

Segmented Principal Component Analysis for Parallel Compression of Hyperspectral Imagery

Qian Du, *Senior Member, IEEE*, Wei Zhu, *Student Member, IEEE*, He Yang, *Student Member, IEEE*, and James E. Fowler, *Senior Member, IEEE*

Abstract—Principal component analysis (PCA) is widely used for spectral decorrelation in the JPEG2000 compression of hyperspectral imagery. However, due to the data-dependent nature of principal components, the principal component transform matrix is stored in the JPEG2000 bitstream, constituting an overhead that is often negligible if the spatial size of the image is large. However, in parallel compression in which the data set is partitioned to multiple independent processing nodes, the overhead may no longer remain negligible. It is shown that a segmented approach to PCA can greatly mitigate the detrimental effects of transform-matrix overhead and can outperform wavelet-based decorrelation which entails no such overhead.

Index Terms—Hyperspectral compression, principal component analysis (PCA), spectral segmentation.

I. INTRODUCTION

FOR THE compression of hyperspectral imagery, JPEG2000 coupled with principal component analysis (PCA) for spectral decorrelation is becoming a preferred paradigm (e.g., [1]–[3]). A number of studies have shown that PCA-based spectral decorrelation in conjunction with JPEG2000 can provide rate-distortion performance superior to the corresponding JPEG2000 coder using a discrete wavelet transform (DWT) for spectral decorrelation as well as other techniques based on 3-D wavelet transforms.

However, there are two main drawbacks in using PCA for hyperspectral image compression that both stem from the fact that the PCA transform is data-dependent and is calculated at the encoder for each data set to be coded: 1) the computational complexity of PCA, which typically involves a covariance-matrix calculation as well as an eigendecomposition, and 2) the requirement that the PCA transform matrix be stored within the compressed bitstream in order for the decoder to be able to invert the spectral transform. Although the first of these certainly serves as a hindrance to PCA in many applications, the second issue is the primary focus of this letter, since it may lead to compression failure. Specifically, the JPEG2000 encoder embeds the mean vector, as well as the inverse transform matrix, into the JPEG2000 bitstream in the form of a multicomponent transform marker, as specified in part 2 of the JPEG2000 standard [4]. The burden of this transform-matrix

overhead is thus dependent on the number of spectral bands but may be negligible if the spatial size of the hyperspectral image is large. However, if the spatial size of the image is small with respect to the number of spectral bands, the transform-matrix overhead may no longer be negligible. In fact, if the overhead is so large as to consume the entire budget of bits allocated to the bitstream, then no bits are available to represent the transform coefficients, and the compression operation can be considered to have failed.

In this letter, we investigate this issue of transform-matrix overhead in the context of parallel compression in which we partition a hyperspectral image spatially or spectrally into multiple subscenes to be distributed to and compressed by multiple independent processing elements (PEs) (e.g., [5]). In this parallel-compression paradigm, the hyperspectral image as a whole is likely to have a very large spatial extent (hence, the motivation for parallel compression in the first place); however, depending on how the data set is partitioned, the individual subscene compressed by a given PE may have a spatial size that is small with respect to the number of spectral bands and, thus, may be detrimentally affected by the transform-matrix overhead.

As the primary contribution of this letter, we propose a segmented PCA (SPCA) [6] to mitigate this problem of transform-matrix overhead. We observe that, even for small spatial sizes relative to spectral dimension, SPCA coupled with a global JPEG2000 bit allocation is largely immune to detrimental effects from transform-matrix overhead and outperforms the DWT-based spectral transform which incurs no such overhead. In this sense, SPCA prevents compression failure that may plague non-SPCA when spatial sizes are small relative to spectral dimension. As an added benefit, SPCA also significantly reduces the computational complexity associated with the spectral transform, the other main drawback of PCA. As a consequence, SPCA enables highly parallelized implementation of JPEG2000 compression that avoids compression failure due to the transform overhead while, at the same time, reducing the computational burden placed on each PE.

We note that our primary focus in this letter is on the archiving and distribution scenario in which end-stage data products are stored and made available to customer applications. We assume that the data needs of these applications are known such that lossy compression may provide significant compression while preserving the information in the original data that is essential to the application. Along these lines, we focus on the compression of reflectance data since reflectance-domain compression is preferable to radiance-domain compression for applications requiring atmospheric correction prior to data analysis [7]. That said, the strategies we employ in this letter to

Manuscript received April 3, 2009; revised May 13, 2009. First published July 28, 2009; current version published October 14, 2009.

The authors are with the Department of Electrical and Computer Engineering, Geosystems Research Institute—High Performance Computing Collaboratory, Mississippi State University, Mississippi State, MS 39762 USA (e-mail: du@ece.msstate.edu; wz40@msstate.edu; hy58@msstate.edu; fowler@ece.msstate.edu).

Digital Object Identifier 10.1109/LGRS.2009.2024175

TABLE I
SNR PERFORMANCE IN DECIBELS

	Bitrate				
	$0.5R_{\min}$	R_{\min}	$1.1R_{\min}$	$1.5R_{\min}$	$2R_{\min}$
Size: $512 \times 512 \times 224$; $R_{\min} = 0.0273$ bpppb					
DWT	12.54	13.85	14.10	14.95	15.92
PCA	5.45	5.45	12.55	16.08	18.68
Size: $256 \times 256 \times 224$; $R_{\min} = 0.1094$ bpppb					
DWT	17.31	20.58	21.09	23.05	25.16
PCA	7.49	7.49	16.24	23.96	28.55
Size: $128 \times 128 \times 224$; $R_{\min} = 0.4375$ bpppb					
DWT	25.40	31.06	31.84	34.40	36.85
PCA	9.86	9.86	21.67	33.28	37.87
Size: $256 \times 256 \times 112$; $R_{\min} = 0.0547$ bpppb					
DWT	18.27	21.38	21.91	23.78	25.71
PCA	7.97	7.97	16.95	26.03	30.59
Size: $128 \times 128 \times 112$; $R_{\min} = 0.2188$ bpppb					
DWT	27.03	32.46	33.21	35.78	38.08
PCA	11.85	11.85	23.86	36.29	41.05
Size: $128 \times 128 \times 56$; $R_{\min} = 0.1094$ bpppb					
DWT	25.17	29.67	30.57	32.84	34.99
PCA	12.53	12.53	15.25	27.16	34.37

mitigate PCA-overhead issues are general and likely to extend to other settings.

II. SPATIAL SIZE AND PCA FOR JPEG2000

Our focus in this section is on parallel implementations of compression in which a data set is partitioned spectrally or spatially in order to distribute data to multiple parallel processing units (e.g., [5]). However, cropping data sets to smaller spatial or spectral sizes can occur in other situations as well—for example, when end users are interested in only a certain part of a larger scene or in only several bands out of a much larger hyperspectral volume, i.e., a region of interest. To illustrate typical results, we focus on the Jasper Ridge AVIRIS hyperspectral reflectance data set, and we focus on only JPEG2000 used with DWT or PCA spectral decorrelation. To study the impact of image spatial size on compression performance, we consider the original image with size 512×512 and subscenes of sizes 256×256 and 128×128 obtained by cropping the original image. Results are listed in Table I, where distortion is measured as a signal-to-noise ratio (SNR) defined as the log ratio of signal variance to noise variance, and bitrate is measured in terms of bits per pixel per band (bpppb).

To understand Table I, recall that, as aforementioned, when PCA is employed for spectral decorrelation, the JPEG2000 encoder embeds the transform matrix into the bitstream. Assuming that the matrix is represented using 32-b floating-point values, for N spectral bands, we have an overhead of $32L^2$ bits for the transform matrix; in terms of bitrate, this overhead constitutes a minimum bitrate of $R_{\min} = 32L/N^2$ bpppb for a data set of size $N \times N \times L$. That is, the PCA transform matrix will occupy R_{\min} bpppb out of the bitstream rate of R bpppb; thus, we must code at a rate of $R > R_{\min}$ in order for the transform coefficients to receive any bits dedicated to their coding. If, on the other hand, $R \leq R_{\min}$, the JPEG2000 decoder will essentially output zero for all coefficients, leading to an exceedingly poor representation. Additionally, the rate-distortion performance will be reduced at rates near R_{\min} due to

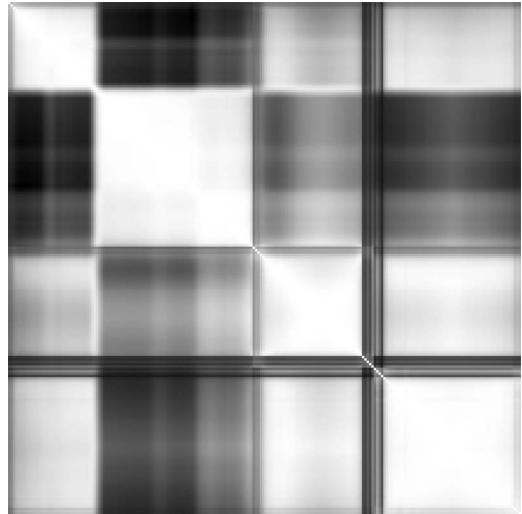


Fig. 1. 224×224 matrix of cross-band correlation coefficients of the Jasper Ridge AVIRIS data set. White = ± 1 . Black = 0.

the overly large portion of the rate consumed by the transform matrix. In this sense, compression with PCA *fails* at rates near or below R_{\min} , since no coefficients are transmitted.

Table I indicates R_{\min} for the various cropped data-set sizes considered. Indeed, DWT spectral decorrelation outperforms PCA at low rates near R_{\min} , but PCA spectral decorrelation starts to outperform the DWT as the bitrate increases. We note that the spectral DWT is fixed and not stored in the JPEG2000 bitstream and, so, does not suffer from the same issue of transform-matrix overhead.

To study the impact of image spectral size (i.e., the number of bands) on compression performance, we crop the data set to the first 112 or first 56 bands. Again, Table I tabulates the resulting rate-distortion performances. We see that, as expected, decreasing L decreases R_{\min} ; the reduced transform-matrix overhead results then in better PCA performance at low rates near R_{\min} with respect to the DWT.

III. SPCA FOR JPEG2000 COMPRESSION

It is clear from the results in Section II that if the spatial size of the data set being coded is relatively small with respect to the spectral size, then the overhead required for transmitting the PCA transform matrix within the compressed bitstream becomes relatively large and results in a detrimental impact on compression performance. Consequently, the straightforward approach to parallel data compression—simply tiling the data set spatially and running PCA and JPEG2000 independently on each tile—is insufficient for efficient compression performance.

However, one can dramatically reduce the storage overhead associated with PCA by partitioning a data set spectrally and applying PCA independently within each group of spectral bands. This straightforward approach, SPCA [6], can be carried out in a number of ways. Perhaps the simplest segmentation is to partition the $N \times N \times L$ data set into M groups of spectral bands, each containing the same number (i.e., L/M) of bands. The transform-matrix overhead is cut significantly, going from $O(L^2)$ to $O(L^2/M)$. We call this approach SPCA with *uniform* spectral partitioning (SPCA-U).

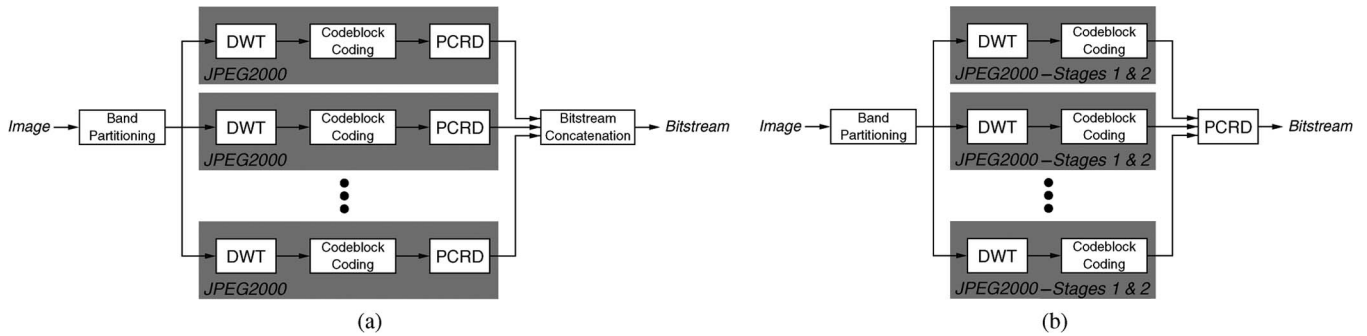


Fig. 2. Bit allocation for SPCA coding. (a) LBA. (b) GBA. Note that, in LBA, each band group is allocated the same number of bits while, in GBA, PCRD distributes available bits optimally between band groups.

A more sophisticated band-partitioning scheme was proposed in [6]. As shown in Fig. 1, the spectral bands in a hyperspectral image can be naturally grouped based on the correlation coefficient between bands such that the bands in each group are highly correlated with one another. In this case, the original L bands are partitioned into M groups with each group having L_m bands such that $L = \sum_{m=1}^M L_m$. The corresponding transform-matrix overhead is then $O(\sum_{m=1}^M L_m^2)$. We call this approach SPCA with *correlation-coefficient* spectral partitioning (SPCA-CC).

In either case of SPCA-U or SPCA-CC, JPEG2000 is applied to the groups of spectral bands in the following manner. Within a given group of spectral bands, the PCA transform is determined and applied to decorrelate the spectral bands. Then, the three stages of JPEG2000 coding are applied—a spatial DWT, codeblock coding (CBC), and, finally, post-compression rate-distortion (PCRD) optimal truncation of the codeblock bitstreams to form the final compressed bitstream. The most straightforward approach would be to perform all three JPEG2000 stages on each spectral group independently with the resulting bitstreams concatenated as a final step, as shown in Fig. 2(a). In this case, JPEG2000’s PCRD optimization is applied only locally within a group of bands. An alternative would be to apply the first two stages (DWT and CBC) independently within each group while the final stage (PCRD) is applied globally across all codeblocks in all groups of bands, as shown in Fig. 2(b). We call the former local bit allocation (LBA) while the latter is termed global bit allocation (GBA). We note that LBA is perhaps better suited conceptually to the parallel-compression paradigm, since the entire JPEG2000 encoding process along with its associated spectral PCA can be entirely computed on a separate PE, while GBA requires that the final PCRD process be conducted in a nonparallel fashion. However, PCRD is relatively lightweight from a computational perspective such that the final nonparallel PCRD step shown in Fig. 2(b) is not likely to be much more of a hindrance to parallelism than the nonparallel concatenation step shown in Fig. 2(a).

We note that, in addition to a reduced transform-matrix overhead as described earlier, SPCA offers other advantages. For one, band decorrelation may be more efficient, since PCA is applied to bands that share a higher degree of correlation [6]. This is why SPCA can improve the performance of PCA even when PCA works. Additionally, computational complexity of determining the PCA transform is greatly reduced. Specifically, the complexity of calculating the data covariance matrix Σ

of the original data is $O(N^2L^2)$; similarly, for the covariance Σ_m of group m of SPCA, we have complexity $O(N^2L_m^2)$, yielding total complexity $O(\sum_{m=1}^M N^2L_m^2)$ which is less than $O(N^2L^2)$, since $\sum_{m=1}^M L_m^2 < L^2 = (\sum_{m=1}^M L_m)^2$. Furthermore, the computational complexity of the requisite eigendecomposition of Σ is $O(L^3)$; again, we have $O(\sum_{m=1}^M L_m^3) < O(L^3)$. Thus, SPCA can mitigate the computational burden of the PCA significantly [8].

We note that, in general, the larger the degree of spectral partitioning, the faster SPCA will be. Yet, smaller spectral sizes in each subsense will cause the overall principal component decomposition effectuated by SPCA to deviate to a greater degree from the PCA decomposition on the original spectral size. That is, subsenses with small spectral sizes have covariance matrices and corresponding eigendecompositions that deviate significantly from that of the original data set, resulting in less efficient spectral decorrelation. Thus, increasing spectral partitioning will decrease not only the computational complexity of SPCA but also the quality of its decorrelation.

IV. EXPERIMENTAL RESULTS

We now investigate the performance of SPCA on the Jasper Ridge reflectance data, considering the same six spatially and spectrally cropped subsenses that were examined in Section II. We employ SPCA using both uniform-based spectral segmentation (SPCA-U) as well as segmentation driven by the correlation coefficient (SPCA-CC), and we use the same number of spectral groups in both cases, as listed in Table II. Finally, we use both LBA [Fig. 2(a)] and GBA [Fig. 2(b)], yielding four SPCA approaches: SPCA-U-LBA, SPCA-U-GBA, SPCA-CC-LBA, and SPCA-CC-GBA. SNR results are compiled in Table III for three bitrates— R_{\min} , $1.1 R_{\min}$, and $1.5 R_{\min}$, where, as before, R_{\min} depends on the spatial and spectral sizes of the subsense being coded.

In Table III, we see that, in all cases, GBA provides a significant performance advantage over LBA, with the best performance resulting from CC-based segmentation coupled with GBA. Interestingly, if LBA is used, then, in many cases, CC-based segmentation performs slightly worse than uniform segmentation—this is due to some large band groups in the CC-based spectral segmentation which yield a relatively high rate overhead for the PCA transform matrix. GBA, which adjusts bitrate allocation across all groups, compensates for this effect. We note that, in all cases, SPCA-CC-GBA yields significant performance improvement over both the DWT-based spectral

TABLE II
PARTITIONING OF SPECTRAL BANDS

Size	SPCA-CC	SPCA-U
$512 \times 512 \times 224$	1-38, 39-107, 108-113, 114-154, 155-166, 167-224	1-37, 38-74, 75-111, 112-148, 149-185, 189-224
$256 \times 256 \times 224$	1-38, 39-107, 108-113, 114-154, 155-166, 167-224	1-37, 38-74, 75-111, 112-148, 149-185, 189-224
$128 \times 128 \times 224$	1-38, 39-82, 83-107, 108-113, 114-154, 155-166, 167-224	1-32, 33-64, 65-96, 97-128, 129-160, 161-192, 193-224
$256 \times 256 \times 112$	1-38, 39-107, 108-112	1-37, 38-74, 75-112
$128 \times 128 \times 112$	1-38, 39-82, 83-107, 108-112	1-28, 29-56, 57-84, 85-112
$128 \times 128 \times 56$	1-19, 20-38, 39-56	1-18, 19-36, 37-56

transform as well as the non-SPCA spectral transform. The gain in SNR of SPCA-CC-GBA over the non-SPCA transform is particularly large when R_{\min} is relatively high, which occurs when the spectral size of the subscene is large or the spatial size is small. This is as expected, since a large R_{\min} implies that non-SPCA is excessively burdened with rate overhead needed to store the PCA transform matrix in the bitstream. SPCA significantly reduces this overhead.

To provide a more comprehensive view of performance, we show in Figs. 3 and 4 SNR for rates ranging from 0.1 to 1.0 bpppb for both the $512 \times 512 \times 224$ and $128 \times 128 \times 224$ subscene sizes. These sizes are chosen because, in Table III, the non-SPCA transform achieves the best performance for the former and the worst for the latter. As shown in Fig. 3, when PCA-based JPEG2000 coding works properly (i.e., the spatial size is relatively large with respect to the spectral size such that the PCA transform matrix occupies negligible rate overhead in the compressed bitstream), the performance of SPCA-CC-GBA is slightly below that of non-SPCA. However, as shown in Fig. 4, when the non-SPCA is overwhelmed with transform-matrix rate overhead (i.e., at rates close to R_{\min} for data sets with spatial size relatively small with respect to spectral size), SPCA can significantly improve the performance. In both cases, the CC-based partition outperforms the uniform partition.

Fig. 5 shows the effect that an increasing number of PEs in a parallel processing system has on the SNR performance of non-SPCA. Since the spectral transform is not segmented, additional PEs therefore increase the degree of spatial partitioning. The original image size is $512 \times 512 \times 224$; after spatial partitioning, each subscene still has 224 bands but the spatial size is 256×256 (4 PEs), 128×128 (16 PEs), or 64×64 (64 PEs). In all cases, the rate is fixed at 0.5 bpppb. In Fig. 5, it is shown that the SNR decreases slightly as the number of PEs increases from one to four, while further increase to 16 PEs entails a significant quality reduction. At 64 PEs, non-SPCA actually fails, since the spatial size of the subscenes is much smaller than the spectral

TABLE III
SNR PERFORMANCE IN DECIBELS

Algorithm	Bitrate		
	R_{\min}	$1.1R_{\min}$	$1.5R_{\min}$
Size: $512 \times 512 \times 224$; R_{\min} : 0.0273 bpppb			
SPCA-U-LBA	15.01	15.30	16.34
SPCA-U-GBA	15.60	15.98	17.52
SPCA-CC-LBA	15.27	15.52	16.30
SPCA-CC-GBA	16.36	16.83	18.64
PCA	5.45	12.55	16.08
DWT	13.85	14.10	14.95
Size: $256 \times 256 \times 224$; R_{\min} : 0.1094 bpppb			
SPCA-U-LBA	20.49	21.01	22.61
SPCA-U-GBA	24.33	25.21	27.82
SPCA-CC-LBA	19.07	19.33	20.34
SPCA-CC-GBA	25.54	26.45	28.99
PCA	7.49	16.24	23.96
DWT	20.58	21.09	23.05
Size: $128 \times 128 \times 224$; R_{\min} : 0.4375 bpppb			
SPCA-U-LBA	28.24	29.20	32.61
SPCA-U-GBA	34.77	35.63	38.25
SPCA-CC-LBA	24.60	25.15	27.75
SPCA-CC-GBA	35.50	36.27	38.69
PCA	9.86	21.67	33.28
DWT	31.06	31.84	34.40
Size: $256 \times 256 \times 112$; R_{\min} : 0.0547 bpppb			
SPCA-U-LBA	22.63	23.48	26.65
SPCA-U-GBA	24.37	25.52	28.60
SPCA-CC-LBA	21.97	23.50	27.65
SPCA-CC-GBA	25.65	26.91	30.21
PCA	7.97	16.95	26.03
DWT	21.38	21.91	23.78
Size: $128 \times 128 \times 112$; R_{\min} : 0.2188 bpppb			
SPCA-U-LBA	33.75	34.81	38.01
SPCA-U-GBA	34.93	35.94	39.16
SPCA-CC-LBA	33.94	35.08	38.12
SPCA-CC-GBA	35.75	36.77	39.91
PCA	11.85	23.86	36.29
DWT	32.46	33.21	35.78
Size: $128 \times 128 \times 56$; R_{\min} : 0.1094 bpppb			
SPCA-U-LBA	24.48	25.84	29.19
SPCA-U-GBA	29.03	30.29	33.67
SPCA-CC-LBA	24.48	25.84	29.19
SPCA-CC-GBA	29.03	30.29	33.67
PCA	12.53	15.25	27.16
DWT	29.67	30.57	32.84

size, causing the PCA overhead to overwhelm the bitstream and leave no bits available for coefficient encoding.

Fig. 6 shows that SPCA can counter the decreasing SNR due to increasing parallelism, as shown in Fig. 5. Specifically, starting with the 16-PE case (i.e., a subscene size of $128 \times 128 \times 224$), additional PEs are used to spectrally partition the subscenes by applying SPCA. Again, the rate is fixed at 0.5 bpppb. In Fig. 6, each $128 \times 128 \times 224$ subscene is spectrally partitioned into 2 $128 \times 128 \times 64$ subscenes (32 PEs), 4 $128 \times 128 \times 32$ subscenes (64 PEs), 8 $128 \times 128 \times 64$ subscenes (128 PEs), or 16 $128 \times 128 \times 32$ subscenes (256 PEs). The aforementioned subscene sizes are for the case of uniform spectral partitioning (i.e., SPCA-U). Fig. 6 also shows the performance for CC-based segmentation (SPCA-CC) which produces the same number of spectral partitions but of differing sizes. It is seen that the reconstruction quality increases dramatically once spectral partitioning is employed, since

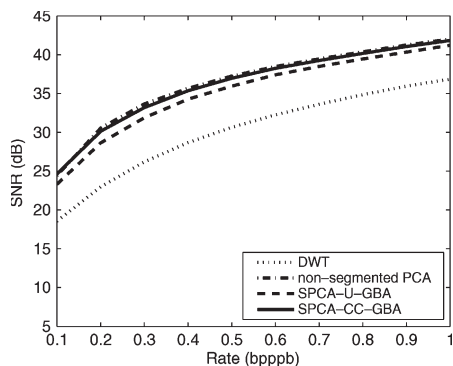


Fig. 3. Rate-distortion performance for the $512 \times 512 \times 224$ subscene.

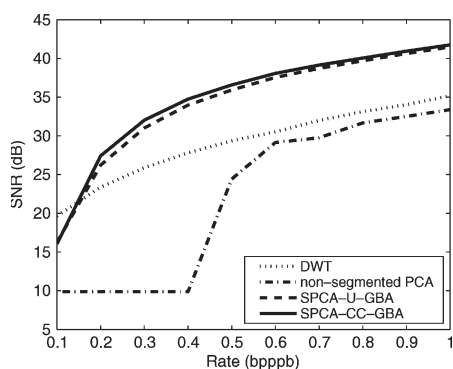


Fig. 4. Rate-distortion performance for the $128 \times 128 \times 224$ subscene.

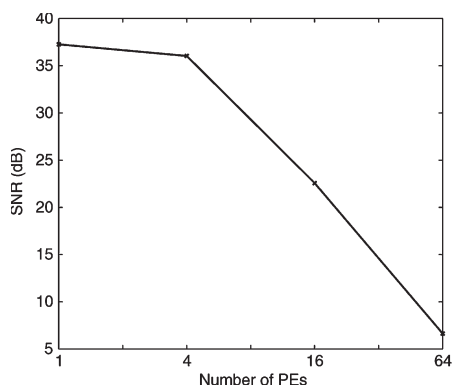


Fig. 5. SNR for non-SPCA as the number of PEs, and thereby the degree of spatial partitioning, increases. Data size before partitioning is $512 \times 512 \times 224$, and the bitrate is fixed at 0.5 bpppb.

the PCA overhead decreases rapidly with decreasing spectral size. We note, however, that a maximum is obtained around 64–128 PEs, and this maximum is not quite as large as the peak for non-SPCA in Fig. 5 (i.e., for a single PE). This is due to the fact that increasing spectral partitioning decreases the quality of the spectral decorrelation of the SPCA transform with respect to that of non-SPCA, as aforementioned in Section II. Consequently, in Fig. 6, increasing spectral partitioning initially increases SNR significantly, since the decrease in PCA overhead outweighs the increasing inefficiency in SPCA decorrelation. For much larger degrees of parallelism with very small spectral sizes, the decreased decorrelation efficiency, however, becomes increasingly important.

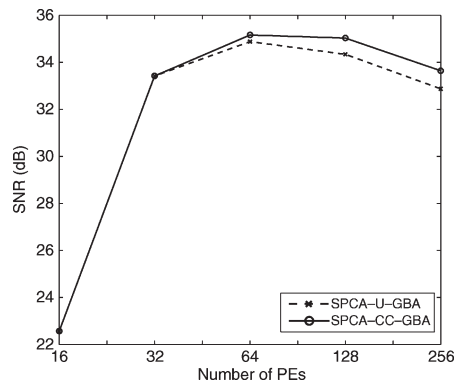


Fig. 6. SNR for SPCA as the number of PEs, and thereby the degree of spectral partitioning, increases. Data size before partitioning is $512 \times 512 \times 224$, and the bitrate is fixed at 0.5 bpppb.

V. CONCLUSION

In this letter, we have investigated the performance of JPEG2000 for the compression of hyperspectral imagery when PCA is used for spectral decorrelation. Due to the data-dependent nature of PCA, the JPEG2000 encoder stores the PCA transform matrix in the compressed bitstream—this overhead may be negligible when the spatial size of the hyperspectral image is large relative to the number of spectral bands. However, even if the hyperspectral image is very large spatially, partitioning the data set into smaller subscenes for distribution to and compression by independent processing nodes may result in a given node facing an image subscene with small spatial size. PCA-based compression will be detrimentally affected in such a case. However, segmenting the spectral bands into multiple groups in which PCA is performed independently can dramatically reduce the transform-matrix overhead. Our results have indicated that such SPCA, when used in conjunction with global JPEG2000 bit allocation, is largely immune to transform-matrix effects and can significantly outperform a DWT-based spectral transform which incurs no such overhead. Additionally, spectral segmentation based on the correlation coefficient outperforms straightforward uniform segmentation.

REFERENCES

- [1] B. Penna, T. Tillo, E. Magli, and G. Olmo, "Progressive 3-D coding of hyperspectral images based on JPEG 2000," *IEEE Geosci. Remote Sens. Lett.*, vol. 3, no. 1, pp. 125–129, Jan. 2006.
- [2] 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.
- [3] Q. Du and J. E. Fowler, "Hyperspectral image compression using JPEG2000 and principal component analysis," *IEEE Geosci. Remote Sens. Lett.*, vol. 4, no. 2, pp. 201–205, Apr. 2007.
- [4] Information Technology—JPEG 2000 Image Coding System—Part 2: Extensions, ISO/IEC 15444-2, 2004.
- [5] H. Yang, Q. Du, W. Zhu, I. Banicescu, and J. E. Fowler, "Parallel data compression for hyperspectral imagery," in *Proc. Int. Geosci. Remote Sens. Symp.*, Boston, MA, Jul. 2008, vol. 2, pp. 986–989.
- [6] X. Jia and J. A. Richards, "Segmented principal components transformation for efficient hyperspectral remote-sensing image display and classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 37, no. 1, pp. 538–542, Jan. 1999.
- [7] Q. Du, J. E. Fowler, and W. Zhu, "On the impact of atmospheric correction on lossy compression of multispectral and hyperspectral imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 47, no. 1, pp. 130–132, Jan. 2009.
- [8] Q. Du and J. E. Fowler, "Low-complexity principal component analysis for hyperspectral image compression," *Int. J. High Perform. Comput. Appl.*, vol. 22, no. 4, pp. 438–448, Nov. 2008.