Research Article

Low-Complexity Compression Algorithm for Hyperspectral Images Based on Distributed Source Coding

Yongjian Nian,[1] Mi He,[2] and Jianwei Wan[1]

[1] *College of Electronic Science and Engineering, National University of Defense Technology, Changsha, Hunan 410073, China*
[2] *School of Biomedical Engineering, Third Military Medical University and Chongqing University, Chongqing 400038, China*

Correspondence should be addressed to Yongjian Nian; yongjian_nian@163.com

A low-complexity compression algorithm for hyperspectral images based on distributed source coding (DSC) is proposed in this paper. The proposed distributed compression algorithm can realize both lossless and lossy compression, which is implemented by performing scalar quantization strategy on the original hyperspectral images followed by distributed lossless compression. Multilinear regression model is introduced for distributed lossless compression in order to improve the quality of side information. Optimal quantized step is determined according to the restriction of the correct DSC decoding, which makes the proposed algorithm achieve near lossless compression. Moreover, an effective rate distortion algorithm is introduced for the proposed algorithm to achieve low bit rate. Experimental results show that the compression performance of the proposed algorithm is competitive with that of the state-of-the-art compression algorithms for hyperspectral images.

1. Introduction

Hyperspectral images compression has received more and more attention in the field of remote sensing. Different from panchromatic images, hyperspectral images are three-dimensional data cube. The spectral resolution of hyperspectral images has achieved nanometer grade, which is widely used in geological survey, environment observing, and military scout. With the incessant increase of the spectral and spatial resolution, the data volume of hyperspectral images expands rapidly, which brings heavy pressure for storage and transmission of hyperspectral images. With respect to the onboard hyperspectral images compression, there is a serious contradiction between data acquirement and transmission. Lossless compression may preserve the image information perfectly with a very low compression ratio, while lossy compression may generate images at different quality levels according to the practical requirements. Therefore, it is necessary to develop an efficient algorithm for onboard hyperspectral images compression, which can realize both lossless and lossy compression.

Onboard compression systems exhibit limited storage memory, computational capacity, and power consumption. Such systems prefer compression algorithms that have an excellent compression performance, a low complexity, and resiliency against errors that are caused by a bad channel environment. The most classical lossy compression methods for hyperspectral images are of a three-dimensional approach, in which some transform (e.g., a wavelet transform, discrete cosine transform, or Karhunen-Loeve transform) is used for spectral decorrelation followed by a two-dimensional transform for spatial decorrelation [1–3]. Although this method and other methods can achieve good performance, their complexity is too high and the memory requirement is too large to be implemented onboard. In general, lossless compression algorithms for hyperspectral images are based on spectral linear prediction. Magli also introduced Kalman filter to implement the spectral linear prediction [4]. Tang et al. introduced ground classification to improve the lossless compression performance [5], which has high encoder complexity and poor error resilience. Distributed source coding (DSC) has received increased attention in the past few years

and has provided separate encoding and joint decoding, which moves the computational complexity from the encoder to the decoder, thus meeting the requirements of onboard compression [6, 7]. The basis of distributed lossless compression is the Slepian-Wolf theory [8], while that of distributed lossy compression is the Wyner-Ziv theory [9]. Compression algorithms based on DSC can be carried out by means of binary error-correcting codes or multilevel coset codes. At present, some DSC compression algorithms have been proposed gradually. For the aspect of binary error-correcting codes, Pan et al. proposed a low-complexity DSC algorithm based on the discrete cosine transform; the experimental results demonstrated that the performance of the proposed algorithm is comparable to that of the distributed algorithm, that is, based on informed quantization [10]. Tang et al. proposed a distributed lossy compression algorithm based on spectral prediction and wavelet transform that computed the syndrome of each bitplane by using LDPC (low density parity check) code, the difficulty of which lies in estimating the correlation between bitplanes quickly and accurately [11]. Cheung et al. studied the problems of correlation estimation subject to complexity constraints and its impact on coding efficiency [12]. The resulting model-based approach for lossy compression of hyperspectral images achieved both accurate estimation results and good compression performance. For the aspect of multilevel coset codes, Magli et al. proposed s-DSC and v-DSC [13]. On the basis of s-DSC, Abrardo et al. proposed DSC-based lossless compression algorithms, called A1, A2, and A3, providing both low complexity and error resilience [14]. Abrardo et al. also proposed a DSC lossy compression algorithm based on informed quantization, which provides competitive lossy compression performance [15]. Nian et al. proposed a lossless and near lossless compression algorithm based on DSC, which only obtains excellent compression performance at high bit rate [16, 17]. At present, although the distributed compression algorithm has low encoder complexity, its compression performance is lower than that of traditional classical algorithms. Furthermore, existed distributed compression algorithm can only realize lossless compression. In order to satisfy the different quality requirements of onboard hyperspectral images, this paper presents a low-complexity compression algorithm for hyperspectral images based on DSC, which can perform both lossless and lossy compression. Experimental results demonstrate the effectiveness of the proposed algorithm, making it suitable for onboard compression.

This paper is organized as follows: Section 2 describes the proposed distributed lossless compression; Section 3 presents the proposed distributed lossy compression algorithm of hyperspectral images; experimental results are demonstrated in Section 4; and the conclusions are given in Section 5.

2. Distributed Lossless Compression

2.1. Introduction of DSC.
DSC introduces the coset partitioning manner to realize lossless compression. The data space is divided into a great deal of subsets with no intersection between arbitrary two subsets, and each pixel must belong to one of the subsets. Let Ω be the data space, which is divided into several subsets with the number of Z, and the aforementioned description can be expressed as

$$\Omega = \mathbf{C}_1 \cup \mathbf{C}_2 \cdots \cup \mathbf{C}_z,$$
$$\mathbf{C}_i \cap \mathbf{C}_j = \Phi \left(i \neq j \right), \tag{1}$$

where Φ is an empty aggregate. DSC encoder realizes data compression only by means of binary error-correcting codes or multilevel coset codes. Binary error-correcting codes demonstrate excellent performance by decomposing the source into a series of bitplanes and compressing each bitplane with Turbo, LDPC, Trellis, or other channel codes. Because binary error-correcting codes neglect the correlation between bitplanes, the compression performance is not satisfactory. Generally, the compression performance of multilevel coset codes is better than that of binary error-correcting codes with a much lower complexity [13]. Suppose the source is represented on n bits; then the multilevel coset codes adopt the principle of (n, k) linear codes, partitioning the set of 2^n possible values into 2^r $(r = n - k)$ cosets with 2^k elements in each coset, and the Euclidian distance between the adjacent elements of each coset is 2^r. A Slepian-Wolf encoder only needs to transmit the label of the coset to which each pixel belongs to the decoder; the decoder then employs the correlated side information to reconstruct the pixel in the corresponding coset indexed by the received coset label.

2.2. Distributed Lossless Compression.
A DSC encoder can compress each source only when an exact knowledge of the correlation is available at the encoder. However, in many practical applications, correlation information may not be available beforehand. Therefore, it is necessary to establish an efficient correlation model subject to the complexity constraints to construct high-quality side information. As we know, the spectral curves of hyperspectral images are nonlinear, and thus, a nonlinear model is suitable for the correlation estimation of hyperspectral images. Unfortunately, nonlinear model of hyperspectral images is difficult to be established. Because the current band is typically correlated with a few previous bands, multilinear regression model can be introduced to construct the high-quality side information of the current band. For the algorithm proposed in this paper, each band of the hyperspectral images is divided into nonoverlapping blocks with a size of $N \times N$. Each block is processed independently, which offers several advantages: first, it can adapt to the local spatial features of a hyperspectral image; second, it can provide error resilience; and finally, it can be easily parallelized. Let $x_{k,i,j}$ denote the pixel of the current block in ith line, jth pixel, and kth band, with $k = 1, 2, \ldots, L$ and $i, j = 1, 2, \ldots, N$. The pixel $x_{k,i,j}$ is predicted linearly from the decoded pixels $x_{k-1,i,j}, x_{k-2,i,j}, \ldots, x_{k-P,i,j}$ of the previous blocks to make the constructed side information as similar as possible to $x_{k,i,j}$ with respect to the standard of minimum mean-squared error. Let μ_k be the average value of the current block, and let μ_{k-l} $(l = 1, 2, \ldots, P)$ be the average

value of the colocated block in the lth band. The constructed side information of the current block is then expressed as

$$\widehat{x}_{k,i,j} = \sum_{l=1}^{P} \alpha_l \left(x_{k-l,i,j} - \mu_{k-l} \right) + \mu_k, \quad i, j = 1, 2, \ldots, N, \quad (2)$$

where $\boldsymbol{\alpha}_k = [\alpha_1, \alpha_2, \ldots, \alpha_P]^T$ are the prediction coefficients minimizing the energy of the prediction errors which can be written as

$$\left(\mathbf{A}\boldsymbol{\alpha}_k - \mathbf{B} \right)^T \left(\mathbf{A}\boldsymbol{\alpha}_k - \mathbf{B} \right), \quad (3)$$

where

$$\mathbf{A} = \begin{bmatrix} x_{k-1,1,1} - \mu_{k-1} & \cdots & x_{k-P,1,1} - \mu_{k-P} \\ \vdots & \ddots & \vdots \\ x_{k-1,N,N} - \mu_{k-1} & \cdots & x_{k-P,N,N} - \mu_{k-P} \end{bmatrix},$$

$$\mathbf{B} = \begin{bmatrix} x_{k,1,1} - \mu_k \\ \vdots \\ x_{k,N,N} - \mu_k \end{bmatrix}. \quad (4)$$

The optimal $\boldsymbol{\alpha}_k$ can be computed as follows:

$$\boldsymbol{\alpha}_k = \begin{bmatrix} r_{k-1,k-1} & \cdots & r_{k-P,k-1} \\ \vdots & \ddots & \vdots \\ r_{k-1,k-P} & \cdots & r_{k-P,k-P} \end{bmatrix}^{-1} \begin{bmatrix} r_{k,k-1} \\ \vdots \\ r_{k,k-P} \end{bmatrix}, \quad (5)$$

where

$$r_{k,k-l} = \sum_{i=1}^{N} \sum_{j=1}^{N} \left(x_{k,i,j} - \mu_k \right) \left(x_{k-l,i,j} - \mu_{k-l} \right). \quad (6)$$

The corresponding errors are written as

$$e_{k,i,j} = x_{k,i,j} - \widehat{x}_{k,i,j}. \quad (7)$$

It should be noted that the correct reconstruction can be obtained only when the Euclidian distance between the adjacent elements in the same coset is twice as large as the maximum absolute error, which can be expressed as

$$2^{r-1} > \max_{i,j=1,2,\ldots,N} \left(\left| e_{k,i,j} \right| \right). \quad (8)$$

Hence, the corresponding rate of the block is given as

$$r = \left\lfloor \log_2 \left(\max_{i,j=1,2,\ldots,N} \left(\left| e_{k,i,j} \right| \right) \right) \right\rfloor + 2. \quad (9)$$

3. Distributed Lossy Compression

3.1. Scalar Quantization Strategy. Compared with distributed lossless compression, distributed lossy compression employs quantization strategy to quantize the original hyperspectral images followed by distributed lossless compression, which can introduce distortion for the original hyperspectral images. Therefore, the performance of quantization strategy is crucial for distributed lossy compression. In order to keep the encoder low complexity, practical scalar quantization manner is performed on each block. Let q_k denote the quantizer step size of the current block in the kth band. The quantized values of the current block and its side information are given as

$$y_{k,i,j} = \text{round} \left(\frac{x_{k,i,j}}{q_k} \right), \qquad \overline{y}_{k,i,j} = \text{round} \left(\frac{\overline{x}_{k,i,j}}{q_k} \right). \quad (10)$$

Let $\widehat{x}_{k,i,j}$ be the reconstructed value of $x_{k,i,j}$, and $\widehat{\mu}_k$ is the average value of the corresponding reconstructed block in the kth band. The side information of the current block is constructed by the reconstructed version of the corresponding blocks in the previous bands, which is written as

$$\overline{x}_{k,i,j} = \sum_{l=1}^{P} \widehat{\alpha}_l \left(\widehat{x}_{k-l,i,j} - \widehat{\mu}_{k-l} \right) + \widehat{\mu}_k, \quad (11)$$

where $\widehat{\alpha}_k(\widehat{\alpha}_1, \widehat{\alpha}_2, \ldots, \widehat{\alpha}_P)$ are the quantized versions of $\boldsymbol{\alpha}_k$. As aforementioned, in order to ensure the correct reconstruction, the following condition must be satisfied:

$$\left| y_{k,i,j} - \overline{y}_{k,i,j} \right| < 2^{R_k - 1}; \quad (12)$$

thus, the optimal quantized step can be computed as follows:

$$q_k = \left\lfloor \frac{\max_{i,j=1,2,\ldots,N} \left| \overline{e}_{k,i,j} \right|}{2^{R_k - 1}} \right\rfloor + c, \quad (13)$$

where

$$\overline{e}_{k,i,j} = x_{k,i,j} - \overline{x}_{k,i,j} \quad (14)$$

and c is a positive integer. For most blocks, the value of c equals one. If any block fails to satisfy (12), the value of c is to add one every time until (12) is satisfied. The final output bit rate can be computed as

$$R_k = \left\lceil \log_2 \left\lceil \frac{\max_{i,j=1,2,\ldots,N} \left| \overline{e}_{k,i,j} \right|}{q_k - c} \right\rceil \right\rceil + 1. \quad (15)$$

It should be noted that the quantization strategy introduces no loss for the original hyperspectral images when $q_k = 1$; in this case, the distributed lossy compression algorithm turns to the distributed lossless compression algorithm.

3.2. Rate Distortion for Low Bit-Rate Compression. As can be seen from (9), the minimum value of the coding rate is 2; in other words, the bit rate acquired by the above quantization algorithm cannot be lower than 2, which can be regarded as a near lossless compression. In order to achieve low bit-rate compression, we perform an effective rate-distortion (RD) algorithm on the proposed compression algorithm to obtain the low bit rate. For a certain block, if its energy of prediction errors is low, which means that the prediction performance of this block is perfect, the encoding process of this block may be neglected, otherwise, the encoding process must be executed

[18]. This working manner provides an appropriate tradeoff between the distortion introduced by neglecting the encoding process and the corresponding rate saving. In practice, the energy of the prediction errors of the block which is shown as (16) is used to determine whether the block should be neglected:

$$D = \sum_{i=1}^{N} \sum_{j=1}^{N} e_{k,i,j}^2. \tag{16}$$

If the distortion D is higher than the threshold, which means that the prediction error has too much energy, neglecting the encoding process will result in a poor compression performance. On the contrary, if D is below the threshold, neglecting the encoding process of this block will have limited impact on the compression performance. In this situation, the prediction errors of this block are set to be zero and only the predictor parameters are written in the encoding stream. By setting different values of the threshold, we can change the output bit-rate flexibly.

4. Results and Discussion

Hyperspectral images acquired by the AVIRIS sensor in 1997 were employed in our experiments. AVIRIS was devised by the JPL (jet propulsion laboratory) of NASA (National Aeronautics and Space Administration, USA), and it covers the 0.41–2.5 μm spectrum range in 10 nm bands. This instrument contains four spectrometers that are flown at a 20 km altitude with a 17 m spatial resolution. The four scenes are Cuprite, Jasper Ridge, Lunar Lake, and Low Altitude, which are widely used for compression testing and the evaluation of hyperspectral images. The radiance data of the above four scenes were represented in 16 bits; each image has 512 lines, 224 bands, and 512 pixels/line. In this paper, the four scenes with image size of $256 \times 256 \times 224$ are used in our experiment.

The proposed algorithm can realize both lossless and lossy compression. We first evaluated the lossless compression performance of the proposed algorithm with several existed classical algorithms. In general, the multiband linear prediction model is expected to improve the quality of side information; however, it also increases the computational complexity. Therefore, a reasonable number P should be typically selected that guarantees both a high quality for the constructed side information and a low complexity for the multilinear regression model. Figure 1 shows the prediction performance and the computational complexity with various values of P, where $P = 0$ refers to the average entropy over all the bands for each scene. As shown in Figure 1, the multilinear regression model using two previous bands provides both high prediction performance and low computational complexity. Note that as the number of P increases, the computational complexity increases rapidly while the prediction gain tends to decrease because bands further away from the current band in the wavelength domain are less correlated. Therefore, the optimal number of P is selected to be 2.

For the proposed algorithm, the block size is also influential for the compression performance. We use bpppb (bit per
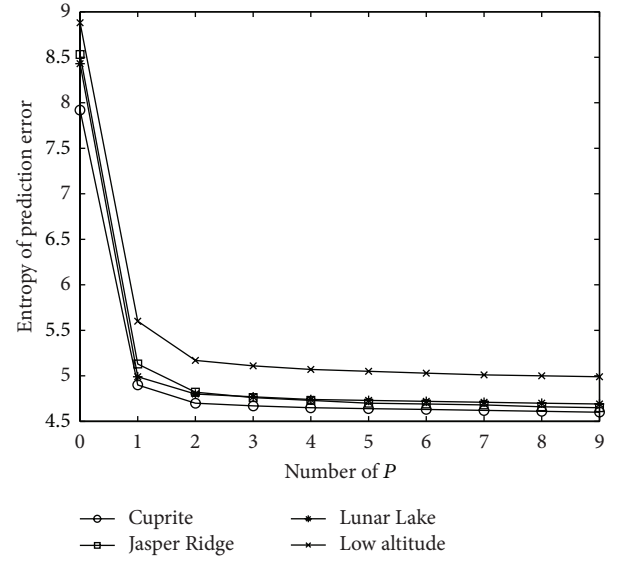
(a) Comparison of the entropy of prediction error
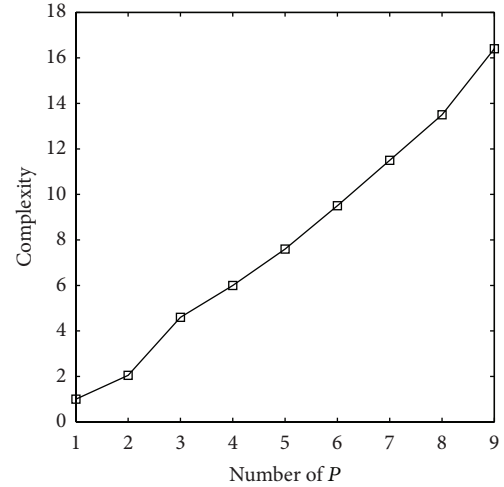
(b) Comparison of the complexity

Figure 1: Prediction performance and complexity with various P values.

pixel per band) to measure the lossless compression performance. Figure 2 gives the lossless compression performance with different block size ($N = 8, 16, 32, 64, 128, 256$). It can be seen that the best compression performance is obtained when $N = 16$. With the increase of the N number, the compression performance decreases rapidly. For the proposed algorithm, small block can make full use of the spatial correlation, which is beneficial to achieve better compression performance. However, small block size also introduces large additive information, such as predictive coefficients and average value of corresponding block. Although large block introduces little additive information, it cannot make full use of the spatial correlation, which leads to a bad compression performance.

The proposed distributed lossless compression algorithm is compared with several classical algorithms, such as JPEG-LS, 3D CALIC [19], KSP (kalman spectral prediction)
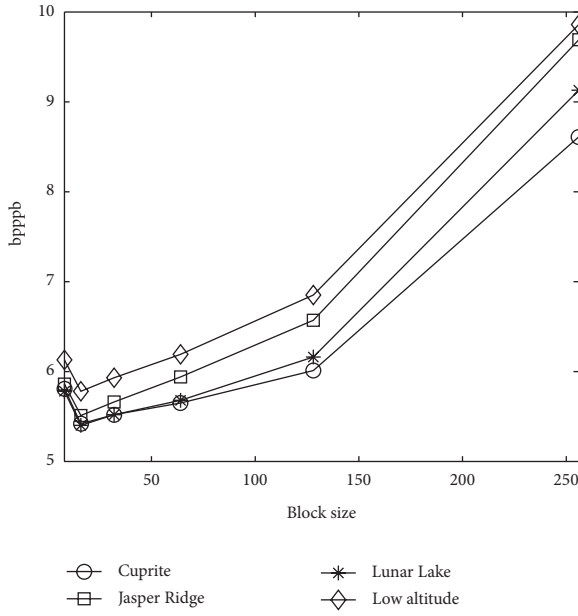
Figure 2: Lossless compression performance with different block size.

Table 1: Comparison of lossless compression performance of various algorithms.

	Cuprite	Jasper Ridge	Lunar Lake
JPEG-LS	7.62	8.16	7.55
3D CALIC	5.18	5.14	5.19
KSP	4.88	4.95	4.89
LUT	4.65	4.95	4.71
s-DSC	6.08	6.25	6.23
A1	5.50	5.60	5.51
DSC-lossless	5.42	5.51	5.40

Table 2: Comparison of complexity of various algorithms.

Algorithm	Complexity
JPEG-LS	1
LUT	1.14
3D CALIC	28.4
KSP	25.2
s-DSC	0.89
DSC-lossless	0.78

[4], LUT (look-up table) [20], s-DSC [13], and A1 [14], which are shown in Table 1. JPEG-LS is an excellent lossless compression standard. 3D CALIC, KSP, and LUT are three traditional classical algorithms. s-DSC and A1 are two classical distributed lossless compression algorithms. As can be seen, JPEG-LS performs poorly due to the lack of spectral correlation elimination. LUT has the highest compression performance, and KSP also has the excellent performance by using Kalman filter to eliminate the spectral correlation. The performance of 3D CALIC is much better than JPEG-LS but worse than KSP. s-DSC is only better than JPEG-LS, and A1 is much better than s-DSC. The performance of the proposed algorithm (DSC-lossless) is slightly better than A1. However, there is also a gap between the proposed algorithm and the above two traditional algorithms.

The complexity of the proposed algorithms has been compared with that of existing algorithms, which are reported in Table 2 and are normalized with respect to the complexity of JPEG-LS. As can be seen, LUT is slightly more complex than JPEG-LS; 3D CALIC and KSP are both more complex than JPEG-LS. Because only LSBs are transmitted without any entropy coding of the predictive errors, the proposed DSC-Lossless has very low encoder complexity, even lower than JPEG-LS, while providing better compression performance than the other two distributed lossless compression algorithms. Despite that the complexity of s-DSC is significantly small, which is slightly higher than DSC-lossless, the performance of s-DSC is much worse than DSC-lossless.

In this paper, we use bpppb and signal to noise ratio (SNR) to measure the lossy compression performance. Figure 3 gives the comparison of lossy compression performance of various algorithms with different bit rate, where DSC-lossy is the proposed algorithm and the other algorithms are the classical algorithms for hyperspectral images compression. JPEG2000

is an excellent compression standard which is primary used for still image compression. DWT-JPEG2000 is also a popular compression algorithm which removes spectral redundancy by using a 9–7 biorthogonal wavelet transform followed by JPEG2000 for spatial compression. Similar with DWT-JPEG2000, KLT-JPEG2000 removes spectral redundancy by using KLT transform followed by JPEG2000 for spatial compression. For the hyperspectral images compression, it is very important to remove its spectral correlation. As can be seen from Figure 3, despite JPEG2000 has the perfect compression performance for still images, its compression performance for hyperspectral images is the worst because it do not remove the spectral correlation. KLT-JPEG2000 has the best compression performance due to the excellent performance of removing spectral correlation. Note that because the filter coefficients of DWT are fixed, its performance of removing spectral correlation is moderate. DWT-JPEG2000 employs DWT to remove the spectral correlation, and the performance of DWT-JPEG2000 is better than that of JPEG2000 while worse than that of KLT-JPEG2000. It should be noted that the performance of the proposed DSC-lossy is comparative to that of DWT-JPEG2000. The performance of DSC-lossy is worse than that of DWT-JPEG2000 at low bit rate while better than that of DWT-JPEG2000 at high bit rate; this is because that the multilinear regression model has serious error accumulation at low bit rate, while at high bit rate this disadvantage can be neglected.

In terms of encoder complexity, the results are reported in Table 3 and normalized with respect to the complexity of JPEG2000. The complexity of KLT-JPEG2000 is the highest, although KLT-JPEG2000 provides the best compression performance. DWT-JPEG2000 has the modest complexity, which is faster than that of KLT-JPEG2000 but lower than that of JPEG2000, but there is a performance gap with KLT-JPEG2000. The proposed algorithm has similar
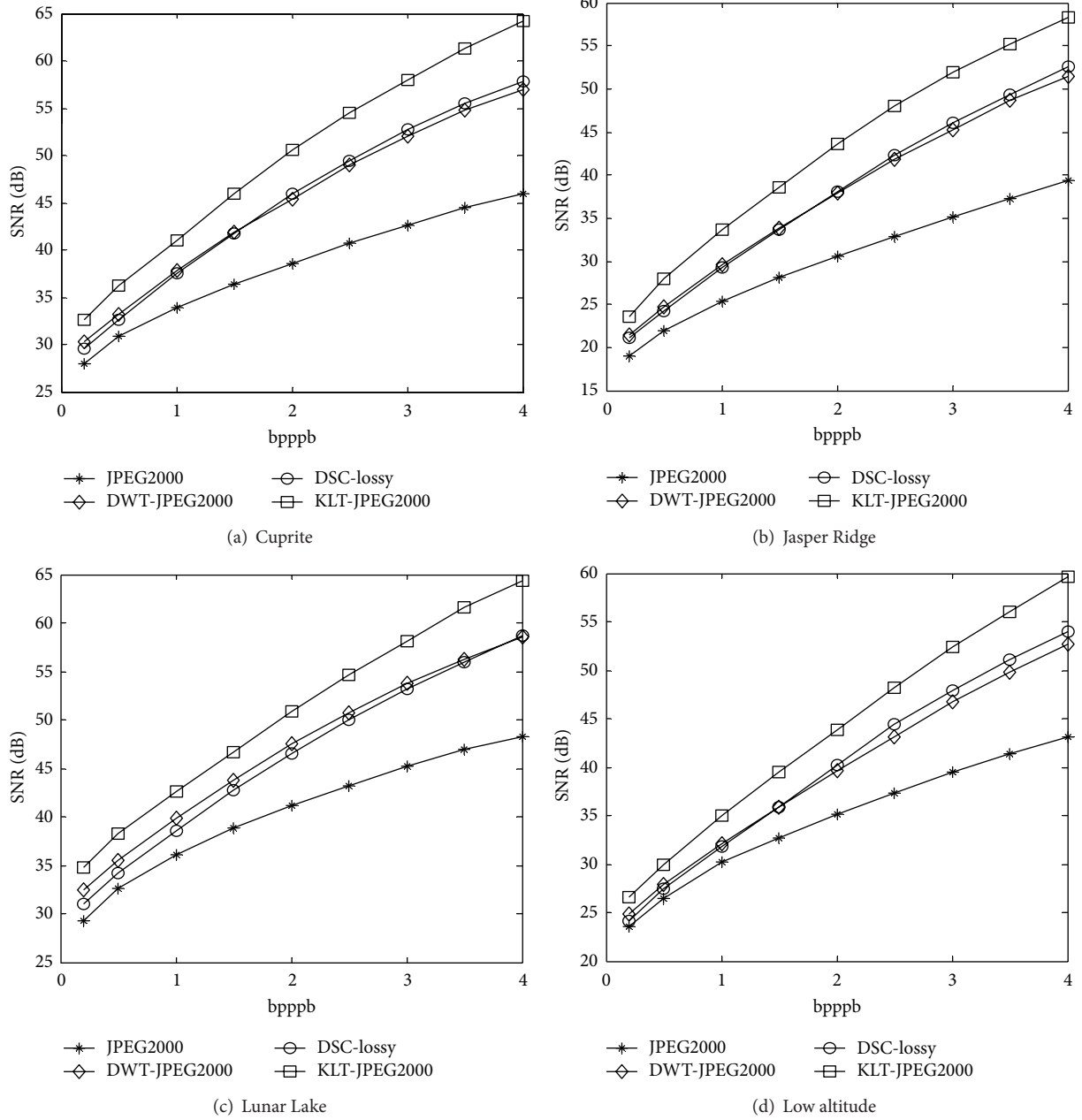
(a) Cuprite

(b) Jasper Ridge

(c) Lunar Lake

(d) Low altitude

FIGURE 3: Comparison of RD performance of various algorithms.

TABLE 3: Comparison of encoder complexity of various algorithms.

Algorithm	Complexity
JPEG2000	1
DWT-JPEG2000	1.78
KLT-JPEG2000	4.12
DSC-lossy	0.96

compression performance with DWT-JPEG2000, but its encoder complexity is the lowest. Furthermore, the proposed algorithm can be easily parallelized by having encoding

blocks compressed at the same time, which is suitable for the implementation on an FPGA (field-programmable gate array).

5. Conclusion

Although existing classical algorithms can provide excellent compression performance for hyperspectral images, they are not suitable for onboard compression due to the high encoder complexity and poor error resilience. This paper proposes a low-complexity compression algorithm for hyperspectral images based on distributed source coding. Multilinear

regression model is introduced to improve the compression performance of distributed lossless compression algorithm, and optimal scalar quantization is proposed to perform distributed lossy compression. In order to achieve low bit-rate compression, an effective rate-distortion algorithm is also performed on the proposed distributed lossy algorithm. Experimental results show that the proposed algorithm has competitive compression performance and low encoder complexity, making it suitable for onboard compression. In order to further improve the performance of the proposed algorithm at low bit rate, powerful RD algorithm should be paid more attention.

Conflict of Interests

The authors have no conflict of interests to disclose.

Acknowledgment

References

[1] I. Blanes and J. Serra-Sagrista, "Pairwise orthogonal transform for spectral image coding," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 49, no. 3, pp. 961–972, 2011.

[2] B. Penna, T. Tillo, E. Magli, and G. Olmo, "Transform coding techniques for lossy hyperspectral data compression," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 45, no. 5, pp. 1408–1421, 2007.

[3] X. Tang, S. Cho, and W. A. Pearlman, "3D set partitioning coding methods in hyperspectral image compression," in *Proceedings of the International Conference on Image Processing (ICIP '03)*, pp. 239–242, Barcelona, Spain, September 2003.

[4] E. Magli, "Multiband lossless compression of hyperspectral images," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 47, no. 4, pp. 1168–1178, 2009.

[5] Y. Tang, Q. Xin, G. Li, and J.-W. Wan, "Lossless compression of hyperspectral images based on contents," *Optics and Precision Engineering*, vol. 20, no. 3, pp. 668–674, 2012.

[6] S. S. Pradhan and K. Ramchandran, "Distributed source coding using syndromes (DISCUS): design and construction," *IEEE Transactions on Information Theory*, vol. 49, no. 3, pp. 626–643, 2003.

[7] Z. Xiong, A. D. Liveris, and S. Cheng, "Distributed source coding for sensor networks," *IEEE Signal Processing Magazine*, vol. 21, no. 5, pp. 80–94, 2004.

[8] D. Slepian and J. K. Wolf, "Noiseless coding of correlated information sources," *IEEE Transactions on Information Theory*, vol. 19, no. 4, pp. 471–480, 1973.

[9] A. D. Wyner and J. Ziv, "The rate-distortion function for source coding with side information at the decoder," *IEEE Transactions on Information Theory*, vol. 22, no. 1, pp. 1–10, 1976.

[10] X. Pan, R. Liu, and X. Lv, "Low-complexity compression method for hyperspectral images based on distributed source coding," *IEEE Geoscience and Remote Sensing Letters*, vol. 9, no. 2, pp. 224–227, 2012.

[11] C. Tang, N.-M. Cheung, A. Ortega, and C. S. Raghavendra, "Efficient inter-band prediction and wavelet based compression for hyperspectral imagery: a distributed source coding approach," in *Proceedings of the Data Compression Conference*, pp. 437–446, Snowbird, Utah, March 2005.

[12] N.-M. Cheung, H. Wang, and A. Ortega, "Sampling-based correlation estimation for distributed source coding under rate and complexity constraints," *IEEE Transactions on Image Processing*, vol. 17, no. 11, pp. 2122–2137, 2008.

[13] E. Magli, M. Barni, A. Abrardo, and M. Grangetto, "Distributed source coding techniques for lossless compression of hyperspectral images," *EURASIP Journal on Advances in Signal Processing*, vol. 2007, Article ID 45493, 13 pages, 2007.

[14] A. Abrardo, M. Barni, E. Magli, and F. Nencini, "Error-resilient and low-complexity onboard lossless compression of hyperspectral images by means of distributed source coding," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 48, no. 4, pp. 1892–1904, 2010.

[15] A. Abrardo, M. Barni, and E. Magli, "Low-complexity lossy compression of hyperspectral images via informed quantization," in *Proceedings of the 17th IEEE International Conference on Image Processing (ICIP '10)*, pp. 505–508, Hong Kong, September 2010.

[16] Y. J. Nian, Q. Xin, Y. Tang, and J. W. Wan, "Distributed lossless compression of hyperspectral images based on multi-band prediction," *Optics and Precision Engineering*, vol. 20, no. 4, pp. 906–912, 2012.

[17] Y. J. Nian, J. W. Wan, Y. Tang, and B. Chen, "Near lossless compression of hyperspectral images based on distributed source coding," *Science China Information Sciences*, vol. 55, no. 11, pp. 2646–2655, 2012.

[18] A. Abrardo, M. Barni, and E. Magli, "Low-complexity predictive lossy compression of hyperspectral and ultraspectral images," in *Proceedings of the 36th IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '11)*, pp. 797–800, Prague, Czech Republic, May 2011.

[19] X. Wu and N. Memon, "Context-based lossless interband compression: extending CALIC," *IEEE Transactions on Image Processing*, vol. 9, no. 6, pp. 994–1001, 2000.

[20] J. Mielikainen, "Lossless compression of hyperspectral images using lookup tables," *IEEE Signal Processing Letters*, vol. 13, no. 3, pp. 157–160, 2006.