Interim Progress Report on the Application of an Independent Components Analysis-based Spectral Unmixing Algorithm to Beowulf Computers

By George Lemeshewsky[1]

Open-File Report 03-478

_____

[1]Eastern Region Geography, Reston, VA 20192

Interim progress report on the development of an independent-components-analysis-based spectral unmixing algorithm for Beowulf computers.

George Lemeshewsky
U.S. Geological Survey
Eastern Region Geography
Reston, VA 20192
glemeshe@usgs.gov

INTRODUCTION

This report describes work done to implement an independent-components-analysis (ICA) -based blind unmixing algorithm on the Eastern Region Geography (ERG) Beowulf computer cluster. It gives a brief description of blind spectral unmixing using ICA-based techniques and a preliminary example of unmixing results for Landsat-7 Thematic Mapper multispectral imagery using a recently reported[1,2,3] unmixing algorithm. Also included are computer performance data. The final phase of this work, the actual implementation of the unmixing algorithm on the Beowulf cluster, was not completed this fiscal year and is addressed elsewhere. It is noted that study of this algorithm and its application to land-cover mapping will continue under another research project in the *Land Remote Sensing* theme into fiscal year 2004.

PURPOSE OF WORK

The overall goal of this work was implementation of an ICA-based blind spectral unmixing algorithm, specifically in C code, on the Beowulf computer cluster. This work included choosing an unmixing algorithm that could be easily adapted to multicomputer processing, evaluating a MATLAB® version of this algorithm, translating MATLAB® code to C code, and measuring the performance of the compiled algorithm on a Linux-based personal computer (PC).

Specific objectives are as follows:

1. Identify an image processing algorithm for blind spectral unmixing.

2. Generate and test the algorithm as implemented in MATLAB® scientific programming language (M-code).

3. Translate the M-code algorithm to C code, for standalone installation on individual Beowulf PCs.

4. Measure performance of compiled C code on a PC having Linux operating system.

Some details of and progress toward each of these steps are provided in the next section.

WORK COMPLETED

This work included a review of several blind unmixing techniques and selection of an unmixing algorithm appropriate for parallel computation. The algorithm is based on a recently reported innovative technique with high relevance to unsupervised classification of multispectral and hyperspectral image data.[1,2,3]

1. Identify ICA blind unmixing algorithm

Given the model $\mathbf{X} = \mathbf{AS}$ for observed signal vector $\mathbf{X}$, ICA refers to a type of blind signal processing that consists of finding unknown vector $\mathbf{S}$ and matrix $\mathbf{W}$, where $\mathbf{S} = \mathbf{WX}$ and $\mathbf{W}=\mathbf{A}^{-1}$ such that the components $\{s_i\}$ of $\mathbf{S}$ are as independent as possible. It is assumed that the $\{s_i\}$ are statistically independent and have non-

Gaussian distributions. Thus **X** is a linear mixture the unknown components {s$_i$} combined by unknown mixing matrix **A**.

$$S \longrightarrow \boxed{A} \xrightarrow{X} \boxed{W=A^{-1}} \longrightarrow S$$
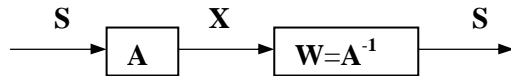
With imagery, vector **X** represents the observed multispectral pixel, **A** is the unknown linear mixing matrix, and vector **S** contains the unknown independent sources {s$_i$}. ICA-based techniques for solving this problem include the information-maximization approach of Bell-Sejnowski[4]. They use an artificial neural networklike learning approach where the neuron input weights (elements of **W**) are determined by an unsupervised learning process. These techniques have potential limitations when applied to large data files (e.g., imagery) because they use ensemble average statistics, although applied to individual pixel samples. Alternative methods have used image sub-areas (using a similar to tiling process). However there are potential problems with statistics based on smaller sample sizes.

More recently and with respect to multispectral unmixing, Szu[1,2,3] reported a new pixel-by-pixel independent classes analysis method that does not need ensemble statistics. This method combines artificial neural network unsupervised learning with Lagrangian constraint optimization methods to find unknown mixing matrix **A** and source vector **S** representing independent classes of source components {s$_i$}. (Their algorithm, somewhat similar to ICA, will be referred to here as ICA.) Their reported results to date with their MATLAB® coded algorithm for Thematic Mapper data unmixing are promising. One reason is that mixing conditions vary from pixel to pixel, and this algorithm unmixes each pixel independently.

This algorithm was chosen for implementation on the ERG Beowulf computers because it has a distinct advantage over other ICA methods: it processes each pixel independently. This approach simplifies implementation on a multicomputer Beowulf cluster. Increased computational performance is possible by merely dividing the data between computational nodes. There is no need for ensemble statistics based on image subregions (or tiles) because each pixel is processed independently.

2. Generate and test the algorithm

MATLAB® is a programming language for technical computing that includes an extensive library of mathematical functions and linear algebra tools for use with matrix-vector data types. Source code files (M-code) may be compiled into C object code for improved performance.

The ICA unmixing algorithm[4], implemented in M-code, was applied to seven-spectral-band Landsat 7 Enhanced Thematic Mapper Plus (ETM+) data (fig. 1). Figure 2 shows three of seven independent class components as a red, green, and blue band color composite. Figure 3 shows iterative unmixing of one pixel.

3. Translate MATLAB® M-code to C code, for standalone application

The ICA unmixing algorithm, in M-code, was translated to C source code by using the MATLAB® compiler[5] as part of another ongoing project to apply blind unmixing techniques to determine land cover from multispectral and hyperspectral data.

This compiler translates M-code functions to C source code, then compiles and links with MATLAB® math function libraries, and finally generates a standalone application as object code. Standalone applications do require the run-time shared MATLAB® library, but no additional licenses are needed for multiple PCs. For this preliminary evaluation, the compiled and linked executable was run from the MATLAB® command window.

4.   Measure performance data for PC having a Linux operating system.

MATLAB® was installed on a PC (Pentium 400 mHz, with 256 megabytes of RAM) operating under the RedHat® Linux 7.2 distribution, and the performance of two versions of the ICA-based blind-unmixing algorithm was compared. They were (1) MATLAB® M-code version and (2) compiled and linked C object code version. In each case, the 1024-sample, seven-spectral-band ETM+ data were in random access memory. This computation did not require disk accesses. The unmixing algorithm performed 150 iterations per pixel. Table 1 shows that the compiled version was about 2.65 times faster then the MATLAB® M-code version.

Version of ICA Code	CPU time, in seconds
MATLAB M-code	737.82
Compiled C	278.03

Table 1. Performance comparisons for implementations of the ICA unmixing algorithm.

RECOMMENDATION FOR FOLLOW-ON WORK

Because the ICA-based blind unmixing algorithm of Szu offers such a new and promising approach to the unsupervised processing of multispectral satellite data, this work will continue in fiscal year 2004. The unmixing algorithm will be converted for processing on the ERG Beowulf. Because binary file I/O programs (implemented in M-code) used to read image data already exist, the conversion for Beowulf processing can probably be completed in 2 work-weeks by a skilled C programmer having Linux experience.

REFERENCES

1. Szu, H., "A Priori MaxEnt H(S) independent class analysis (ica) vs. A Posteriori MaxEnt H(V) ICA", Third International Conference on Blind Signal Separation and ICA, San Diego, Dec., 2001, p. 80-89.

2. Szu, H. and H. Ren, "Unified Lagrangian Neural Network Method for Subpixel Classificationin Hyperspectral Imagery", Proc. of SPIE Vol. 4391, Wavelet Applications VIII, 2001, p. 314-329.

3. Szu, H. and I. Kopriva, "Constrained Equal A Priori Entropy for Unsupervised Remote Sensing," submitted to IEEE Trans. Geoscience and Remote Sensing, 2002.

4. Bell, A.J. and T.J. Sejnowski, "An information maximization approach to blind separation and blind deconvolution," Neural Computation v. 7, no. 6, p. 1129-1159, Nov. 1995.

5. MATLAB Compiler User's Guide, The MathWorks, Natick, MA, 2002.

Figure 1. Landsat 7 bands 5 4 2.

Shown are spectral bands 5 4 2 as red, green, blue color composite Landsat 7 data.
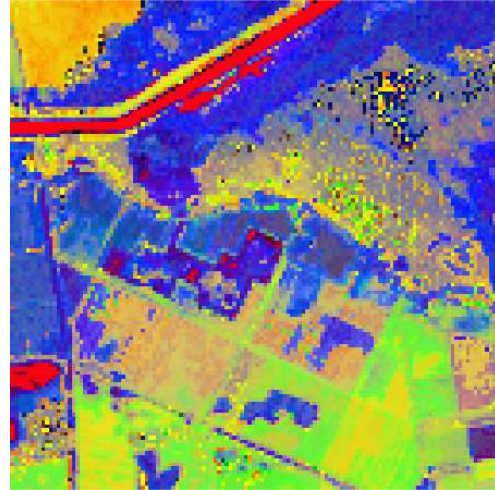

Figure 2. Independent components.

Three (of seven) independent source components computed from the seven band ETM+ image of fig. 2.
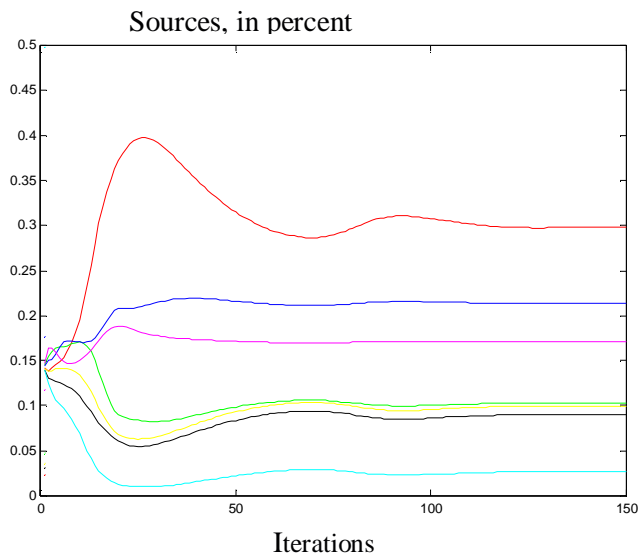

Sources, in percent

Iterations

Figure 3. Landsat 7 unmixing example.

This shows the iterative calculation of up to seven independent classes of sources, $\{s_i\}$ (values in percent) for a seven spectral band ETM+ pixel.