ2K.

image	type	volumes	slices	size
volunteer4d	N/A	001 - 016	001 - 016	128×120
ct4d	CT	001 - 018	001 - 192	256×256
siem	fMRI	001 - 112	001 - 016	64×64

TABLE IV
INFORMATION OF THE TESTED 4D MEDICAL IMAGES

Sequence	MD-SPIHT (T_{PSNR})	2D SPIHT	MJ2K
volunteer4d	3.28 (39 dB)	3.39	3.35
ct4d	3.17 (41 dB)	3.33	3.24
siem	4.81 (37 dB)	4.81	5.13
average (normalization)	**3.75** (100 %)	**3.84** (102.4 %)	**3.91** (104.3 %)

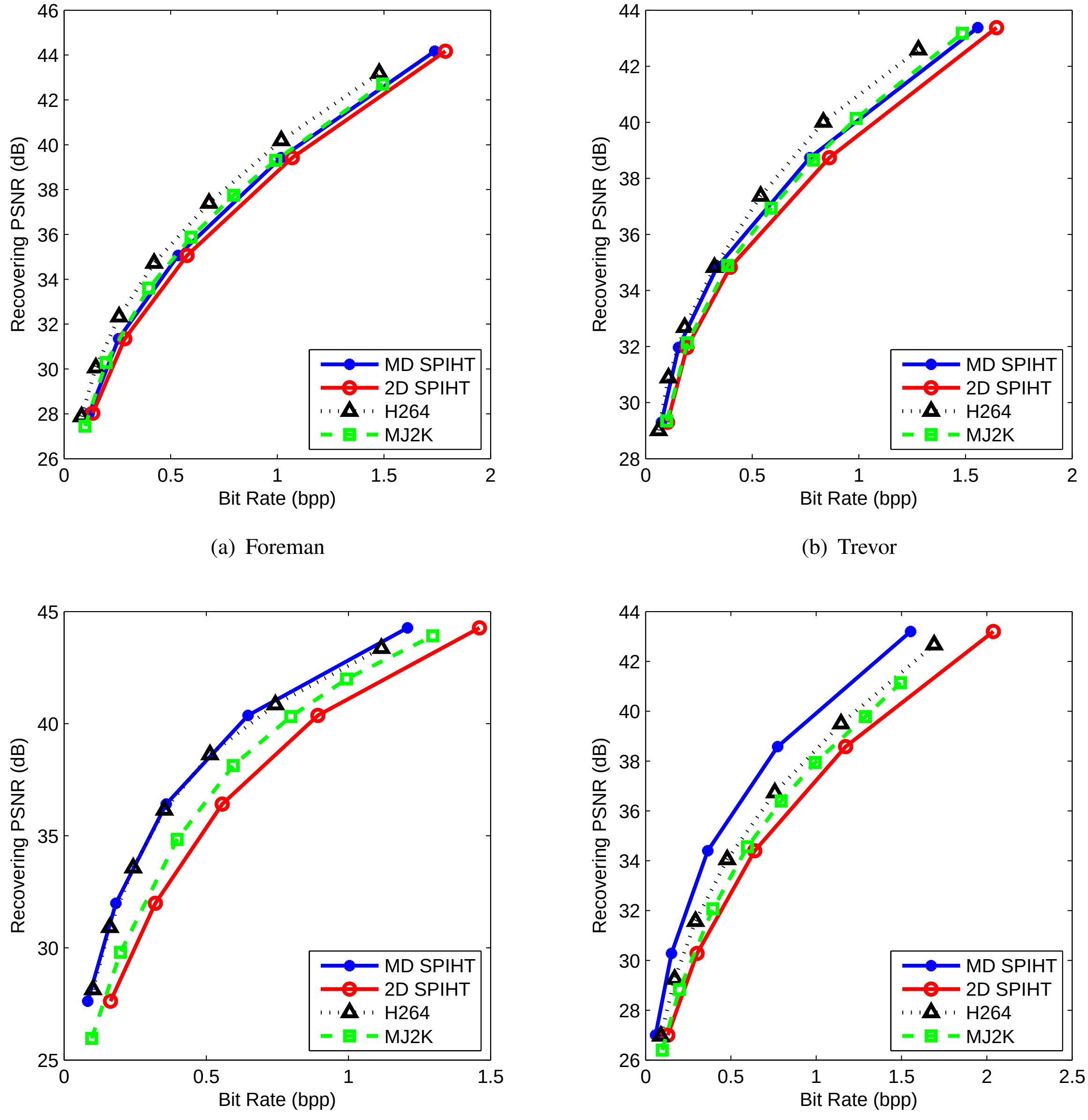
TABLE V
4D MEDICAL IMAGE LOSSLESS BIT RATES (BPP)

C. Simulation Results of Video

Lossy compression is performed for video sequences *foreman* (352×288, 300 frames), *salesman* (360×288, 449 frames), *hall monitor* (352×288, 300 frames), and *trevor* (256×256, 99 frames). As in hyperspectral image compression, a simulation is first performed on *foreman* at different T_{PSNR}'s, and we get the optimal $T_{PSNR} = 27$ dB, which is adopted by the MD-SPIHT for the video simulation in this sub-section.

The simultation results are provided in terms of the rate-distortion curves in Figure 6. As a reference, H.264 [10] all intra coding results are also given in the figures. Tests are with the H.264 reference software JM 11.0 [2] default configuration, including CABAC (context-adaptive binary arithmetic coding), Rate-Distortion optimization, and other advanced tools. It is worthwhile to indicate that the H.264 all intra coding is a different framework with the other tested methods. It is based on DCT, instead of DWT; its spatial-scalability and rate-control is based on a layered structure, which is not as flexible as the DWT based methods; lossless coding is not supported by H.264.

From the test results, the MD-SPIHT outperforms the conventional 2D SPIHT. Actually, with a properly chosen T_{PSNR}, the conventional 2D SPIHT is the lower performance boundary of



(a) Foreman (b) Trevor

(c) Hall Monitor (d) Salesman

Fig. 6. Rate-distortion curves for tested video sequences.

MD-SPIHT. For example, the video sequence *calendar and train* includes complex movements, and the maximum inter-frame PSNR is about 23 dB only. With the $T_{PSNR} = 27$ dB, there are no differential SPIHT coded frames, and, therefore, the MD-SPIHT performs as the conventional 2D SPIHT.

Generally speaking, MD-SPIHT performs better for sequences with higher inter-frame correlation, i.e., with slower movements. For sequences with similar movements, such as the sequences *trevor*, *hall monitor*, and *salesman*, which are all with still camera, the degree of background complexity also decides the performance of MD-SPIHT. Complex texture requires many bits to encode, while simple texture requires much less. Therefore, MD-SPIHT is expected to show more improvement in efficiency in encoding complex, nearly static background than with simple background texture. The results verify this expectation. As the background complexity increases (from *trevor* to *hall monitor* to *salesman*), the performance of the MD-SPIHT increases accordingly. For *salesman* at 30 dB recovery PSNR, MD-SPIHT reduces the overall bit rate by about 50% of that of the 2D SPIHT or MJ2K.

V. Conclusion

MD-SPIHT is an effective system for image sequence compression. It applies the significance map of a SPIHT coded frame to the following one or more frames to remove the inter-frame correlations. From the simulation results of hyperspectral images, 3D/4D medical images, and video sequences, MD-SPIHT outperforms 2D SPIHT and MJ2K significantly, especially for sequences with high inter-frame correlation. At the same time, MD-SPIHT maintains such features as low computational complexity, low memory requirement, support of parallel coding, support of lossless compression, no latency, and no fidelity loss. We consider MD-SPIHT to be an attractive alternative for so-called *motion coding*, e.g., motion JPEG and motion JPEG2000, where frames of an image sequence are coded individually in order to preserve the desirable properties mentioned above. The idea of re-using the significance map to explore inter-frame correlation is presented based on SPIHT, but it can be extended to other set partition coding methods, such as SPECK [12], SBHP [6], EZBC, and EZW [15].

References

[1] AVIRIS Home Page. http://aviris.jpl.nasa.gov/.

[2] H.264/AVC Reference Software JM 11.0. http://iphome.hhi.de/suehring/tml/download/old_jm/.

[3] M. Antonini, M. Barlaud, P. Mathieu, and I. Daubechies. Image coding using wavelet transform. *IEEE Trans. on Image Processing*, 1(2):205–220, 1992.

[4] A. Calderbank, I. Daubechies, W. Sweldens, and B. Yeo. Lossless image compression using integer to integer wavelet transforms. In *International Conference on Image Processing*, pages 596–599, 1997.

[5] S. Cho and W. Pearlman. Error-resilient video coding with improved 3D SPIHT and error concealment. In *Proceedings of SPIE*, volume 5022, pages 125–136, 2003.

[6] C. Chrysafis, A. Said, A. Drukarev, A. Islam, and W. Pearlman. SBHP-a low complexity wavelet coder. In *Proc. ICASSP*, 2000.

[7] Y. Hu and W. Pearlman. Differential-SPIHT for image sequence coding. In *Proc. ICASSP*, 2010.

[8] ISO/IEC. Information Technology—JPEG 2000 Image Coding System, Part-2: Core Coding System. *ISO/IEC 15444-2*, 2001.

[9] ISO/IEC. Information Technology—JPEG 2000 Image Coding System, Part-3: Motion JPEG 2000. *ISO/IEC 15444-3*, 2001.

[10] ITU-T and I. J. 1. Advanced video coding for generic audio-visual services. *ITU-T Recommendation H.264 and ISO/IEC 14496-10 (MPEG-4 AVC)*, May 2003.

[11] J. Mielikainen. Lossless compression of hyperspectral images using lookup tables. *IEEE signal processing letters*, 13(3):157–160, 2006.

[12] W. Pearlman, A. Islam, N. Nagaraj, and A. Said. Efficient, low-complexity image coding with a set partitioning embedded block coder. In *IEEE Trans. Circuits & Syst. for Video Tech.*, volume 14, pages 1219–1235, 2004.

[13] W. Pennebaker and J. Mitchell. *JPEG still image data compression standard*. Kluwer Academic Publishers, 1993.

[14] A. Said and W. A. Pearlman. A new, fast, and efficient image codec based on set partitioning inhierarchical trees. *IEEE Trans. on Circuits & Syst. for Video Tech.*, 6(3):243–250, 1996.

[15] J. Shapiro. Embedded image coding using zerotrees of wavelet coefficients. *IEEE Trans. on Signal Processing*, 41(12):3445–3462, 2002.

[16] J. Teuhola. A compression method for clustered bit-vectors. *Information Processing Letters*, 7(6):308–311, 1978.

[17] I. Witten, R. Neal, and J. Cleary. Arithmetic coding for data compression. *Communications of the ACM*, 30(6):540, 1987.

[18] J. Xu, Z. Xiong, S. Li, and Y.-Q. Zhang. 3-D embedded subband coding with optimal truncation (3-D ESCOT). *J. Applied and Computational Harmonic Analysis*, 10(5):290–315, 2001.