

Parallel Implementation of Linear and Nonlinear Spectral Unmixing of Remotely Sensed Hyperspectral Images

Antonio Plaza^a and Javier Plaza^a

^aHyperspectral Computing Laboratory
Department of Technology of Computers and Communications
University of Extremadura, Avda. de la Universidad s/n
10071 Cáceres, Spain

ABSTRACT

Hyperspectral unmixing is a very important task for remotely sensed hyperspectral data exploitation. It addresses the (possibly) mixed nature of pixels collected by instruments for Earth observation, which are due to several phenomena including limited spatial resolution, presence of mixing effects at different scales, etc. Spectral unmixing involves the separation of a mixed pixel spectrum into its pure component spectra (called endmembers) and the estimation of the proportion (abundance) of endmember in the pixel. Two models have been widely used in the literature in order to address the mixture problem in hyperspectral data. The linear model assumes that the endmember substances are sitting side-by-side within the field of view of the imaging instrument. On the other hand, the nonlinear mixture model assumes nonlinear interactions between endmember substances. Both techniques can be computationally expensive, in particular, for high-dimensional hyperspectral data sets. In this paper, we develop and compare parallel implementations of linear and nonlinear unmixing techniques for remotely sensed hyperspectral data. For the linear model, we adopt a parallel unsupervised processing chain made up of two steps: i) identification of pure spectral materials or endmembers, and ii) estimation of the abundance of each endmember in each pixel of the scene. For the nonlinear model, we adopt a supervised procedure based on the training of a parallel multi-layer perceptron neural network using intelligently selected training samples also derived in parallel fashion. The compared techniques are experimentally validated using hyperspectral data collected at different altitudes over a so-called Dehesa (semi-arid environment) in Extremadura, Spain, and evaluated in terms of computational performance using high performance computing systems such as commodity Beowulf clusters.

Keywords: Hyperspectral imaging, high performance computing, spectral unmixing, linear spectral unmixing, nonlinear spectral unmixing.

1. INTRODUCTION

Spectral mixture analysis (also called *spectral unmixing*) has been an alluring exploitation goal from the earliest days of hyperspectral imaging¹ to our days.^{2,3} No matter the spatial resolution, the spectral signatures collected in natural environments are invariably a mixture of the signatures of the various materials found within the spatial extent of the ground instantaneous field view of the imaging instrument.⁴ The availability of hyperspectral imagers with a number of spectral bands that exceeds the number of spectral mixture components⁵ has allowed to cast the unmixing problem in terms of an over-determined system of equations in which, given a set of pure spectral signatures (called *endmembers*) the actual unmixing to determine apparent pixel *abundance fractions* can be defined in terms of a numerical inversion process.⁶

A standard technique for spectral mixture analysis is *linear* spectral unmixing,^{7,8} which assumes that the collected spectra at the spectrometer can be expressed in the form of a linear combination of endmembers weighted by their corresponding abundances. It should be noted that the linear mixture model assumes minimal secondary reflections and/or multiple scattering effects in the data collection procedure, and hence the measured spectra can be expressed as a linear combination of the spectral signatures of materials present in the mixed

Send correspondence to Antonio J. Plaza:

E-mail: aplaza@unex.es; Telephone: +34 927 257000 (Ext. 51662); URL: <http://www.umbc.edu/rssipl/people/aplaza>

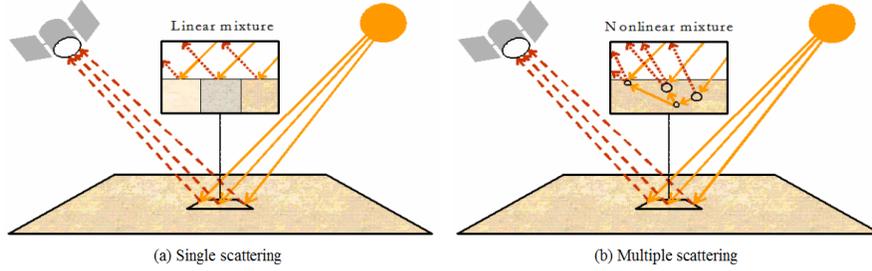


Figure 1. Linear versus nonlinear mixture models: single versus multiple scattering.

pixel [see Fig. 1(a)]. Although the linear model has practical advantages such as ease of implementation and flexibility in different applications,⁹ *nonlinear* spectral unmixing may best characterize the resultant mixed spectra for certain endmember distributions, such as those in which the endmember components are randomly distributed throughout the field of view of the instrument.^{10, 11} In those cases, the mixed spectra collected at the imaging instrument is better described by assuming that part of the source radiation is multiply scattered before being collected at the sensor [see Fig. 1(b)]. The unmixing process is quite computationally expensive, due to the extremely high dimensionality of hyperspectral data cubes, although it has been shown to map nicely to high performance computing systems such as commodity clusters of computers.^{12–16}

In this paper, we develop and compare different cluster-based parallel implementations of linear and nonlinear techniques for spectral unmixing of remotely sensed hyperspectral data. The remainder of the paper is organized as follows. Section 2 briefly outlines the differences between the linear and the nonlinear unmixing techniques adopted in this work. For the linear model, we adopt a parallel unsupervised processing chain made up of three steps: i) automatic identification of endmembers, and ii) estimation of the abundance of each endmember in each pixel of the scene. For the nonlinear model, we adopt a supervised procedure based on the training of a parallel multi-layer perceptron neural network using intelligently selected training samples also derived in parallel fashion. Section 3 presents parallel implementations of the considered linear and nonlinear unmixing models. Section 4 provides an experimental validation of the proposed parallel techniques using hyperspectral data collected at different altitudes over a so-called Dehesa (semi-arid environment) in Extremadura, Spain. The computational performance of the discussed techniques is evaluated using Thunderhead, a Beowulf cluster available at NASA’s Goddard Space Flight Center in Maryland. Section 5 concludes the paper with some remarks and hints at plausible future research lines.

2. LINEAR VERSUS NONLINEAR SPECTRAL UNMIXING

2.1 Linear spectral unmixing

Let us assume that a remotely sensed hyperspectral scene with n bands is denoted by \mathbf{I} , in which a certain pixel of the scene is represented by a vector $\mathbf{x} = [x_1, x_2, \dots, x_n] \in \mathfrak{R}^n$, where \mathfrak{R} denotes the set of real numbers in which the pixel’s spectral response x_k at sensor channels $k = 1, \dots, n$ is included. Under the linear mixture model assumption, each pixel vector in the original scene can be modeled using the following expression:

$$\mathbf{x} = \sum_{z=1}^p \Phi_z \cdot \mathbf{E}_z + \mathbf{n}, \quad (1)$$

where \mathbf{E}_z denotes the spectral response of endmember z , Φ_z is a scalar value designating the fractional abundance of the endmember z at the pixel \mathbf{x} , p is the total number of endmembers, and \mathbf{n} is a noise vector. Two physical constrains are generally imposed into the model described in (1), these are the abundance non-negativity constraint (ANC), i.e., $\Phi_z \geq 0$, and the abundance sum-to-one constraint (ASC), i.e., $\sum_{z=1}^p \Phi_z = 1$.⁷ The solution of such fully constrained linear spectral unmixing (FCLSU) described in (1) relies on the correct determination of a set $\mathbf{E} = \{\mathbf{E}_z\}_{z=1}^p$ of endmembers.¹⁷

Over the last years, many techniques have been developed for automatic or semi-automatic extraction of spectral endmembers.^{8, 18} In this work, we focus on a classic technique called orthogonal subspace projection

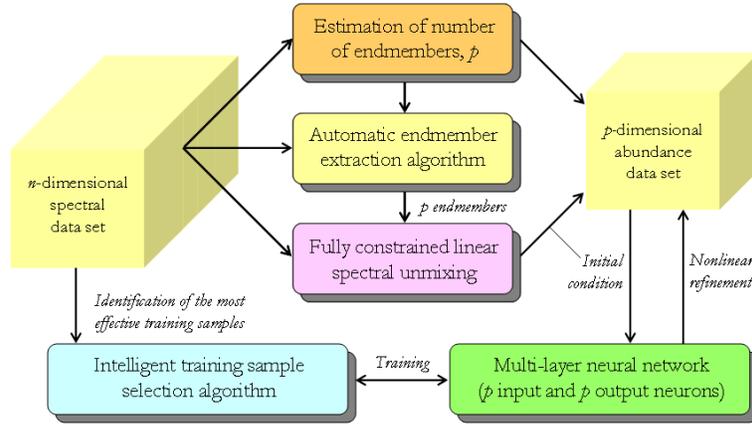


Figure 2. Neural network-based spectral unmixing architecture.

(OSP).¹⁹ This algorithm starts by selecting the pixel vector with maximum length in the scene as the first endmember. Then, it looks for the pixel vector with the maximum absolute projection in the space orthogonal to the space linearly spanned by the initial pixel, and labels that pixel as the second endmember. A third endmember is found by applying an orthogonal subspace projector to the original hyperspectral image, where the signature that has the maximum orthogonal projection in the space orthogonal to the space linearly spanned by the first two endmembers. This procedure is repeated until the desired number of endmembers, p , is found.²⁰

2.2 Nonlinear spectral unmixing

Under the nonlinear mixture model assumption, each pixel vector in the original scene can be modeled using the following expression:

$$\mathbf{x} = f(\mathbf{E}, \Phi) + \mathbf{n}, \quad (2)$$

where f is an unknown nonlinear function that defines the interaction between \mathbf{E} and Φ . Various learning-from-data techniques have been proposed in the literature to estimate f . In particular, artificial neural networks have demonstrated great potential to decompose mixed pixels due to their inherent capacity to approximate complex functions.²¹ Although many neural network architectures exist, for decomposition of mixed pixels in terms of nonlinear relationships mostly feed-forward networks of various layers, such as the multi-layer perceptron (MLP), have been used.^{11, 22, 23} It has been shown in the literature that MLP-based neural models, when trained accordingly, generally outperform other nonlinear models such as regression trees or fuzzy classifiers.²⁴

Fig. 2 shows a schematic block diagram of the MLP neural network-based unmixing architecture considered in this work. The first step consists of an estimation of the number of endmembers, p , in the input data. For this purpose, in this work we use the hyperspectral signal identification by minimum error (HySime).²⁵ Then, the model is initialized via FCLSU based on automatic endmember extraction using the OSP method.¹⁹ Finally, the model is refined by a supervised MLP neural network. The latter step is supported by an unsupervised algorithm for intelligent selection of training samples (both pure and mixed) from the data in order to estimate the final endmember fractional abundances. The number of neurons at the input layer of the MLP architecture equals the number of spectral endmembers found in the initialization stage. The input patterns to the input layer are vectors of endmember fractional abundances for each sample vector \mathbf{x} , first estimated by fully constrained unmixing. The second layer is the hidden layer, and the third layer is the output layer. The number of neurons at the output layer, p , equals the number of estimated endmembers. Based on previous results in the literature and our own experimentation, we have considered one hidden layer only, with the number of neurons empirically set to the square root of the product of the number of input features and information classes, a configuration that has been shown to be successful for MLP-based mixed pixel characterization in previous work.²⁴

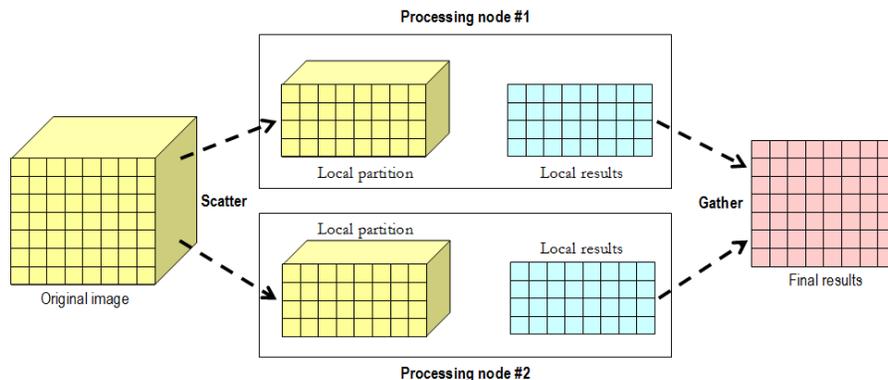


Figure 3. Spatial-domain decomposition for parallel implementation of the linear spectral unmixing chain.

3. PARALLEL IMPLEMENTATIONS

3.1 Parallel implementation of linear spectral unmixing

In this subsection, we describe a master-slave parallel version of linear spectral unmixing which maps effectively to parallel computer systems such as clusters. To reduce code redundancy and enhance reusability, our goal was to reuse much of the code for the sequential algorithm in the parallel implementation. For that purpose, we adopted a spatial-domain decomposition approach that subdivides the image cube into multiple blocks made up of entire pixel vectors, and assigns one or more blocks to each processing element (see Fig. 3). The processing elements apply the linear unmixing chain locally and then the master gathers the partial results provided by the workers and produces a final result. While the FCLSU technique for abundance estimation is pixel-based and can be executed in parallel without further communications, a parallel version of OSP requires more inter-processor communications.²⁶ It should be noted that a spectral-domain partitioning scheme (which subdivides the whole multi-band data into blocks made up of contiguous spectral bands or sub-volumes, and assigns one or more sub-volumes to each processing element), is not appropriate in our application as it breaks the spectral identity of the data because each pixel vector is split amongst several processing element. A further reason that justifies the above decision is that, in spectral-domain partitioning, the calculations made for each hyperspectral pixel need to originate from several processing elements, and thus require intensive inter-processor communication.

The proposed parallel version has been implemented in the C++ programming language, using calls to message passing interface (MPI)*. We emphasize that, in order to implement step one of the parallel algorithm, we resorted to MPI *derived datatypes* to directly scatter hyperspectral data structures, which may be stored non-contiguously in memory, in a single communication step. As a result, we avoid creating all partial data structures on the master node (thus making a better use of memory resources and compute power).

3.2 Parallel implementation of nonlinear spectral unmixing

Contrary to the linear spectral unmixing chain, several partitioning schemes can be analyzed when mapping MLP-based nonlinear unmixing onto a cluster architecture using MPI.²⁷ The choice is application-dependent, and a key issue is the number of training patterns to be used during the learning stage. In the following, we present two different schemes for the MLP neural network partitioning that will be compared in this work:

1. **Exemplar partitioning.** This approach, also called training example parallelism, explores data level parallelism and can be easily obtained by simply partitioning the training pattern data set (see Fig. 4). Each process determines the weight changes for a disjoint subset of the training population and then changes are combined and applied to the neural network at the end of each epoch. As will be shown by experiments, this scheme requires a large number of training patterns to produce significant speedups, which is the most common situation in most remote sensing applications due to the limited availability

*<http://www.mcs.anl.gov/research/projects/mpich2/>

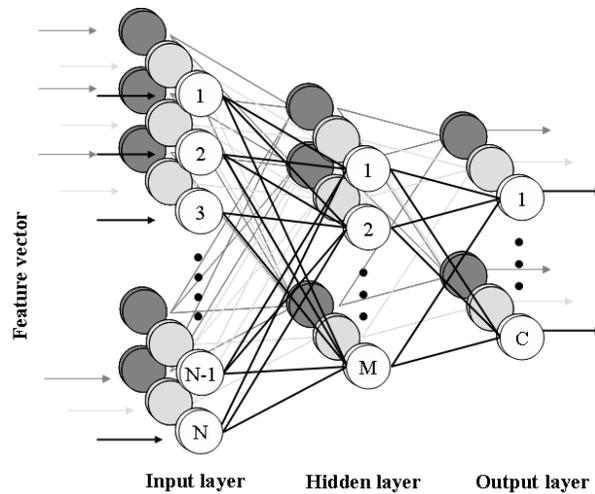


Figure 4. Exemplar partitioning scheme over MLP topology. Training data is divided into different subsets which are used to train three different subnetworks, i.e., white, grey and black.

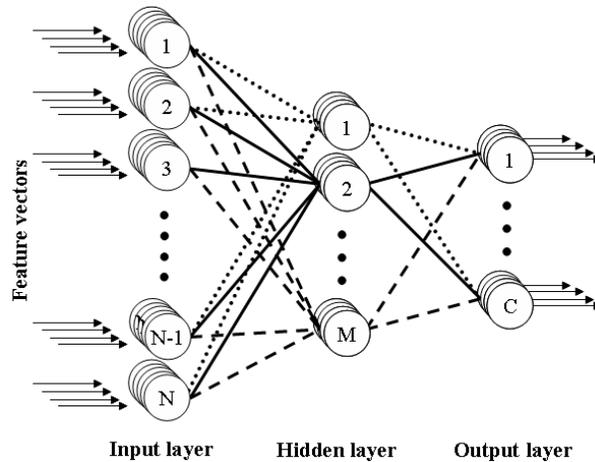


Figure 5. Hybrid partitioning scheme over MLP topology. The input and output layers are common to all processors. The hidden nodes are distributed among different processors (lines, dotted-lines and dashed-lines denote weight connections corresponding to three different processors).

of training samples and the great difficulty to generate accurately labeled ground-truth samples prior to analyzing the collected data.

2. **Hybrid partitioning.** This approach is based on a combination of neuronal level as well as synaptic level parallelism²⁸ which allows us to reduce the processor intercommunications at each iteration (see Fig. 5). This approach results from a combination of neuronal level parallelism and synaptic level parallelism. In the former (also called vertical partitioning) all the incoming weights to the neurons local to the processor are computed by a single processor. In the latter, each workstation will compute only the outgoing weight connections of the nodes (neurons) local to the processor. The hybrid scheme combines those approaches, i.e., the hidden layer is partitioned using neuronal parallelism while weight connections adopt synaptic scheme. As a result, all inter-processor communications will be reduced to a cumulative sum at each epoch, thus significantly reducing processing time on parallel platforms.



Figure 6. Full flightline of a ROSIS hyperspectral scene over a Dehesa area in Cáceres, Extremadura, Spain.

4. EXPERIMENTAL RESULTS

4.1 Hyperspectral data used in experiments

The hyperspectral data set used in our experiments consisted of two main components: image data and field measurements of land-cover fractions, collected at the time of image data acquisition. The image data is formed by a scene collected at high spatial resolution by the reflective optics systems imaging spectrographic system (ROSIS), with 1.2-meter pixels, and its corresponding digital airborne imaging spectrometer (DAIS) 7915 scene, collected at low spatial resolution with 6-meter pixels. Both instruments were managed by the German Aerospace Agency (DLR), and operated simultaneously at multiple resolutions. The spectral range from 504 to 864 nm was selected for experiments, not only because it is adequate for analyzing the spectral properties of the landscape under study, but also because this spectral range is well covered by the two considered sensors through narrow spectral bands. Fig. 6 shows the full flightline of the ROSIS scene, which comprises a Dehesa area located between the facilities of University of Extremadura in Cáceres (leftmost part of the flightline) and Guadiloba water reservoir at the center of the flightline. Fig. 7(a) shows the Dehesa test site selected for experiments, which corresponds to a highly representative Dehesa area that contains several cork-oak trees (appearing as dark spots) and several pasture (gray) areas on a bare soil (white) background. Several field techniques were applied to obtain reliable estimates of fractional land cover for each DAIS 7915 pixel in the considered Dehesa test site:

1. First, the ROSIS image was roughly classified into the three land-cover components above using a maximum-likelihood supervised classification approach based on image-derived spectral endmembers, where Fig. 7(b) shows the three endmembers used for mapping that were derived using the OSP algorithm. Our assumption was that the pixels in the ROSIS image were sufficiently small to become spectrally simple to analyze.
2. Then, the classified ROSIS image was registered with the DAIS 7915 image using a ground control point-based method with sub-pixel accuracy.²⁹
3. The classification map was then associated with the DAIS 7915 image to provide an initial estimation of land cover classes for each pixel at the DAIS 7915 image scale. For that purpose, a 6x6-meter grid was overlaid on the 1.2×1.2 -meter classification map derived from the ROSIS scene, where the geographic coordinates of each pixel center point were used to validate the registration with sub-pixel precision.

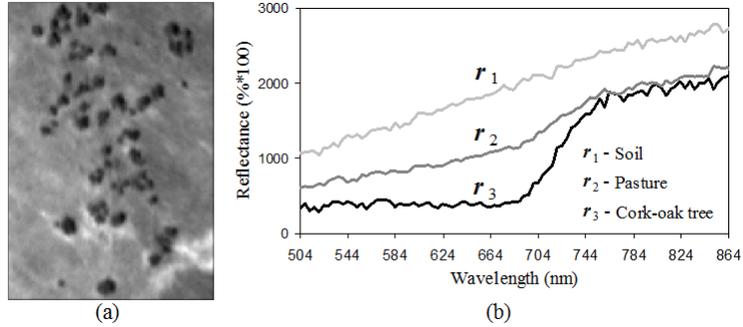


Figure 7. (a) Spectral band (584 nm) of a ROSIS Dehesa subset selected for experiments. (b) Endmember signatures of soil, pasture and cork-oak tree extracted by the OSP algorithm, where scaled reflectance values are multiplied by a constant factor.

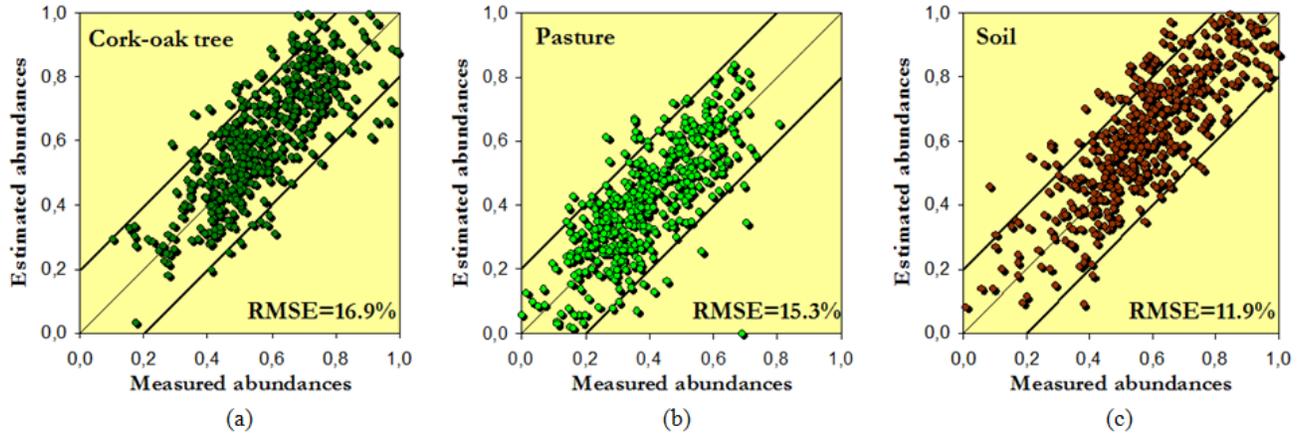


Figure 8. Abundance estimations of cork-oak tree (a), pasture (b) and soil (c) by fully constrained linear spectral unmixing (FCLSU) from the DAIS 7915 image.

4. Next, fractional abundances were calculated within each 6x6-meter grid as the proportion of ROSIS pixels labeled as cork-oak tree, pasture and soil located within that grid, respectively.
5. Most importantly, the abundance maps at the ROSIS level were thoroughly refined using field measurements before obtaining the final proportions.

4.2 Analysis of unmixing accuracy

In order to evaluate the accuracy of linear mixture modeling in the considered application, Fig. 8 shows the scatterplots of measured versus FCLSU-estimated fractional abundances for the three considered land-cover materials in the DAIS 7915 (low spatial resolution) image data set, where the diagonal represents perfect match and the two flanking lines represent plus/minus 20% error bound. Here, the three spectral endmembers were derived using the OSP algorithm. As expected, the flatness of the test site largely removed topographic influences in the remotely sensed response of soil areas. As a result, most linear predictions for the soil endmember fall within the 20% error bound [see Fig. 8(a)]. On the other hand, the multiple scattering within the pasture and cork-oak tree canopies (and from the underlying surface in the latter case) complicated the spectral mixing in nonlinear fashion, which resulted in a generally higher number of estimations lying outside the error bound, as illustrated in Figs. 8(b) and 8(c). Also, the root mean square error (RMSE) scores in abundance estimation for the soil (11.9%), pasture (15.3%) and cork-oak tree (16.9%) were all above 10% estimation error in percentage, which suggested that linear mixture modeling was not flexible enough to accommodate the full range of spectral variability throughout the landscape.

In order to characterize the Dehesa ecosystem structure better than linear models do, we hypothesized that intelligently selected training data might be required to better characterize nonlinear mixing effects. For this

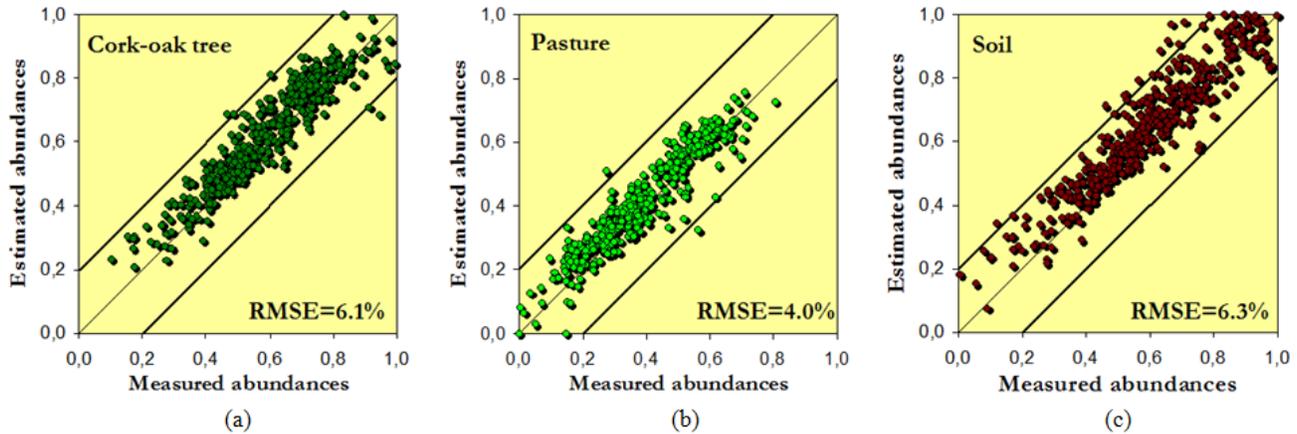


Figure 9. Abundance estimations of cork-oak tree (a), pasture (b) and soil (c) by the MLP-based mixture model (trained using automatically derived training samples) from the DAIS 7915 image.

purpose, we applied an automatic algorithm to automatically locate highly descriptive training sites in the DAIS 7915 scene²³ and then used the obtained samples (and the ground-truth information associated to those samples) to train the proposed MLP-based neural network. Fig. 9 shows the scatter plots of measured versus predicted fractional abundances for soil, pasture and cork-oak tree by the proposed MLP-based model, trained with the three endmembers derived by OSP [see Fig. 7(b)] plus 40 additional training samples selected by an automatic training sample selection algorithm, which represent less than 1% of the total number of pixels in the DAIS 7915 scene. These samples were excluded from the testing set made up of all remaining pixels in the scene. From Fig. 9, it is clear that the utilization of intelligently selected training samples resulted in fewer points outside the two 20% difference lines, most notably, for both pasture and cork-oak abundance estimates. The pattern of the scatter plots obtained for the soil predictions [see Fig. 9(a)] was similar (in particular, when the soil abundance was high). Most importantly, the RMSE scores in abundance estimation were significantly reduced (with regards to the experiment using FCLSU) for the soil (6.1%), pasture (4%) and cork-oak tree (6.3%). These results confirm our intuition that nonlinear effects in Dehesa landscapes mainly result from multiple scattering effects in vegetation canopies and can be better modeled using a neural network-based approach which accounts for such nonlinearities at the unmixing stage.

4.3 Analysis of parallel performance

In order to analyze the scalability of the parallel algorithms we use Thunderhead, a massively parallel Beowulf cluster at NASA's Goddard Space Flight Center in Maryland. It consists of 268 dual 2.4 Ghz Intel 4 Xeon nodes, each with 1 GB of memory and 80 GB of hard disk[†]. The total disk space available in the system is 21.44 Tbyte, and the theoretical peak performance of the system is 2.5728 Tflops (1.2 Tflops on the Linpack benchmark). Along with the 568-processor computer core, Thunderhead has several nodes attached to the core with Myrinet 2000 connectivity. Our parallel algorithms were run from one of such nodes, called thunder1. The operating system is Linux Fedora Core, and MPICH was the message-passing library used[‡].

Table 1 reports the measured execution times achieved by all tested algorithms after processing the Dehesa hyperspectral data set on Thunderhead, using different numbers of processors. The parallel times reported on Table 1 reveal that the combination of OSP for endmember extraction, followed by FCLSU for abundance estimation, is able to provide linear spectral unmixing results in only 22 seconds for the maximum number of processors available in Thunderhead. Then, the application of parallel identification of training samples followed by a hybrid MLP for nonlinear unmixing (initialized with FCLSU) is able to provide highly accurate unmixing results, taking into account potential nonlinearities, in only 7 additional seconds using all processing units available in Thunderhead. In this regard, the measured processing times represent a significant improvement

[†]<http://thunderhead.gsfc.nasa.gov>

[‡]<http://www-unix.mcs.anl.gov/mpi/mpich>

Table 1. Processing times in seconds and speedups (in the parentheses) achieved by multi-processor runs of the considered parallel algorithms on the Thunderhead Beowulf cluster at NASA’s Goddard Space Flight Center.

	Number of processors							
	4	16	36	64	100	144	196	256
OSP	797 (2.5)	203 (10.0)	79 (25.8)	39 (52.3)	23 (88.73)	17 (120.0)	13 (157.0)	10 (204.1)
FCLSU	826 (2.4)	215 (9.5)	88 (23.3)	45 (45.7)	27 (76.2)	20 (102.8)	16 (128.5)	12 (171.5)
	2	4	8	16	32	64	128	256
MLP (Exemplar)	1041 (1.9)	414 (4.8)	248 (8.1)	174 (11.5)	142 (14.1)	99 (20.2)	120 (16.7)	120 (16.7)
MLP (Hybrid)	973 (1.6)	458 (3.5)	222 (7.2)	114 (14.0)	55 (29.2)	27 (59.5)	15 (107.1)	7 (229.5)

over commonly used processing strategies for this kind of high-dimensional data sets, which can take up to more than 40 minutes of computation for the considered problem size as evidenced by the sequential computation times reported also in Table 1.

Taking into account the results presented above, and with the ultimate goal of exploring issues of scalability (considered to be a highly desirable property in the design of parallel algorithms), Table 1 also includes the speedups achieved by multi-processor runs of the developed parallel algorithms over the corresponding single-processor runs of each considered algorithm on Thunderhead. As shown in Table 1, our parallel versions of linear spectral unmixing achieve great parallel efficiency on Thunderhead. As for the parallel strategies used to implement MLP, the scalability of the hybrid approach is very close to linear. This is because this strategy relies on the distribution of weight connections among different processors, which does not affect the learning algorithm used by the MLP. However, the speedup of exemplar parallel method saturates when the number of processors increases, as indicated in Table 1. This is mainly due to the limited number of training patterns used during the training stage. It should be noticed that the learning process in the neural network is strongly dependent of the number of selected training samples. Since the exemplar method divides the total training pattern set into as many subsets as processors available in the parallel system, each local processing will likely have different local minima and convergence challenges, thus introducing load imbalance problems. According to our experimentation, the exemplar partitioning approach is suitable for applications in which the number of available training patterns is very high, which is often not the case in the context of remote sensing applications.

5. CONCLUSIONS AND FUTURE RESEARCH LINES

Spectral unmixing has been a very active area in hyperspectral data exploitation. Although linear spectral unmixing can provide accurate results from a macroscopic point of view, nonlinear mixture models may best characterize the hyperspectral data when complex mixtures are present. In order to address the computational complexity of linear and nonlinear unmixing models, in this paper we have developed and compared different parallel implementations of linear and nonlinear unmixing techniques for remotely sensed hyperspectral data. For the linear model we considered the general case in which an endmember extraction algorithm is followed by a linear inversion process for abundance estimation. The latter is pixel-based and can be executed very effectively in parallel, while the endmember extraction stage generally requires more inter-processor communications. For the nonlinear model we considered a multi-layer perceptron neural network, which can be implemented very effectively in parallel using a hybrid approach. Our experimental results suggest the potential of both linear and nonlinear parallel unmixing techniques for hyperspectral data exploitation on a Beowulf cluster of computers at NASA’s Goddard Space Flight Center in Maryland. The techniques are validated using hyperspectral data collected at multiple spatial resolutions over a Dehesa area in Extremadura, Spain. Future developments should focus on drawing comparisons of parallel implementations of linear versus nonlinear unmixing techniques in other types of high performance computing architectures, such as heterogeneous networks of (possibly distributed) workstations and specialized hardware devices such as field programmable gate arrays (FPGAs) or commodity graphics processing units (GPUs).

ACKNOWLEDGEMENT

This work has been supported by the European Community’s Marie Curie Research Training Networks Programme under reference MRTN-CT-2006-035927 (HYPER-I-NET). Funding from the Spanish Ministry of Science and Innovation (HYPERCOMP/EODIX project, reference AYA2008-05965-C04-02) and from Junta de Extremadura (PRI09A110 and GR10035 projects) are also very gratefully acknowledged.

REFERENCES

1. A. F. H. Goetz, G. Vane, J. E. Solomon, and B. N. Rock, "Imaging spectrometry for Earth remote sensing," *Science* **228**, pp. 1147–1153, 1985.
2. A. Plaza, J. A. Benediktsson, J. Boardman, J. Brazile, L. Bruzzone, G. Camps-Valls, J. Chanussot, M. Fauvel, P. Gamba, J. Gualtieri, M. Marconcini, J. C. Tilton, and G. Trianni, "Recent advances in techniques for hyperspectral image processing," *Remote Sensing of Environment* **113**, pp. 110–122, 2009.
3. M. E. Schaepman, S. L. Ustin, A. Plaza, T. H. Painter, J. Verrelst, and S. Liang, "Earth system science related imaging spectroscopy – an assessment," *Remote Sensing of Environment* **113**, pp. 123–137, 2009.
4. J. B. Adams, M. O. Smith, and P. E. Johnson, "Spectral mixture modeling: a new analysis of rock and soil types at the Viking Lander 1 site," *Journal of Geophysical Research* **91**, pp. 8098–8112, 1986.
5. R. O. Green, M. L. Eastwood, C. M. Sarture, T. G. Chrien, M. Aronsson, B. J. Chippendale, J. A. Faust, B. E. Pavri, C. J. Chovit, M. Solis, *et al.*, "Imaging spectroscopy and the airborne visible/infrared imaging spectrometer (AVIRIS)," *Remote Sensing of Environment* **65**(3), pp. 227–248, 1998.
6. N. Keshava and J. F. Mustard, "Spectral unmixing," *IEEE Signal Processing Magazine* **19**(1), pp. 44–57, 2002.
7. D. Heinz and C.-I. Chang, "Fully constrained least squares linear mixture analysis for material quantification in hyperspectral imagery," *IEEE Transactions on Geoscience and Remote Sensing* **39**, pp. 529–545, 2001.
8. A. Plaza, P. Martinez, R. Perez, and J. Plaza, "A quantitative and comparative analysis of endmember extraction algorithms from hyperspectral data," *IEEE Transactions on Geoscience and Remote Sensing* **42**(3), pp. 650–663, 2004.
9. C.-I. Chang, *Hyperspectral Imaging: Techniques for Spectral Detection and Classification*, Kluwer Academic/Plenum Publishers: New York, 2003.
10. K. J. Guilfoyle, M. L. Althouse, and C.-I. Chang, "A quantitative and comparative analysis of linear and nonlinear spectral mixture models using radial basis function neural networks," *IEEE Trans. Geosci. Remote Sens.* **39**, pp. 2314–2318, 2001.
11. J. Plaza, A. Plaza, R. Perez, and P. Martinez, "On the use of small training sets for neural network-based characterization of mixed pixels in remotely sensed hyperspectral images," *Pattern Recognition* **42**, pp. 3032–3045, 2009.
12. A. Plaza and C.-I. Chang, *High Performance Computing in Remote Sensing*, Taylor & Francis: Boca Raton, FL, 2007.
13. A. Plaza and C.-I. Chang, "Special issue on high performance computing for hyperspectral imaging," *International Journal of High Performance Computing Applications* **22**(4), pp. 363–365, 2008.
14. A. Plaza, "Special issue on architectures and techniques for real-time processing of remotely sensed images," *Journal of Real-Time Image Processing* **4**(3), pp. 191–193, 2009.
15. A. Plaza, Q. Du, Y.-L. Chang, and R. L. King, "High performance computing for hyperspectral remote sensing," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* **4**(3), 2011.
16. A. Plaza, J. Plaza, A. Paz, and S. Sanchez, "Parallel hyperspectral image and signal processing," *IEEE Signal Processing Magazine* **28**(3), pp. 119–126, 2011.
17. C.-I. Chang and D. Heinz, "Constrained subpixel target detection for remotely sensed imagery," *IEEE Transactions on Geoscience and Remote Sensing* **38**, pp. 1144–1159, 2000.
18. Q. Du, N. Raksuntorn, N. H. Younan, and R. L. King, "End-member extraction for hyperspectral image analysis," *Applied Optics* **47**, pp. 77–84, 2008.
19. J. C. Harsanyi and C.-I. Chang, "Hyperspectral image classification and dimensionality reduction: An orthogonal subspace projection," *IEEE Transactions on Geoscience and Remote Sensing* **32**(4), pp. 779–785.
20. H. Ren and C.-I. Chang, "Automatic spectral target recognition in hyperspectral imagery," *IEEE Transactions on Aerospace and Electronic Systems* **39**(4), pp. 1232–1249, 2003.
21. C. M. Bishop, *Neural networks for pattern recognition*, Oxford: Oxford University Press, 1995.
22. A. Baraldi, E. Binaghi, P. Blonda, P. A. Brivio, and P. Rampini, "Comparison of the multilayer perceptron with neuro-fuzzy techniques in the estimation of cover class mixture in remotely sensed data," *IEEE Trans. Geosci. Remote Sens.* **39**, pp. 994–1005, 2001.

23. J. Plaza and A. Plaza, "Spectral mixture analysis of hyperspectral scenes using intelligently selected training samples," *IEEE Geoscience and Remote Sensing Letters* **7**, pp. 371–375, 2010.
24. W. Liu and E. Y. Wu, "Comparison of non-linear mixture models," *Remote Sens. Environ.* **18**, pp. 1976–2003, 2004.
25. J. M. Bioucas-Dias and J. M. P. Nascimento, "Hyperspectral subspace identification," *IEEE Transactions on Geoscience and Remote Sensing* **46**(8), pp. 2435–2445, 2008.
26. A. Remon, S. Sanchez, A. Paz, E. S. Quintana-Orti, and A. Plaza, "Real-time endmember extraction on multi-core processors," *IEEE Geoscience and Remote Sensing Letters* **8**(5), pp. 924–928.
27. A. P. P. M. J. Plaza, R. Perez and D. Valencia, "Parallel morphological/neural processing of hyperspectral images using heterogeneous and homogeneous platforms," *Cluster Computing* **11**(1), pp. 17–32.
28. S. Suresh, S. N. Omkar, and V. Mani, "Parallel implementation of back-propagation algorithm in networks of workstations," *IEEE Transactions on Parallel and Distributed Systems* **16**(1), pp. 24–34.
29. A. Plaza, J. L. Moigne, and N. S. Netanyahu, "Morphological feature extraction for automatic registration of multispectral scenes," *Proceedings IEEE International Geoscience and Remote Sensing Symposium* **1**, pp. 421–424, 2007.