A FRAMEWORK FOR EFFICIENTLY PARALLELIZING NONLINEAR HYPERSPECTRAL NOISE REDUCTION

David G. Goodenough[1,2], *Tian Han*[1], *Belaid Moa*[2], *Amanpreet Dhaliwal*[2], *Hao Chen*[1], and *Ashlin Richardson*[1],

[1]Pacific Forestry Centre, Natural Resources Canada
[2]Department of Computer Science, University of Victoria
506 West Burnside Road, Victoria, BC, Canada, V8Z 1M5
(250-363-0776, office; 250-363-0775, fax; dgoodeno@nrcan.gc.ca)

ABSTRACT

Noise reduction is a necessary and common pre-processing step in remote sensing, especially hyperspectral image analysis. Until recently, hyperspectral data was modeled using linear stochastic processes and the noise was assumed to manifest itself in a narrow spatial frequency band. The signal and noise were thus considered independent and most of the proposed noise reduction algorithms, such as Maximum Noise Fraction (MNF) [14] and the Wavelet-based algorithms [15, 16], transformed linearly the hyperspectral data from one space to another for noise and signal separation. Hyperspectral data, however, exhibit nonlinear characteristics making the noise frequency and signal dependent [1, 2]. Therefore, to accurately reduce the noise in hyperspectral data, a nonlinear noise reduction algorithm was developed.

Techniques borrowed from the nonlinear and chaotic time series analysis [4, 17], especially Local Geometric Projection (LGP) [18], were adapted to develop a nonlinear denoising algorithm [1, 2, 6]. As Fig. 1 shows, this algorithm involves four main steps: 1) constructing state vectors in the phase space, 2) specifying the neighborhood of these state vectors, 3) finding projection directions and 4) reducing the noise by projecting the state vectors on these directions. The steps 2), 3) and 4) are executed until no further denoising is possible. The intuition behind LGP is that a nonlinear data series on a smooth manifold can be approximated linearly at each point on the manifold.

The nonlinear denoising algorithm was shown to achieve approximately 30% noise reduction and twice the SNR boost than the classical noise reduction algorithms. Moreover, the algorithm was able to maintain the spectral absorption features while effectively reducing noise. Fig.2 shows the original, denoised and difference images of an AVIRIS image (R = band 104 (1503 nm), G = band 35 (750 nm), and B = band 24 (645 nm)).

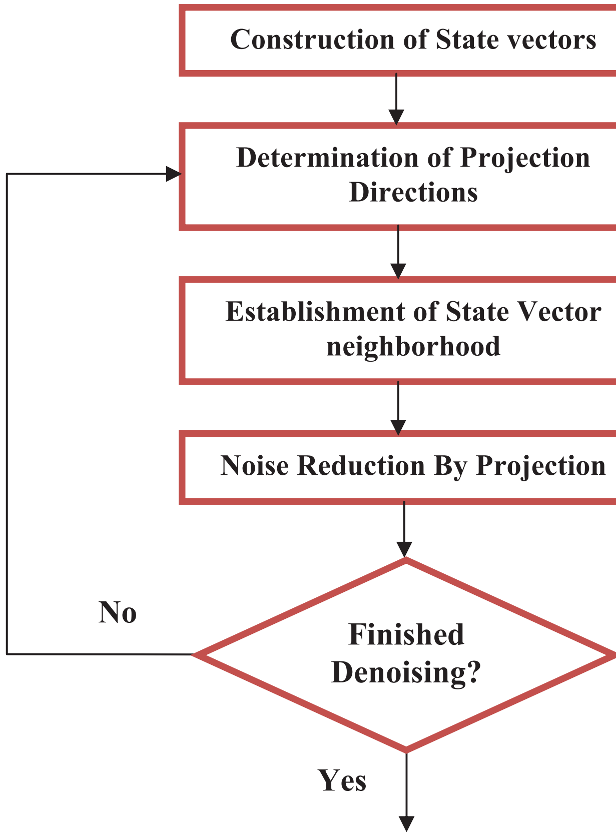

Figure 1: The main steps involved in the nonlinear denoising algorithm

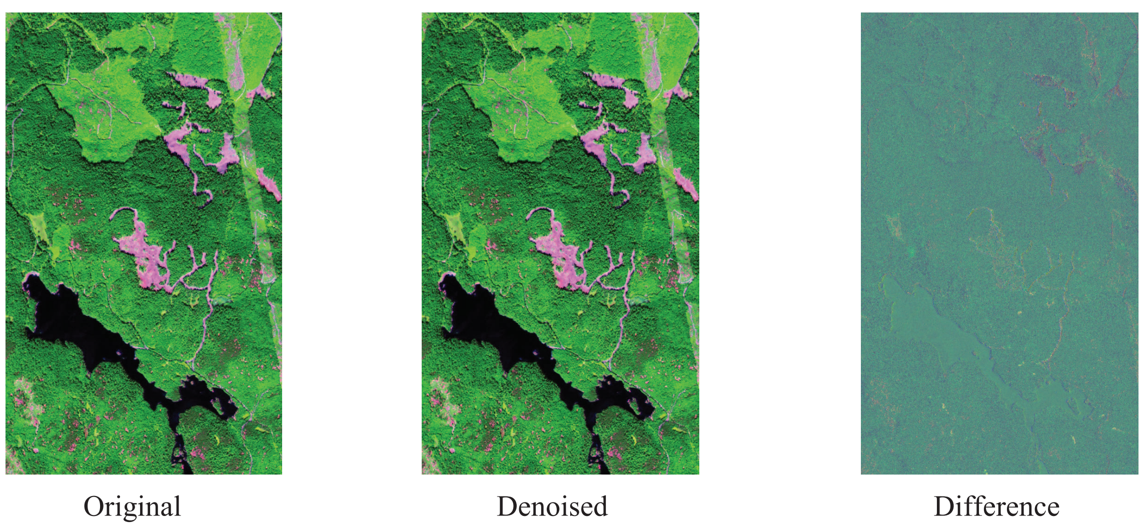

Figure 2: Original image, after denoising, and difference image (denoised - original) (AVIRIS images with R = 1503 nm, G = 750 nm, and B = 645 nm). These results were obtained with the nonlinear denoising algorithm.

Although the nonlinear denoising algorithm demonstrates impressive noise reduction results, it is very computationally expensive. The iteration over the three steps 2), 3) and 4) can take hours or even days to complete. In fact, our two implementations of the nonlinear noise reduction, including our C++ TISEAN-based implementation [7], took five days to denoise a 204MB hyperspectral image. To improve the efficiency of the algorithms, we introduced a framework that reduced its computational time. This framework did not require changing the existing code. The framework followed the Single Instruction Multiple Data streams (data parallelization) approach and required only a pool of computational resources accessible through a job scheduler. The job scheduler was a cluster scheduler such as PBS (Portable Batch System) [3] or Condor [11] or a metascheduler such as GridWay [12] and Gavia [8-10]. The computational resources could reside in a grid or cloud environment as long as a job scheduler that submitted jobs to those resources was available. As a testing environment, a production cluster composed of 210 IBM blade servers each with a dual Intel or AMD processors was setup at the University of Victoria. The cluster has a PBS queue and is grid-enabled. A cloud environment will be added to the cluster soon. This will allow us to test our framework in different environments.

The framework uses a design and a model to achieve its purpose. The design specifies the main components of the system while the model gives a formal view for it. A brief description of the design is as follows. The Splitter partitions the hyperspectral image into a set of blocks. The Scatterer generates the necessary scripts and job descriptions for each block and submits them to the job scheduler, which in turn submits them to a computational resource for execution. As the blocks are processed, the results are collected by the Gatherer. Since the blocks are processed independently and in parallel, the Gatherer has to do the necessary bookkeeping to ensure that all the blocks are processed. The Joiner then combines the processed blocks in the right order and outputs the denoised image. This brief description of the design leaves some questions unanswered: how many partitions should the Splitter generate? What's the optimal strategy for partitioning the image? What's the policy for the Gatherer when a block fails or did not finish on time? To formally address these questions, a model for the system is needed and more testing is required. The model used is based on reinforcement learning model [5, 13]. The beauty of this model is that it does not require a complete knowledge of the environment as is the case for distributed computational resources across the internet.

A C++ TISEAN-based implementation was completed recently. Preliminary results showed that with two dedicated processors, we could reduce the execution time by approximately 50%. With N processors, can we reduce the time to 100/N%? If not, what's the largest N for which this holds? This paper will present the results of this development and provide examples of the denoising of AVIRIS, AISA, and Hyperion hyperspectral data. The operational implementation of the algorithm will be through SAFORAH (www.saforah.org) where the user will select the denoised product as an output. The user will not need to deal with the parallel processing, distributed storage, and computational complexities. The denoised product will be delivered to the user using high bandwidth connectivity over the internet.

REFERENCES

[1] T. Han and D. G. Goodenough, “Investigation of nonlinearity in hyperspectral remotely sensed Imagery – a nonlinear time series analysis approach,” *Proceedings of the IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, pp. 1556 - 1560, Barcelona, Spain, July 23 – 27, 2007.
[2] T. Han," Investigation of Nonlinearity in Hyperspectral remotely Sensed Imagery" in Ph.D. Thesis in Computer Science. Victoria: University of Victoria, 2009.
[3] OpenPBS Works^{TM}, "OpenPBS", referred by December 05, 2009, http:// *www.openpbs.org/*.
[4] H. Kantz and T. Schreiber, "Nonlinear Time Series Analysis," Pub. Cambridge University Press, pp. 93, 1997.
[5] R. S. Sutton, "Generalization in reinforcement learning: Successful examples using sparse coding," in D. S. Touretzky, M. C. Mozer and M. E. Hasselmo (eds.) Advances in Neural Information Processing, pp. 1038-1044, MIT Press, Cambridge, CA, 1996.
[6] D. G. Goodenough and Tian Han, "Reducing noise in hyperspectral data — A nonlinear data series analysis approach," First Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing, 2009. WHISPERS 2009, pp. 1-4, 2009.
[7] R. Hegger, H. Kantz, and T. Schreiber, *Practical implementation of nonlinear time series methods: The TISEAN package*, CHAOS **9**, 413 (1999).
[8] G. Mateescu, W. Podaima, A. Charbonneau, R. Impey, M. Viswanathan, A. Agarwal, P. Armstrong, R. Desmarais, I. Gable, S. Popov, S. Ramage, R. Sobie, D.C. Vanderster. "The GridX1 Computational Grid: from a set of service-specific protocols to a service oriented approach", Proceedings of the 21st International Symposium on High Performance Computing and Applications (Regina, Canada), Page 8, 2007.
[9] The Globus Alliance, "Incubator/Gavia-MS", referred by December 01, 2009, http://dev.globus.org/wiki/Incubator/Gavia-MS.
[10] A. Agarwal, M. Ahmed, A. Berman, B.L. Caron, A. Charbonneau, D. Deatrich, R. Desmarais, A. Dimopoulos, I. Gable, L.S. Groer, R. Haria, R. Impey, L. Klektau, C. Lindsay, G.Mateescu, Q.Matthews, A. Norton, W. Podaima, D. Quesnel, R. Simmonds, R.J. Sobie, B. St. Arnaud, C. Usher, D.C. Vanderster, M. Vetterli, R. Walker, M. Yuen. *GridX1: A Canadian computational grid*. Future Generation Computer Systems, Volume 23, Issue 5, June 2007, Pages 680-687.
[11] Condor, “Condor – high throughput computing”, referred by December 02, 2009, http://www.cs.wisc.edu/condor.
[12] GridWay, " GridWay Metascheduler - Metascheduling Technologies for the Grid", referred by February 10, 2009, http://www.gridway.org/doku.php .
[13] R.S. Sutton, and A.G. Barto, Reinforcement Learning: an Introduction, MIT Press, 1998.
[14] A. A. Green, M. Berman, P. Switzer, M. D. Craig, “A transformation for ordering multispectral data in terms of image quality with implications for noise removal,” IEEE Transactions on Geoscience and Remote Sensing, vol. 26, pp. 65-74, 1988.
[15] K. S. Schmidt and A. K. Skidmore, “Smoothing vegetation spectra with wavelets,” International Journal of Remote Sensing, vol. 25, no. 6, pp. 1167 – 1184, 2004.
[16] H. Othman and S. Qian, “Noise reduction of hyperspectral imagery using hybrid spatial-spectral derivative-domain wavelet shrinkage,” IEEE Transactions on Geoscience and Remote Sensing, vol. 44, no. 2, pp. 397 – 408, 2006.
[17] M. Davis, “Noise reduction schemes for chaotic time series,” Physica D, vol. 79, no. 2-4, pp. 174- 192, 1994.
[18] R. Cawley and G-H. Hsu, “Local-geometric method for noise reduction in chaotic maps and flows,” Physical Review A, vol. 46, no. 6, pp. 3057 – 3082, 1992.