EXTENSIONS TO THE CCSDS-IDC RECOMMENDATION FOR ON-BOARD HYPERSPECTRAL IMAGE CODING

Ian Blanes, Fernando García-Vílchez, and Joan Serra-Sagristà

Dept. of Information and Communications Engineering
Universitat Autònoma de Barcelona
Edifici Q, UAB, E-08193 Cerdanyola del Valles (Barcelona), Spain
Email: {ian.blanes,joan.serra}@uab.cat, fgarcia@deic.uab.cat

Competitive payload data compression methods are required due to advances in remote-sensing sensors that allow the acquisition of huge volumes of data, and due to bandwidth restrictions of downlink channels between space-borne sensors and ground stations. Particularly, for space-borne hyperspectral sensors, on-board coding methods exploit the existing redundancy (both in the spectral and in the spatial domain), and help to achieve longer capture periods or lower bandwidth requirements. In addition, as space-borne hyperspectral sensors are usually pushbroom sensors, vertical streaking artifacts are likely to appear. These artifacts should be taken into account, as they may have a significant impact on coding performance.

This paper describes an approach for progressive-lossy-to-lossless on-board data compression, which is based on a combination of two modern proposals: the recent Pairwise Orthogonal Transform, a low-cost transform based on the composition of Karhunen-Loêve transforms, and TER, an extension of the CCSDS-IDC recommendation providing advantageous functionalities.

Special attention is paid to the operation on fixed-height strips required by memory-constrained environments, and to streaking artifacts produced by pushbroom sensors, which are reduced by a pre-processing stage.

Experimental results on uncalibrated imagery from the Hyperion Sensor of the EO-1 program are provided. On the lossy regime, a moderate performance penalty exists on MSE results when compared with other coders that do not operate under on-board constraints, but such penalty is not translated to PAE results, which are competitive. In addition, lossless performance is still competitive if compared to other lossless methods that do not provide progressive quality refinement.

INTRODUCTION

Hyperspectral image coding for on-board sensors is attracting interest by numerous space agencies and researchers because of the growing use of these acquisition devices. As it is known, hyperspectral images exhibit high redundancy among their components (or bands), and significant coding gains may be achieved by coding techniques able to exploit this redundancy [1, 2, 3]. However, not all coding techniques exploiting spectral redundancy can be used on-board an aircraft or a spacecraft because of the restrictions and limitations in such environments.

On an on-board environment, coding techniques should fulfill several requirements: both their memory requisites and their computational cost have to be low, as they might have to operate on radiation hardened hardware with little power available. Also of concern are the bandwidth restrictions of down-link channels between space-borne sensors and ground stations: in this case, a coding technique might be required to operate on lossy mode, which provides a good representation of the captured image while keeping the consumption of bandwidth resources under a certain threshold.

It seems therefore natural to consider coding techniques capable of operating in Progressive-Lossy-to-Lossless (PLL) mode. A PLL coding technique may supply both lossy and lossless versions of an image, where the information transmitted for a lossy version is later employed and extended to achieve a lossless representation of that image.

In the particular case of space-borne sensors, as the most common type of sensor is a pushbroom sensor, vertical streaking artifacts are likely to appear. These artifacts should also be taken into account by coding techniques, as they may have a significant impact on coding performance.

In this paper we propose a coding approach for on-board space sensors that deals with the requirements and restrictions referred above, i.e., it is able to operate with restricted memory capabilities, with low computational cost, providing PLL compression, and addressing the streaking artifacts. The coding approach combines three existing techniques: a pre-processing method [4] is applied to account for the streaking artifacts, a fast and inexpensive spectral transform with moderate coding performance —the POT [4]— is then performed before encoding the image with TER [5], an extension of the CCSDS-IDC recommendation, employed in fixed-height strips to limit memory requirements. Extensive experimental results are provided for Hyperion data, indicating that the combination of these three existing techniques produces reasonable results considering the constraints imposed for on-board operation.

This paper is organized as follows. First, the three techniques that have been combined in the proposed coding approach are introduced: the pre-processing stage, the spectral transform POT, and the coding technique TER. Then, detailed experimental results are provided for both lossy and lossless mode. Finally, some conclusions are drawn.

PRE-PROCESSING

Pushbroom sensors are common on spacecrafts due to their lack of moving parts. As these sensors capture all the spectral components of an image line at once by a bidimensional sensor array, some vertical streaking artifacts might appear due to small variations between individual sensors. After images have been downloaded from a spacecraft, these artifacts are removed on ground stations with a calibration process. When performing on-board image coding, these artifacts are still there, and penalize coding performance because they disturb the detection of redundancy in adjacent locations.

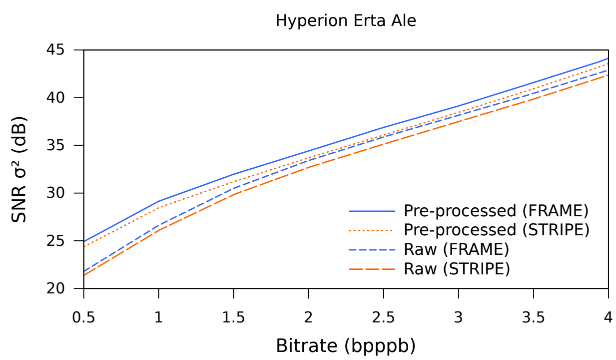


Fig. 1: Comparison of coding performance of CCSDS-IDC with and without a pre-processing stage.

In [4], a very simple pre-processing stage is introduced to limit the impact of these artifacts on uncalibrated images. The pre-processing is a lossless process based on shifting the values of image columns to smooth differences with their neighbors, and can be undone after decompression just as if it were an additional transform. First, shifting offsets are obtained by

$$\delta_{i,k} = \frac{1}{32}\sum_{j=0}^{31} \operatorname*{median}_{|m|\leq 4}\left(I_{w(i+m,0,x-1),j.k}\right), \tag{1}$$

where $I_{i,j,k}$ is the value of original image at band k, row j, column i, and x is the image width. Only the first 32 lines are used to obtain shifting offsets. Afterwards, each image location, as it is retrieved, is modified according to the previous offset by

$$\hat{I}_{i,j,k} = w(I_{i,j,k} - [\delta_{i,k}], -2^{15}, 2^{15}-1) \tag{2}$$

The function w translates any integer value to the interval $[a,b]$ with symmetric reflection around a if $i<a$ or around b if $i>b$, i.e.,

$$w(i,a,b) = b - |b-a-(i-a \bmod 2(b-a))|. \tag{3}$$

A comparison of the effects of a pre-processing stage is reported in Fig. 1, when coding an image from the pushbroom sensor Hyperion using CCSDS-IDC. For the pre-processed image, CCSDS-IDC obtains coding gains between 1 and 3 dB.

Note that the described procedure does not preserve the Parseval's identity, and therefore introduces noise to the distortion estimation of a Rate-Distortion (R-D) optimization; however, as seen in Fig. 1, the effects of such distortion noise are much smaller than the gain obtained by removing the streaking artifacts.

POT

The Pairwise Orthogonal Transform (POT) [4] is a spectral transform derived from the Karhunen-Loêve Transform (KLT) using a divide-and-conquer technique [6, 7]. The main idea is to replace a full KLT by a sequence of KLTs applied to two components each, that has much lower computational cost and side information requirements. Computational cost for the POT is very similar to the cost of wavelets; it is slightly higher than a CDF 5/3 IWT, but lower than a CDF 9/7, while the coding performance is between that of wavelets and the performance of the KLT. Fig. 2 reports a comparison of the performance of the three spectral transforms (wavelets, KLT, and POT) for PLL mode. Of particular interest is the performance penalty that the KLT has in higher bitrates due to the large lifting network required for its lossless application.

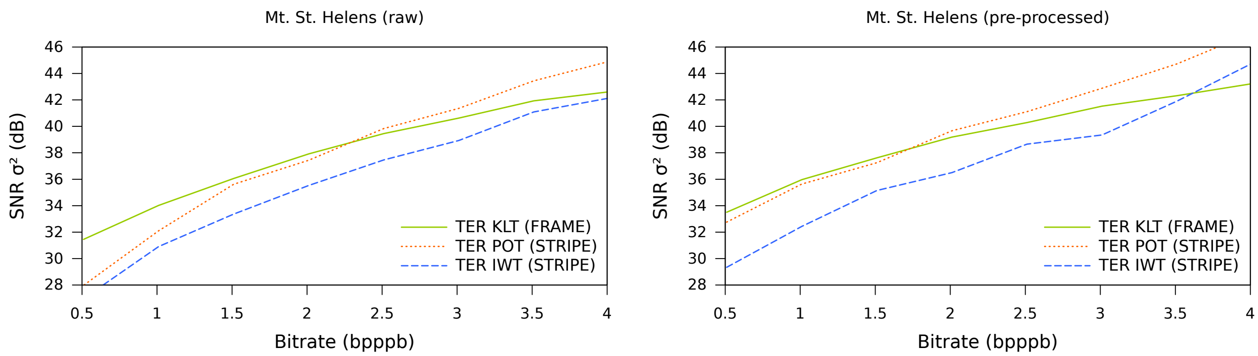
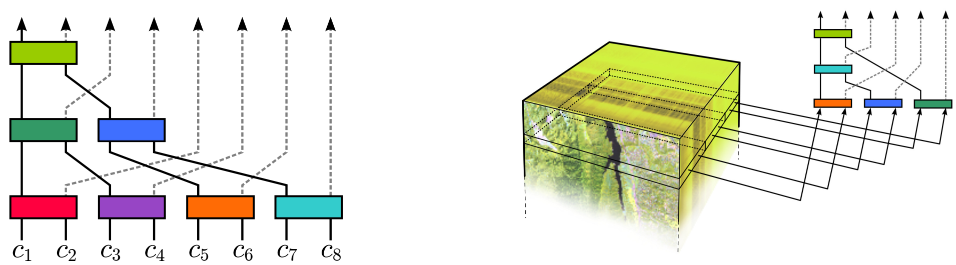


Fig. 2: Coding performance of TER with POT.

The sequence of two-component KLTs is organized in a multi-level structure as follows: on the first level, all spectral components of a hyperspectral image are processed by consecutive two-component KLTs, where the principal output (of each two-component transform) is forwarded to a next level, and no further processing is applied to the other output. On successive levels, the same procedure is applied, until only one two-component KLT is applied on the last level. An example of such a structure for a hyperspectral image of eight components is shown in Fig. 3a.



(a) Example structure (b) Line-based application

Fig. 3: a) The application of a POT in a line-by-line basis for an eight-component image. Each colored rectangle corresponds to a two-component KLT. b) Line-based application of a POT.

A line-base application of the transform (see Fig. 3b) is feasible due to the small amount of side information it requires, and thus, the memory requirements of the POT are very small. That the POT has such small amount of side information is caused, in part, by the use of only two-component KLTs, which simplifies the Eigenvalue Decomposition (ED) procedure with a simple, non-iterative, and numerically stable solution. The lossy operation in each of the two-component KLTs is defined by the following matrix operation:

$$\begin{pmatrix} y_1 \\ y_2 \end{pmatrix} = \begin{pmatrix} p & q \\ t & u \end{pmatrix}^T \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}, \tag{4}$$

with

$$p = u = \sqrt{1-t^2}, t = -q = \frac{b}{|b|}\sqrt{\frac{1}{2} - \frac{(a-d)}{2s}}, s = \sqrt{(a-d)^2 + 4b^2} \tag{5}$$

and

$$\Sigma_{\mathrm{X}} = \begin{pmatrix} a & b \\ b & d \end{pmatrix}, \tag{6}$$

where Σ_{X} is the spectral covariance matrix of the input. Of all the parameters in these operations, only t is needed to invert the transform, as the other three parameters can be directly recovered from it. This fact also enables the use of some quantization on parameter t, in order to reduce the side information, before the other parameters are computed.

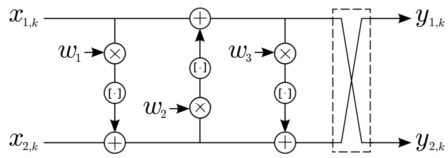


Fig. 4: Lifting structure for a two-component KLT. Conditional permutation included.

As the required operation of the transform has to be lossless so that a PLL encoding can be produced, a lifting application of the transform can be used. Fig. 4 shows a lifting structure associated with Eq. 4 that requires only 9 operations and a conditional permutation. Lifting coefficients are determined by Eq. 7:

$$\begin{cases} P = \left(\begin{smallmatrix} 1 & 0 \\ 0 & 1 \end{smallmatrix}\right), w_1 = w_3 = \frac{p-1}{t}, w_2 = t & \text{for } |t| \geq |p|, \\ P = \left(\begin{smallmatrix} 0 & 1 \\ 1 & 0 \end{smallmatrix}\right), w_1 = w_3 = \frac{1-t}{p}, w_2 = -p & \text{for } |t| < |p|. \end{cases} \tag{7}$$

CCSDS-IDC / TER

The CCSDS-IDC recommendation [8] is addressed to produce both lossless and lossy monoband compression. CCSDS-IDC recommendation was designed for use in rockets, satellites or spacecrafts, thus, it satisfies all the memory and computation restrictions of this kind of equipments [9].

The encoding process of CCSDS-IDC is divided in two functional parts. First, a three-level 2D (spatial) Discrete Wavelet Transform (DWT) is performed in order to decorrelate the input data. Then, the transformed data is rearranged in blocks, which are grouped in segments. Finally, each segment is independently encoded by the Bit Plane Encoder (BPE) stage. The number of blocks in a segment is usually selected using two criteria: the frame and the strip mode. The frame mode consists of encoding all the blocks of the image in one single segment. The strip mode consists of selecting the number of blocks in a segment as the available blocks in a row, thus, this mode is especially suitable for remote sensing scenarios, where memory load and computational complexity are highly restricted.

Table 1: Feature comparison between the CCSDS-IDC and TER.

	IDC	TER
Spatial DWT levels	3	any
Spectral transform	none	none/DWT/KLT/POT/...
Stream scalability	L	L, P, R, and C
Interactive decoding	no	yes
Interactive transmission	no	yes

The introduction of TER [5] provided a set of advantageous functionalities, like scalable encoded streams, or extension to multi-, hyper- or ultraspectral data, while also improving coding efficiency. The main similarities and differences between the CCSDS-IDC Recommendation and TER are summarized in Table 1. The improvement in coding performance stems from the extension in spatial DWT decomposition levels, and the possibility to encode multiple components together for spectral images. A multicomponent bit-plane scanning order paired with a spectral transform improves compression performance, as opposed to the CCSDS-IDC recommendation, for which, since it is restricted to monoband compression, if multiple components are to be encoded together, no image-dependent bit-rate distribution can be performed. Also, thanks to the proposed extensions, the encoded stream can be ordered by: quality (L), position (P), resolution (R), and component (C), obtaining, among others, the following progression orders: LRCP, RLCP, RPCL, PCRL, and CPRL. Embedded headers permit the re-organization of the code-stream from one progression order to any other, without needing to decode the image. Note that neither the computational complexity nor the memory load are increased with the new features –except for the spatial DWT extension–, thus preserving the requirements set out for the CCSDS-IDC Recommendation.

A comparison of coding performance between CCSDS-IDC and TER is provided in Fig. 5 for the Lake Monona image, both for frame and strip mode, and when considering the pre-processing stage to reduce the influence of the vertical streaking artifacts.

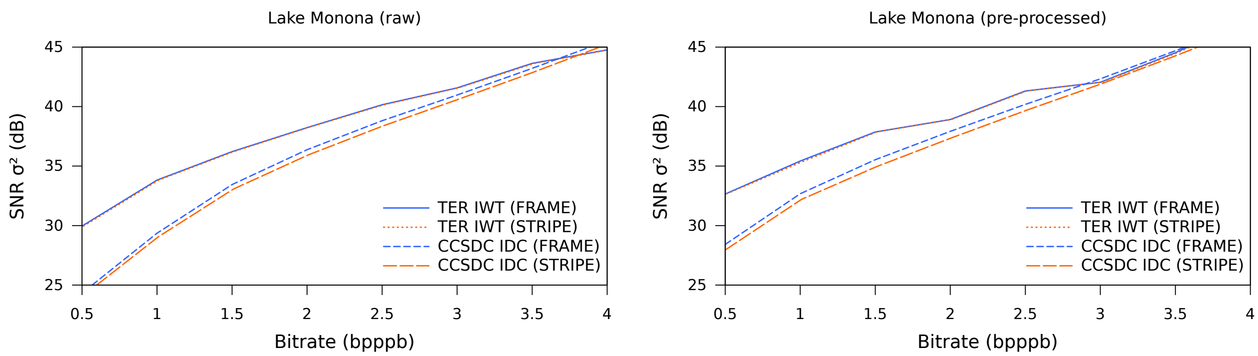


Fig. 5: Coding performance for CCSDS-IDC vs TER (with IWT as spectral transform) on raw and preprocessed.

EXPERIMENTAL RESULTS

Experimental results have been conducted on a corpus of raw Hyperion images, which have not undergone any calibration process. The Hyperion instrument is a pushbroom space-borne sensor on board the Earth Observing 1 mission, which, as other pushbroom sensors, produces vertical streaking artifacts. Hyperion images have a fixed width of 256 columns, a variable height, and 242 spectral components covering wavelengths from 357 to 2576 nm at a sample resolution of 12 bits per pixel per band (bpppb). Table 2 provides full technical names and sizes for the images used. The described images can be downloaded from [10].

Table 2: Technical Names for the raw Hyperion images used.

Image Name	Technical Name	Size ($x \times y \times z$)
Erta Ale	EO1H1680502010057110KF	$256 \times 3187 \times 242$
Lake Monona	EO1H0240302009166110PF	$256 \times 3176 \times 242$
Mt. St. Helens	EO1H0460282009231110KF	$256 \times 3242 \times 242$

The proposed coding approach is compared against JPEG2000 [11], a full coding system without the strong constraints of an on-board system. Software versions used are TER Software [12], as the improved version of the CCSDS-IDC recommendation, and the fast POT implementation [13]; both of them are available as open-source. The JPEG2000 software used is Kakadu [14].

In order to appreciate the performance penalty with respect to such full coding process, memory requirements have to be taken into account. The theoretical peak usage of each stage of the proposal can be approximately measured by counting the equivalent number of lines required to be in memory for each method. These quantities are 32, 1 and 62 lines, respectively for the pre-processing, the POT, and TER (30 lines for a five-level local wavelet transform, and 32 for a full horizontal stripe to be coded by the bitplane encoder), although these quantities are required at different times. Hence, the peak memory usage for the combination of all techniques is in the order of 10 Mb of memory. In comparison, a coder that operates on frame mode requires around 376 Mb of memory.

Fig. 6 reports the progressive-lossy-to-lossless coding performance of TER and JPEG2000 —measured in Signal-to-

Noise Ratio (SNR) [1]— with and without applying the pre-processing stage to account for the vertical streaking artifacts, which seems to be always beneficial, in particular in combination with the POT. Also of interest is the decay of the KLT performance at high bit-rates, due to the large number of steps of its lifting decomposition. In comparison with JPEG2000, TER suffers a moderate performance penalty because of its operating restrictions. Fig. 7 provides a comparison of TER and JPEG2000 when considering the Peak Absolute Error (PAE) measure. As seen, TER with POT yields very competitive results. Numerical results provided in Table 3 are in line with the previous discussion, showing an improvement between 1 and 3 dB by using a pre-processing stage, and an improvement of more than 2 dB by using the POT over an IWT in pre-processed data. PAE results have greater variability, but are also consistent with the previous analysis.

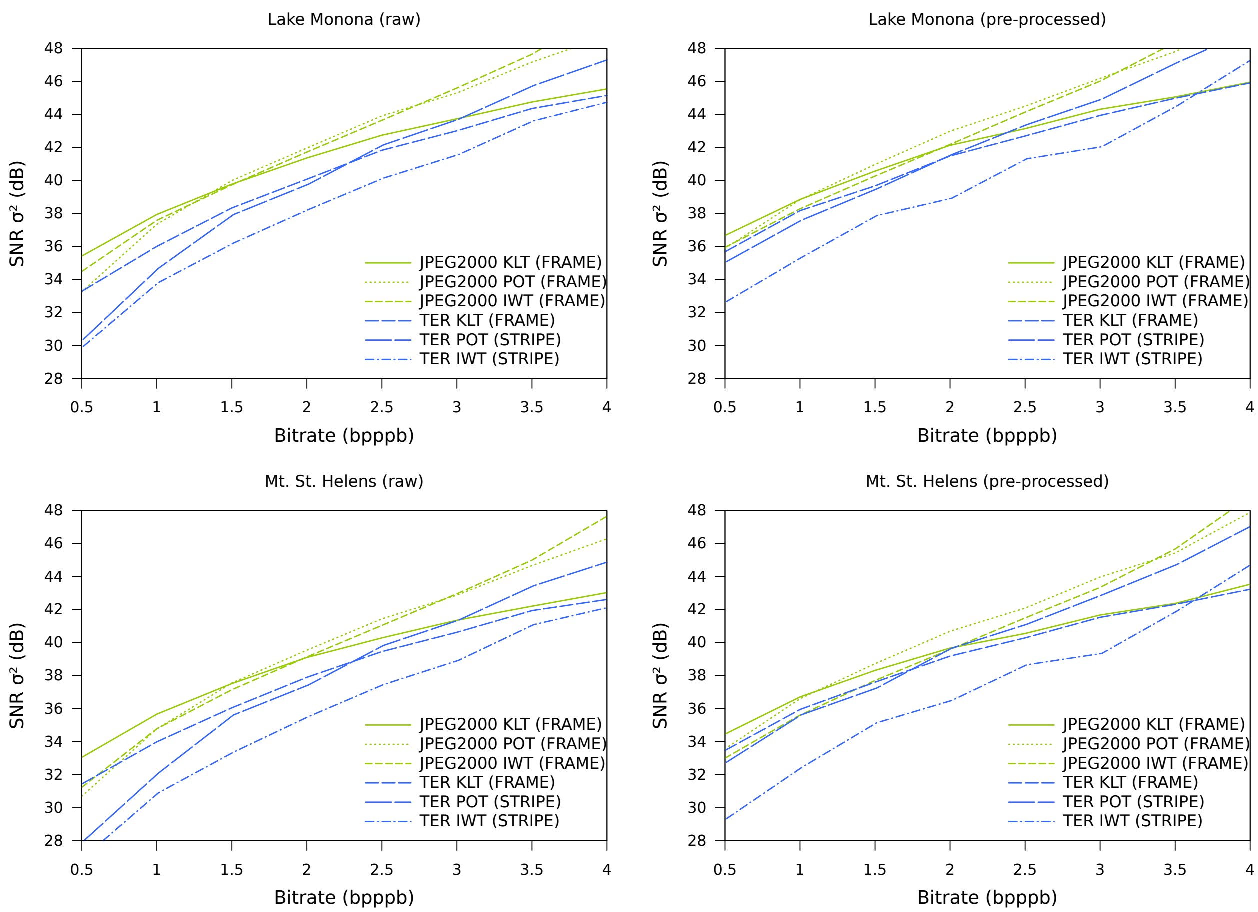


Fig. 6: Coding performance of TER vs JPEG2000 for several spectral transforms, with and without applying a pre-processing stage. SNR results.

Results for the lossless regime are reported in Table 4. TER results significantly improve those of CCSDS-IDC by 0.5 bpppb, yet TER is far from producing the results of JPEG2000 or a pure lossless methods due to its resource constraints. As expected for lossless, no significant performance penalty is observed by the use of the strip mode. The use of a pre-processing stage improves all lossless results, and in particular those of POT, which are otherwise slightly worse than those of IWT.

CONCLUSIONS

This paper contributes a coding approach for on-board data compression that conforms to the resource constraints of on-board environments. The proposal is based on the combination of a pre-processing stage addressing the vertical streaking artifacts common in pushbroom sensors, a low-cost and efficient spectral transform derived from the KLT, and an extension of the CCSDS-IDC recommendation, originally devised for operation in scenarios with memory and

[1] $\mathrm{SNR}\,\sigma^2 = 10\log_{10}(\sigma^2/\mathrm{MSE})$, where σ^2 is the variance of the original image and MSE is the Mean Squared Error.

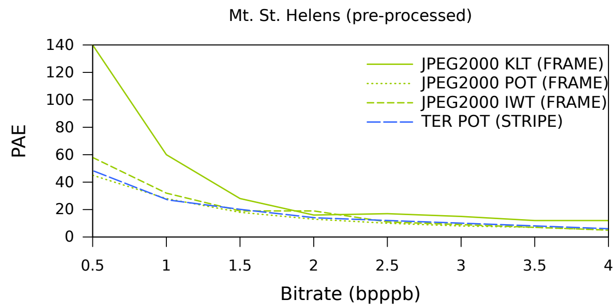


Fig. 7: Coding performance of TER vs JPEG2000 for several spectral transforms, after applying a pre-processing stage. PAE results.

Table 3: Lossy results for the image Erta Ale. PAE in absolute units. SNR-σ^2 in dB.

		0.5 bpppb		1.0 bpppb		2.0 bpppb		3.0 bpppb	
		SNR-σ^2	PAE	SNR-σ^2	PAE	SNR-σ^2	PAE	SNR-σ^2	PAE
Raw	KLT+TER	31.94	109	34.40	67	38.11	18	40.80	13
	POT+TER	28.39	87	32.52	58	37.95	20	41.68	12
	IWT+TER	27.85	77	31.68	48	35.92	25	39.17	15
Pre-processed	KLT+TER	33.41	102	35.96	45	39.25	16	41.63	12
	POT+TER	32.81	52	35.67	27	39.75	14	42.93	10
	IWT+TER	30.21	79	33.31	39	37.08	21	40.42	13

Table 4: Lossless coding results (in bits per pixel per band) for CCSDS-IDC, TER extension, and JPEG2000. Fast Lossless (FL) [15] results provided by A. Kiely (private communication). Results for pre-processed FL correspond in reality to FL-pushbroom, whose pre-processing is different.

			CCSDS-IDC	TER IWT	TER POT	TER KLT	JPEG2000 IWT	JPEG2000 POT	JPEG2000 KLT	FL
Raw	Strip	Erta Ale	5.56	4.93	4.94	4.80	-	-	-	4.74
		Lake Monona	5.56	5.06	5.13	4.95	-	-	-	4.82
		Mt. St. Helens	5.73	5.03	5.07	4.85	-	-	-	4.84
	Frame	Erta Ale	5.54	4.91	4.93	4.79	4.47	4.50	4.44	-
		Lake Monona	5.54	5.05	5.12	4.94	4.59	4.69	4.58	-
		Mt. St. Helens	5.71	5.01	5.06	4.84	4.59	4.65	4.50	-
Pre-processed	Strip	Erta Ale	5.24	4.64	4.60	4.53	-	-	-	"4.39"
		Lake Monona	5.21	4.72	4.73	4.62	-	-	-	"4.52"
		Mt. St. Helens	5.40	4.70	4.65	4.52	-	-	-	"4.43"
	Frame	Erta Ale	5.22	4.62	4.58	4.51	4.34	4.29	4.26	-
		Lake Monona	5.19	4.71	4.72	4.60	4.44	4.45	4.37	-
		Mt. St. Helens	5.39	4.69	4.63	4.50	4.44	4.37	4.27	-

computation restrictions. The introduced approach provides competitive progressive-lossy-to-lossless coding results for both SNR and PAE measures, and also notable results for lossless, when evaluated on Hyperion sensor data.

ACKNOWLEDGMENTS

The authors would like to thank the anonymous reviewers for their valuable comments and the National Aeronautics and Space Administration, the U.S. Geological Survey, and, particularly, A. Kiely for providing the imagery used. This work was supported in part by the Spanish Government, by the Catalan Government, and by FEDER under grants TIN2009-14426-C02-01, TIN2009-05737-E/TIN, SGR2009-1224, and FPU2008. Computational resources used in this work were partially provided by the Oliba Project of the Universitat Autònoma de Barcelona.

REFERENCES

[1] B. Penna, T. Tillo, E. Magli, and G. Olmo, “Transform coding techniques for lossy hyperspectral data compression,” *IEEE Trans. Geosci. Remote Sens.*, vol. 45, no. 5, pp. 1408–1421, May 2007.

[2] J. E. Fowler and J. T. Rucker, *Hyperspectral Data Exploitation: Theory and Applications*. Hoboken, NJ, USA: John Wiley & Sons Inc., 2007, ch. “3D wavelet-Based Compression of Hyperspectral Imager”, pp. 379–407.

[3] J. Zhang, J. E. Fowler, and G. Liu, “Lossy-to-lossless compression of hyperspectral imagery using three-dimensional TCE and an integer KLT,” *IEEE Geosci. Remote Sens. Lett.*, vol. 5, pp. 814–818, Oct. 2008.

[4] I. Blanes and J. Serra-Sagristà, “Pairwise orthogonal transform for spectral image coding,” *IEEE Trans. Geosci. Remote Sens.*, 2010, in press.

[5] F. Garcia-Vilchez and J. Serra-Sagristà, “Extending the CCSDS Recommendation for Image Data Compression for Remote Sensing Scenarios,” *IEEE Trans. Geosci. Remote Sens.*, vol. 47, no. 10, pp. 3431 –3445, 2009.

[6] I. Blanes and J. Serra-Sagristà, “Cost and scalability improvements to the Karhunen-Loêve transform for remote-sensing image coding,” *IEEE Trans. Geosci. Remote Sens.*, vol. 48, no. 7, pp. 2854–2863, Jul. 2010.

[7] I. Blanes, J. Serra-Sagristà, and P. Schelkens, *Recent Advances in Satellite Data Compression*. Springer-Verlag, 2011, ch. Divide-and-conquer decorrelation for hyperspectral data compression.

[8] Consulative Committee for Space Data Systems (CCSDS), *Image Data Compression CCSDS 122.0-B-1*, ser. Blue Book. CCSDS, Nov. 2005. [Online]. Available: http://public.ccsds.org/publications/archive/122x0b1.pdf

[9] Consultative Committee for Space Data Systems (CCSDS), *Image Data Compression CCSDS 120.1-G-1*, ser. Green Book. CCSDS, Jun. 2007. [Online]. Available: http://public.ccsds.org/publications/archive/120x1g1e1.pdf

[10] Jet Propulsion Laboratory, NASA, “Hyperspectral image compression website,” http://compression.jpl.nasa.gov/hyperspectral/.

[11] D. Taubman and M. Marcellin, *JPEG2000: Image Compression Fundamentals, Standards, and Practice*. Kluwer International Series in Engineering and Computer Science, 2002, vol. 642.

[12] Group on Interactive Coding of Images, “TER software; Open Source CCSDS-122-B-1 implementation and extension.” http://www.gici.uab.cat/TER, http://ter.sourceforge.net/, June 2008.

[13] Group on Interactive Coding of Images, “Spectral transform software,” http://gici.uab.cat/, 2010.

[14] D. Taubman, “Kakadu software,” http://www.kakadusoftware.com/, 2000.

[15] M. Klimesh, “Low-complexity adaptive lossless compression of hyperspectral imagery,” *Proc. SPIE*, pp. 63 000N.1–63 000N.9, Sept. 2006.