

Lossy-to-Lossless Compression of Hyperspectral Imagery Using Three-Dimensional TCE and an Integer KLT

Jing Zhang, James E. Fowler, *Senior Member, IEEE*, and Guizhong Liu

Abstract—An embedded lossy-to-lossless coder for hyperspectral images is presented. The proposed coder couples a reversible, integer-valued Karhunen-Loève transform (KLT) with an extension into 3D of the tarp-based coding with classification for embedding (TCE) algorithm that was originally developed for lossy coding of 2D images. The resulting coder obtains lossy-to-lossless operation while closely matching the lossy performance of JPEG2000. Additionally, for lossless compression, it consistently outperforms not only JPEG2000 but often several prominent purely lossless methods.

Index Terms—tarp filtering, TCE, integer KLT, hyperspectral compression, lossy-to-lossless coding

I. INTRODUCTION

With increasing resolution and deployment of hyperspectral imaging devices, compression of hyperspectral datasets is becoming of critical interest in many application areas. There have been a number of compression techniques proposed for hyperspectral images, and these include both prediction-based and transform-based algorithms, capable of both lossy and lossless compression performance.

Lossless compression algorithms are typically prediction-based. For example, in 3D-CALIC [1], the predictor alternates between 2D intraband prediction and 3D interband prediction to best exploit both interband and intraband statistical redundancies. M-CALIC [2] is similar, except that interband prediction is always used, and all parameters and thresholds are adjusted to suit hyperspectral images. A clustered DPCM (C-DPCM) approach was taken in [3]; in C-DPCM, for each band, linear prediction minimizes the squared error within each cluster. Finally, lookup tables (LUTs) were used in [4] to expedite searching the previous band for a pixel identical to that co-located with the one to be coded.

For transform-based algorithms, coding with a discrete wavelet transform (DWT) dominates. The most widely-used approach is the JPEG2000 standard [5, 6]. Wavelet-based coders like JPEG2000 are typically developed for lossy compression but are capable of lossy-to-lossless performance when wavelet transforms that map integers to integers are used. Such

integer DWTs permit coders to provide completely reversible transformation as is needed for lossless coding. However, the inherently progressive nature of the DWT-based coders permits one to truncate the lossless bitstream at any point to produce a lossy representation of the dataset. Since the JPEG2000 standard supports integer-valued reversible transforms, JPEG2000 represents the most prominent example of such lossy-to-lossless coding. While lossy-to-lossless coding is considered to be a useful characteristic for certain applications, it is generally acknowledged that this capability comes at a cost—purely lossless algorithms like M-CALIC, C-DPCM, and LUT typically achieve more efficient lossless performance than their lossy-to-lossless counterparts.

In this paper, we propose a new lossy-to-lossless coder that closely matches the lossy performance of JPEG2000, outperforms JPEG2000 at lossless compression, and yields performance closer to—and often better than—that of purely lossless methods. The starting point is a 2D lossy coder called *tarp-based coding with classification for embedding* (TCE) [7]. Here, we extend this TCE coder to 3D and couple the resulting 3D-TCE with the reversible integer-valued Karhunen-Loève transform (KLT)¹ of [8–10] to yield lossy-to-lossless coding performance. Experimental results on several hyperspectral images demonstrate on the order of 3% better lossless compression than JPEG2000 imbued with an identical reversible KLT and performance often better than that of the purely lossless M-CALIC and LUT techniques. Additionally, lossy compression is very close to that of JPEG2000.

Below, we first present brief overviews of tarp coding, TCE, and the integer KLT in Sec. II before describing our proposed 3D-TCE approach in detail in Sec. III. Experimental results are presented in Sec. IV, while several concluding remarks are made in Sec. V.

II. BACKGROUND

A. Tarp Coding

Let $c[\mathbf{x}]$ be a field of real-valued transform coefficients. Given a threshold t , define the *significance state* of the coefficient at location \mathbf{x} to be

$$v[\mathbf{x}] = \begin{cases} 1, & |c[\mathbf{x}]| \geq t, \\ 0, & \text{otherwise.} \end{cases} \quad (1)$$

For a hyperspectral image, $c[\mathbf{x}]$ would be a 3D subband of wavelet coefficients and $\mathbf{x} = [x_1, x_2, x_3]$ a location within

¹The KLT is also known as *principal component analysis* (PCA).

J. Zhang and G. Liu are with the School of Electronics and Information Engineering, Xi'an Jiaotong University, Xi'an, China.

J. E. Fowler is with the Department of Electrical & Computer Engineering, Mississippi State University, Starkville, MS, USA, and the GeoResources Institute (GRI) at the Mississippi State High Performance Computing Colaboratory (HPC²).

This work was conducted while J. Zhang was visiting GRI at Mississippi State University as a joint Ph.D. candidate funded by China Scholarship Council under the State Scholarship Fund.

the subband. Regardless of the dimensionality of the dataset, however, many wavelet-based compression algorithms transmit the most significant bit of all coefficient magnitudes, then the next-most significant bit of all coefficient magnitudes, etc., such that each coefficient is successively approximated in a procedure known as *bitplane coding*. Fundamental to such bitplane coding is the coding of binary-valued fields, or *significance maps*, $v[\mathbf{x}]$, one for each threshold value t which is successively decreased as coding progresses (see [11] for an overview of wavelet-based coding for hyperspectral imagery).

Although there are many possible approaches for coding a significance map $v[\mathbf{x}]$, of particular interest here is the so-called tarp filter [12] which uses a Parzen window to estimate the probability, $p[\mathbf{x}]$, that $v[\mathbf{x}] = 1$. This probability estimate then is used to drive a non-adaptive arithmetic coder to efficiently code $v[\mathbf{x}]$. For a 1D signal, this probability is generated by a simple first-order recursive filter,

$$p[x] = \alpha p[x - 1] + (1 - \alpha)v[x - 1], \quad (2)$$

where $p[x]$ is the probability estimate for position x , $v[x]$ is the significance map at position x , and α is a learning-rate parameter.

In [12], the main focus is the coding of still images, so an extension of (2) is derived for 2D. In 2D, the tarp-filtering process of (2) becomes the successive application of three 1D filters [12]. In [13], the 2D-tarp algorithm of [12] was extended to 3D for the coding of hyperspectral imagery. In this case, the tarp filter of (2) becomes the successive application of five 1D filters. Specifically, in order to generate the probability estimate $p[\mathbf{x}]$, the information of three neighbors is used—one to the left, one above, and one in the same spatial position in the previous spectral band. In 3D-tarp, the five filters are as follows (see [13] for more detail). For each of the spectral bands, one filter runs each row from left to right, one filter runs this row from right to left, and then one filter runs each column from top to bottom. Then, after a full spectral band is coded with these three filters, another two filters are applied to update the probability in the spectral direction.

B. TCE

The probability estimate resulting from the tarp filter in the algorithms of [12, 13] is most accurate for the areas in significance map $v[\mathbf{x}]$ that are relatively sparsely populated with significant coefficients. In [7], the 2D-tarp algorithm of [12] was extended to make use of *fractional bitplane coding* wherein the significance map is processed in multiple distinct passes to improve the encoding order and, consequently, the compression performance. In the resulting system, TCE [7], coefficients are divided into three classes, and each class is handled by a different coding pass. These passes are:

- *Nonzero-Neighbor (NZN) Pass*: $v[\mathbf{x}]$ is coded using an adaptive arithmetic coder for all coefficients which have neighboring coefficients which are already significant.
- *Zero-Run (ZR) Pass*: $v[\mathbf{x}]$ is coded for all currently insignificant coefficients not processed in the NZN pass; a tarp filter is used to produce a probability estimate that drives a non-adaptive arithmetic coder in this pass.

- *Refinement (RF) Pass*: Refinement bits are coded for currently significant coefficients.

The NZN pass codes $v[\mathbf{x}]$ for coefficients that have a high probability of becoming significant, since their neighboring coefficients are already significant; for this pass, the neighbors are defined to be those coefficients spatially adjacent to the current coefficient as illustrated in Fig. 1(a). Since, in this case, the density of significant coefficients is high, tarp filtering is less effective for these coefficients. On the other hand, coefficients processed in the ZR pass have a lower probability of significance, and the tarp filter is much more effective for them. The third coding pass, the RF pass, is essentially the same as in the original 2D-tarp coder, processing consecutive bitplanes of coefficients already known to be significant from prior coding passes.

In addition to the adoption of fractional bitplane coding, TCE also introduces techniques to improve the accuracy of the probability estimate of the tarp filter within the ZR pass. Briefly, a cross-scale parent-child relationship is used to refine the probability estimates arising in high-resolution subbands based on probabilities previously obtained for tarp filtering in lower-resolution subbands; this process exploits cross-scale correlations among wavelet coefficients that are ignored by the original 2D-tarp coder. Additionally, a so-called “reversed” tarp filter is applied both before and after the ZR pass to incorporate additional information from the NZN pass to update the probability estimate; see [7] for greater detail. For lossy coding, experimental results reported in [7] exhibit a gain of 0.5–0.8 dB in PSNR for 2D-TCE over the original 2D-tarp of [12] and performance essentially comparable to that of JPEG2000.

C. Integer KLT

The typical approach for coding hyperspectral imagery is to couple a 3D extension of an existing 2D coding algorithm with a 1D transform designed to decorrelate the data in the spectral direction. Although DWTs have been commonly used for such spectral transforms, a spectral KLT is known to yield significantly superior results (e.g., [14]). However, the usual KLT is an irreversible transform. For lossy-to-lossless compression—the focus here—what is needed is a reversible KLT that maps integers to integers. Such a reversible integer KLT was proposed in [9, 10] based on reversible matrix factorizations originating in [8].

Specifically, consider an $M \times M$ upper triangular matrix $B = [b_{mn}]$ with ± 1 on the diagonal. From [8], a reversible version of transform $\tilde{\mathbf{y}} = B\mathbf{y}$, can be achieved by lifting steps in the form

$$\tilde{\mathbf{y}}_m = \begin{cases} b_{mm}y_m + \left[\sum_{n=m+1}^M b_{mn}y_n \right]_{\mathbb{Z}}, & m = 1, \dots, M-1, \\ b_{mm}y_m, & m = M, \end{cases} \quad (3)$$

where $[\cdot]_{\mathbb{Z}}$ denotes rounding to the nearest integer. The inverse transform is trivially obtained in the reverse order. Reversible integer lifting steps similar to (3) can be easily derived for the case that B is lower triangular with unit diagonal [8].

In [8], an $M \times M$ transform matrix A with determinant $|A| = \pm 1$ is factored as

$$A = PLUS, \quad (4)$$

where L and S are lower triangular, U is upper triangular, and P is a reversible permutation matrix. Implementing the U transform using (3) and the L and S transforms using its corresponding lower-triangular counterpart results in a reversible calculation of A . Since the KLT is an orthonormal transform (i.e., its determinant is 1), this factorization was used in [9] to obtain a reversible integer transform that approximates the KLT. Here, we employ the variant using quasi-complete pivoting proposed in [10].

III. PROPOSED CODER

The first step in the proposed 3D-TCE coder is a spectral transform that decorrelates the hyperspectral image spectrally. This is followed by a dyadic DWT that decomposes the data spatially, resulting in a subband decomposition of the data cube. For lossless coding, the spectral transform is either a traditional 1D wavelet transform using the popular reversible 5/3 DWT or an integer KLT as described above in Sec. II-C. The former is referred to below as DWT+3D-TCE, the latter as KLT+3D-TCE. The structure of the resulting subbands for DWT+3D-TCE takes the form as illustrated in Fig. 2(a) which is commonly known as the wavelet-packet decomposition (e.g., [11]). On the other hand, when a KLT is used for spectral decorrelation, we consider subbands to be defined so as to include all principal components as illustrated in Fig. 2(b). In either case, the TCE coder is conducted on a subband-by-subband basis as described below.

For a given threshold in TCE, coefficients are partitioned into one of the three passes as described in Sec. II-B—coefficients that are currently insignificant but which have at least one significant neighbor fall into the NZN pass; coefficients that are currently insignificant and have all neighbors being insignificant fall into the ZR pass; while the remaining, significant coefficients are processed in the RF pass. For 3D-TCE, the definition of the neighbors of the current coefficient is extended in the spectral direction to encompass a total of 26 coefficients as illustrated in Fig. 1(b).

The other changes involved in going from 2D to 3D center on the tarp-filter process in the ZR pass. Specifically, 2D tarp filtering is replaced by the 3D tarp filter developed in [13], while the cross-scale parent-child relationship used to refine probability estimates in TCE is suitably modified for the 3D character of the subbands. The 3D parent-child relationship is spatially similar to that of the original 2D-TCE, while there are links in the spectral direction between certain coefficients (i.e., the spatially low-resolution coefficients). These parent-child relations are illustrated in Fig. 3(a) and (b) for DWT+3D-TCE and KLT+3D-TCE, respectively.

Finally, in the KLT+3D-TCE coder, if the maximum coefficient magnitude in a specific principal component resulting from the spectral KLT is less than the value of the current threshold, then coding of that principal component is skipped in the TCE coding of all subbands. This component skipping

helps efficiently code large numbers of insignificant coefficients since relatively few principal components typically contain significant signal energy. Components to be skipped are indicated to the decoder by a list of maximum bitplanes for each subband of each component; the list itself is losslessly coded with DPCM.

IV. EXPERIMENTAL RESULTS

A. Lossless Coding

We performed coding experiments on several hyperspectral image datasets. We used four popular AVIRIS radiance datasets of size 512×512 with 224 bands, specifically Scene 1 of the “Cuprite,” “Jasper Ridge,” “Lunar Lake,” and “Moffett” radiance datasets² cropped spatially to size 512×512 from the upper left corner. We also used a TRWISIII dataset (“DREC”) of spatial size 256×256 with 384 spectral bands as well as a CASI dataset of spatial size 150×250 with 72 spectral bands.

Table I compares the performance at lossless compression for the proposed 3D-TCE algorithm against that of several other state-of-the-art lossy-to-lossless coders. Specifically, Table I tabulates the bitrate in bits per pixel per band (bpppb) of the lossless bitstream produced by the various coders. We investigate the performance of 3D-tarp [13], JPEG2000 [5, 6] and the proposed 3D-TCE algorithm using for spectral decorrelation both a DWT (DWT+3D-tarp, DWT+JP2K, and DWT+3D-TCE, respectively) as well as a KLT (KLT+3D-tarp, KLT+JP2K, and KLT+3D-TCE, respectively). In the case of the spectral DWT, five levels of the reversible, integer-valued 5/3 DWT is applied; for the spectral KLT, the reversible, integer-valued KLT factorization of [10] is applied. All techniques use a 5-level reversible 5/3 spatial DWT. Implementations of 3D-tarp and 3D-TCE are from QccPack³ [15]; Kakadu⁴ Version 5.1 is used for the JPEG2000 results. Bitrate figures for all KLT-based coders include the overhead of the KLT transform matrix. We see from Table I, that, for the same spectral transform, the proposed 3D-TCE coder always outperforms the 3D-tarp coder of [13] from which it originates. Additionally, 3D-TCE outperforms JPEG2000 for both the wavelet-based as well as KLT-based spectral transforms. For example, the KLT+3D-TCE coder averages 4.76 bpppb on the set of AVIRIS images, an improvement of 3.6% over JPEG2000 with the same spectral KLT.

We also include in Table I results for these datasets for two state-of-the-art techniques that are strictly lossless coders not capable of lossy-to-lossless operation, namely, M-CALIC [2] and LUT [4]. Of particular note is that, out of the set of lossy-to-lossless and strictly lossless coders, KLT+3D-TCE obtains the best compression for half of the datasets, being second to LUT for two of the AVIRIS datasets and to M-CALIC for the TRWIS dataset.

B. Lossy Coding

We present results for lossy coding for the hyperspectral datasets and coders under consideration using both traditional,

²<http://aviris.jpl.nasa.gov/>

³<http://qccpack.sourceforge.net/>

⁴<http://www.kakadusoftware.com/>

irreversible, floating-point transforms as well as the reversible, integer-value transforms that enable lossy-to-lossless operation. First, in Table II, we have lossy coding with irreversible transforms—all DWTs use 5 levels of the popular irreversible 9/7 biorthogonal transform, while the spectral KLT is a traditional irreversible floating-point implementation. We measure performance in terms of signal-to-noise ratio (SNR)⁵, average spectral angle mapper (SAM), and preservation of classification (POC). SAM is the angle between original and reconstructed hyperspectral pixel vectors as averaged over the dataset. POC [11] results from the application of unsupervised classification⁶ on the original hyperspectral image as well as on the reconstructed image; specifically, POC is the number of pixels that do not change assigned class as a result of the compression, expressed as a percentage. Table II indicates that KLT+3D-TCE and KLT+JP2K obtain virtually identical performance—in terms of SNR, average SAM, and POC—for all the datasets at a fixed bitrate.

Table III duplicates the results of Table II only this time using the coders with reversible transforms as in Sec. IV-A; in this case, the lossless bitstreams are truncated to achieve a lossy representation at a rate of 1.0 bpppb. Again, we see that KLT+3D-TCE achieves performance very close to that of KLT+JP2K for all three measures.

V. CONCLUSIONS

In this paper, we proposed an algorithm for the lossy-to-lossless compression of hyperspectral imagery that couples a KLT for spectral decorrelation with a 3D extension of TCE, an image coder originally developed for lossy coding of 2D images. When used with a reversible, integer KLT and reversible wavelet transforms, the resulting coder, 3D-TCE, achieves lossy-to-lossless operation while outperforming the widely-used JPEG2000 standard at the lossless compression of several hyperspectral images. To the best of our knowledge, the results presented here—on the order of 3% better compression than JPEG2000 with the same integer KLT—represent the best lossy-to-lossless coding results for hyperspectral imagery to appear in the literature. Although generally acknowledged that the lossy-to-lossless capability comes at a cost to lossless compression performance, the 3D-TCE coder that we developed here offers performance closer to that of purely lossless techniques as compared to other lossy-to-lossless coders, and, in fact, outperforms the prominent lossless LUT and M-CALIC coders for several datasets considered. On the other hand, in terms of lossy compression, rate-distortion performance of 3D-TCE is very close to that of the state-of-the-art JPEG2000.

For lossy compression, it is well known that the calculation of the data covariance matrix as needed to create the KLT transform matrix represents the majority of computation associated with techniques employing KLT spectral decorrelation such as those we consider here; see [16, 17] for example. For lossless compression, factorization and nonlinear transform application are added to this KLT-training to produce a

reversible transform. In our empirical observations, the three stages of this reversible transform dominate the computation within the lossy-to-lossless coders considered here such that our implementations of KLT+3D-tarp, KLT+3D-TCE, and KLT+JP2K have roughly equal execution times. The M-CALIC implementation we use is about 5 times faster, while LUT, on the other hand, is an order of magnitude faster than M-CALIC. LUT would therefore most likely be preferred in applications strictly needing only lossless compression. If, however, lossy-to-lossless operation is desired, the proposed KLT+3D-TCE coder offers not only lossy coding to match that of JPEG2000 but also lossless coding that rivals state-of-the-art purely lossless coders such as M-CALIC and LUT.

REFERENCES

- [1] X. Wu and N. Memon, "Context-based lossless interband compression—Extending CALIC," *IEEE Transactions on Image Processing*, vol. 9, no. 6, pp. 994–1001, June 2000.
- [2] E. Magli, G. Olmo, and E. Quacchio, "Optimized onboard lossless and near-lossless compression of hyperspectral data using CALIC," *IEEE Geoscience and Remote Sensing Letters*, vol. 1, no. 1, pp. 21–25, January 2004.
- [3] J. Mielikainen and P. Toivanen, "Clustered DPCM for the lossless compression of hyperspectral images," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 41, no. 12, pp. 2943–2946, December 2003.
- [4] J. Mielikainen, "Lossless compression of hyperspectral images using lookup tables," *IEEE Signal Processing Letters*, vol. 13, no. 3, pp. 157–160, March 2006.
- [5] *Information Technology—JPEG 2000 Image Coding System—Part 1: Core Coding System*, ISO/IEC 15444-1, 2000.
- [6] *Information Technology—JPEG 2000 Image Coding System—Part 2: Extensions*, ISO/IEC 15444-2, 2004.
- [7] C. Tian and S. S. Hemami, "An embedded image coding system based on tarp filter with classification," in *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing*, vol. 3, Montreal, Canada, May 2004, pp. 49–52.
- [8] P. Hao and Q. Shi, "Matrix factorization for reversible integer mapping," *IEEE Transactions on Signal Processing*, vol. 49, no. 10, pp. 2314–2324, October 2001.
- [9] —, "Reversible integer KLT for progressive-to-lossless compression of multiple component images," in *Proceedings of the International Conference on Image Processing*, vol. 1, Barcelona, Spain, September 2003, pp. 633–636.
- [10] L. Galli and S. Salzo, "Lossless hyperspectral compression using KLT," in *Proceedings of the International Geoscience and Remote Sensing Symposium*, vol. 1, Anchorage, AK, September 2004, pp. 313–316.
- [11] J. E. Fowler and J. T. Rucker, "3D wavelet-based compression of hyperspectral imagery," in *Hyperspectral Data Exploitation: Theory and Applications*, C.-I. Chang, Ed. Hoboken, NJ: John Wiley & Sons, Inc., 2007, ch. 14, pp. 379–407.
- [12] P. Simard, D. Steinkraus, and H. Malvar, "On-line adaptation in image coding with a 2-D tarp filter," in *Proceedings of the IEEE Data Compression Conference*, J. A. Storer and M. Cohn, Eds., Snowbird, UT, April 2002, pp. 23–32.
- [13] Y. Wang, J. T. Rucker, and J. E. Fowler, "3D tarp coding for the compression of hyperspectral images," *IEEE Geoscience and Remote Sensing Letters*, vol. 1, no. 2, pp. 136–140, April 2004.
- [14] Q. Du and J. E. Fowler, "Hyperspectral image compression using JPEG2000 and principal component analysis," *IEEE Geoscience and Remote Sensing Letters*, vol. 4, no. 2, pp. 201–205, April 2007.
- [15] J. E. Fowler, "QccPack: An open-source software library for quantization, compression, and coding," in *Applications of Digital Image Processing XXIII*, A. G. Tescher, Ed. San Diego, CA: Proc. SPIE 4115, August 2000, pp. 294–301.
- [16] B. Penna, T. Tillo, E. Magli, and G. Olmo, "A new low complexity KLT for lossy hyperspectral data compression," in *Proceedings of the International Geoscience and Remote Sensing Symposium*, vol. 7, Denver, CO, August 2006, pp. 3525–3528.
- [17] Q. Du and J. E. Fowler, "Low-complexity principal component analysis for hyperspectral image compression," *International Journal of High Performance Computing Applications*, 2008, to appear.

⁵log ratio of signal variance to mean squared error

⁶ENVI Ver. 4.3 implementation of k -means with a maximum of 10 classes and a change threshold of 5.00

TABLE I
RATE IN BPPPB FOR LOSSLESS COMPRESSION

Algorithm	AVIRIS					TRWIS	CASI
	Cuprite	Jasper Ridge	Lunar Lake	Moffett	Average	DREC	CASI
DWT+3D-tarp	5.58	5.88	5.60	5.95	5.75	7.02	9.70
DWT+JP2K	5.28	5.54	5.30	5.65	5.45	6.68	9.47
DWT+3D-TCE	5.18	5.44	5.19	5.52	5.33	6.53	9.12
KLT+3D-tarp	5.13	5.11	5.24	5.22	5.17	6.71	9.71
KLT+JP2K	4.88	4.87	4.98	4.97	4.92	6.42	9.43
KLT+3D-TCE	4.72	4.71	4.82	4.80	4.76	6.22	9.01
M-CALIC	4.86	4.96	4.96	5.05	4.96	6.13	n/a*
LUT	4.61	4.92	4.77	5.13	4.86	7.19	10.23

*Result not available due to the dynamic range of the dataset exceeding that supported by the M-CALIC implementation.

TABLE II
PERFORMANCE FOR LOSSY COMPRESSION AT 1.0 BPPPB USING
IRREVERSIBLE TRANSFORMS

	SNR (dB), Average SAM (degrees), and POC (%)					
	DWT+ 3D-tarp	DWT+ JP2K	DWT+ 3D-TCE	KLT+ 3D-tarp	KLT+ JP2K	KLT+ 3D-TCE
Cuprite	50.43 0.12° 99.7%	51.00 0.12° 99.8%	50.80 0.12° 99.8%	53.62 0.09° 99.7%	54.13 0.08° 99.8%	54.08 0.08° 99.7%
Jasper Ridge	43.68 0.29° 99.3%	44.85 0.26° 99.1%	44.65 0.26° 99.4%	49.64 0.15° 99.6%	50.33 0.14° 99.8%	50.33 0.14° 99.6%
Lunar Lake	52.17 0.10° 99.6%	52.73 0.10° 99.5%	52.53 0.10° 99.4%	54.43 0.08° 99.5%	55.21 0.07° 99.6%	55.22 0.07° 99.7%
Moffett	44.60 0.25° 99.8%	45.49 0.22° 99.7%	45.43 0.22° 99.8%	50.21 0.13° 99.9%	50.92 0.12° 99.8%	50.97 0.12° 99.9%
DREC	42.17 0.40° 99.0%	42.98 0.36° 99.5%	42.75 0.37° 99.5%	44.50 0.30° 99.3%	45.21 0.28° 99.2%	45.06 0.28° 99.2%
CASI	33.35 0.62° 99.3%	33.76 0.59° 99.2%	33.73 0.60° 99.3%	34.63 0.54° 99.4%	34.84 0.52° 99.2%	34.92 0.52° 99.4%

TABLE III
PERFORMANCE FOR LOSSY COMPRESSION AT 1.0 BPPPB USING
REVERSIBLE TRANSFORMS

	SNR (dB), Average SAM (degrees), and POC (%)					
	DWT+ 3D-tarp	DWT+ JP2K	DWT+ 3D-TCE	KLT+ 3D-tarp	KLT+ JP2K	KLT+ 3D-TCE
Cuprite	49.26 0.14 99.7%	50.03 0.13 99.5%	49.70 0.13 99.7%	51.55 0.11 99.7%	53.05 0.09 99.8%	52.68 0.10 99.7%
Jasper Ridge	42.85 0.32 98.3%	44.09 0.28 99.0%	43.56 0.29 97.9%	47.86 0.18 99.5%	49.22 0.16 98.3%	48.89 0.16 99.3%
Lunar Lake	50.99 0.12 99.6%	51.76 0.11 99.6%	51.36 0.11 99.6%	52.65 0.10 99.3%	54.21 0.08 99.4%	54.08 0.08 99.5%
Moffett	43.78 0.27 99.5%	44.74 0.24 99.5%	44.35 0.25 99.6%	48.52 0.16 99.7%	49.88 0.14 99.6%	49.77 0.14 99.8%
DREC	40.85 0.46 99.0%	42.30 0.39 98.9%	41.88 0.41 98.9%	43.76 0.33 99.6%	44.65 0.30 99.8%	44.13 0.31 99.6%
CASI	32.76 0.65 99.2%	33.35 0.61 98.9%	33.12 0.63 99.3%	34.19 0.56 99.4%	34.28 0.56 99.4%	34.33 0.55 99.4%

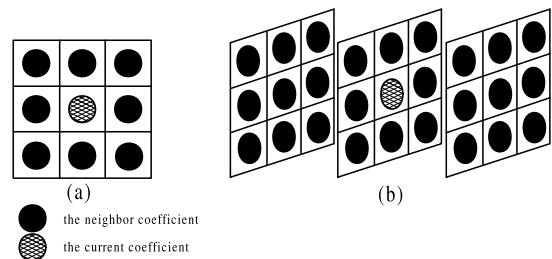


Fig. 1. Neighbors of the current coefficient (a) in 2D, (b) in 3D.

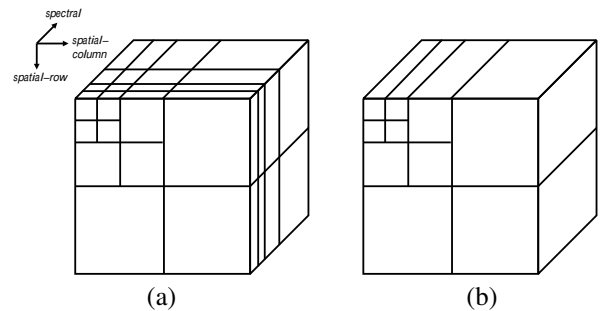


Fig. 2. Subband arrangement for (a) DWT spectral transform, (b) KLT spectral transform

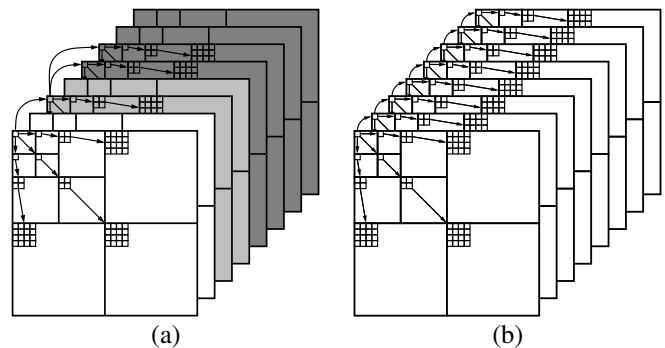


Fig. 3. Parent-child relationship for (a) DWT+3D-TCE, spectral subbands are indicated by different shades of gray; (b) KLT+3D-TCE.