Multicore Real-Time Implementation of a Full Hyperspectral Unmixing Chain

Sergio Bernabé, Luis Ignacio Jiménez, Carlos García, Javier Plaza, *Senior Member, IEEE*, and Antonio Plaza, *Fellow, IEEE*

***Abstract*— Solving the mixture problem in remotely sensed hyperspectral images remains a challenging task. In particular, solutions are needed in order to obtain a response for applications with real-time constraints. In the last decade, several efforts have been developed, many of them using graphics processing units (GPUs) and focused on the exploitation of spectral information alone. However, a few spectral unmixing chains have been developed using other architectures such as multicore processors, field programmable gate arrays, or Intel Xeon Phi coprocessors. In this letter, we develop a new parallel unmixing chain for multicore processors. Compared with other approaches, the proposed spatial–spectral alternative takes advantage of the complementary information provided by the spatial correlation of the pixels in the image in addition to the spectral information. Our implementation has been optimized using the application program interface OpenMP and the Intel Math Kernel Library on two multicore architectures, and using real analysis scenarios. The results reveal competitive real-time performance compared with another compute unified device architecture implementation previously developed for GPUs.**

***Index Terms*— Graphics processing units (GPUs), hyperspectral unmixing, multicore processors.**

I. Introduction

HYPERSPECTRAL imaging sensors collect a large number of images in different wavelength channels for the same area of the surface of the Earth [1], [2]. As a consequence, each pixel of the image is characterized by a vector with hundreds of components representing the materials in these areas at different wavelength values as a spectral signature. These signatures provide distinguishing features and allow us to describe the imaged materials in great level of detail [3], [4]. Accordingly, a hyperspectral image can be represented as a 3-D data cube, where the first two dimensions represent the spatial coordinates in a 2-D space and the last one captures the spectral singularity of each pixel in the scene [5], [6].

Spectral unmixing is among the most popular techniques to process hyperspectral images. In the literature, two models have been mainly used to characterize mixed pixels in hyperspectral images. The nonlinear mixture model [7] assumes that the interactions between the endmembers follow a nonlinear model with multiple scattering effects [8] and other nonlinearities. The linear mixture model [9] considers that the mixed pixels present in the scene can be modeled as a linear combination of the pure spectral signatures (endmembers) weighted by their corresponding abundances. The linear mixture model can be expressed in mathematical form as follows:

$$\mathbf{Y} = \mathbf{E}\mathbf{A} + \mathbf{N} \tag{1}$$

where $\mathbf{Y} = [\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_n]$, $\mathbf{E} = [\mathbf{e}_1, \mathbf{e}_2, \ldots, \mathbf{e}_p]$ is a matrix of l-dimensional endmember signatures containing p endmembers, $\mathbf{A} = [\alpha_1, \alpha_2, \ldots, \alpha_n]$ contains the abundance fractions α_i associated with each endmember in each pixel of the scene, and $\mathbf{N}$ accounts for the noise.

Many linear spectral unmixing techniques exhibit a high computational cost but, at the same time, need to satisfy the demands of real-time applications. To address this problem, several parallel systems have been exploited to achieve real-time performance; from small clusters of computers and general-purpose multicore processors to specific accelerators such as field programmable gate arrays (FPGAs), graphics processing units (GPUs) or Intel Xeon Phi coprocessors.

The goal of this letter is to explore the possibility of achieving real-time spectral unmixing on the state-of-the-art multicore architectures, which are quite general and flexible. Our aim is to optimize these architectures by developing a parallel version of a variety of techniques that cover all the stages of the hyperspectral unmixing chain. A comparative evaluation of the performance of our newly developed multicore implementations against single core and GPU versions developed in previous works is also carried out. The analysis is conducted using real hyperspectral scenes widely used as benchmarks.

The remainder of this letter is structured as follows. Section II briefly describes our hyperspectral unmixing chain. Section III is focused on the parallel implementation of the techniques proposed for each stage of the unmixing chain. In Section IV, we describe the experimental process conducted and analyze the results obtained. Finally, Section V concludes with some remarks and hints at plausible future research.

Manuscript received November 15, 2017; revised January 29, 2018; accepted February 16, 2018. Date of publication March 19, 2018; date of current version April 20, 2018. This work was supported by EU (FEDER) and the Spanish MINECO under Grant TIN 2015-65277-R. *(Corresponding author: Sergio Bernabé.)*

S. Bernabé and C. García are with the Department of Computer Architecture and Automation, Complutense University of Madrid, 28040 Madrid, Spain (e-mail: sebernab@ucm.es; garsanca@ucm.es).

L. I. Jiménez is with the Research, Technological Innovation and Supercomputing Center of Extremadura, 10071 Cáceres, Spain (e-mail: luisignacio.jimenez@cenits.es).

J. Plaza and A. Plaza are with the Hyperspectral Computing Laboratory, University of Extremadura, E-10071 Cáceres, Spain (e-mail: jplaza@unex.es; aplaza@unex.es).

Color versions of one or more of the figures in this letter are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/LGRS.2018.2810600

TABLE II

MEAN EXECUTION TIMES AND SPEEDUPS (IN THE PARENTHESES) FOR THE PARALLEL VERSIONS OF THE PROPOSED CHAIN EXECUTED ON TWO DIFFERENT PLATFORMS (MULTICORE AND GPU) AFTER 10 MONTE CARLO RUNS

AVIRIS CUPRITE						
	VD	**k-means**	**SSPP**	**N-FINDR**	**LSU**	**Total**
Parallel GPU1	0.395	0.398	0.169	0.003	0.061	1.026
Parallel GPU2	0.659	0.377	0.244	0.034	0.060	1.374
Parallel MC1 [4 threads]	0.161 (2.45)	0.336 (1.18)	0.288 (0.59)	0.002 (1.50)	0.022 (2.77)	0.809 (1.27)
Parallel MC2 [26 threads]	0.146 (4.51)	0.204 (1.85)	0.248 (0.98)	0.004 (8.50)	0.007 (8.57)	0.609 (2.25)
AVIRIS WTC						
	VD	**k-means**	**SSPP**	**N-FINDR**	**LSU**	**Total**
Parallel GPU1	0.651	1.598	0.465	0.009	0.189	2.912
Parallel GPU2	1.017	1.691	0.680	0.085	0.171	3.644
Parallel MC1 [4 threads]	0.488 (1.33)	1.786 (0.89)	0.995 (0.56)	0.007 (1.29)	0.080 (2.36)	3.356 (0.87)
Parallel MC2 [26 threads]	0.426 (2.39)	0.949 (1.78)	0.723 (0.94)	0.014 (6.07)	0.020 (8.55)	2.132 (1.71)

To conclude this section, we have evaluated the power consumption considering both MC2 and GPU2 architectures using the software solution PowerMeter daemon (pmlib) [21], which gathers power results periodically from the tools provided by manufacturers (namely, running average power limit for the Intel Xeon CPU and NVIDIA Management Library for the NVIDIA GPU).

From Figs. 3 and 4, a few observations can be made about the power dissipation. First, if we analyze the plots, the best platform is the MC2 architecture, which dissipates 193 W on average (411 J) and 287 W at most. Compare this, for example, with the 191 W (696 J) on average for the GPU2 architecture. Second, the best tradeoff solution between performance measured in Mpixel/s and energy consumption is achieved by the MC2: 0.00034 Mpixel/s/J versus 0.00012 for the GPU2 architecture.

V. CONCLUSION

In this letter, we have presented a new implementation of a full unmixing chain for multicore platforms. The obtained results indicate that it is possible to achieve significant speedups and even a higher performance than those achieved for other parallel implementations specifically developed for CPU/GPU architectures. The parallel approaches achieve real-time performance in all cases studied. Future work will focus on the development of hybrid multicore/GPU implementations exploiting the advantages of both architectures, and studying other hyperspectral analysis algorithms that could be significantly improved in terms of computational performance.

REFERENCES

[1] A. F. H. Goetz, G. Vane, J. E. Solomon, and B. N. Rock, "Imaging spectrometry for earth remote sensing," *Science*, vol. 228, no. 4704, pp. 1147–1153, 1985.

[2] A. F. H. Goetz and B. Kindel, "Comparison of unmixing results derived from AVIRIS, high and low resolution, and Hydice images at Cuprite, NV," in *Proc. IX NASA/JPL Airborne Earth Sci. Workshop*, Pasadena, CA, USA, 1999, pp. 1–10.

[3] A. Plaza *et al.*, "Recent advances in techniques for hyperspectral image processing," *Remote Sens. Environ.*, vol. 113, no. 1, pp. 110–122, Sep. 2009.

[4] M. E. Schaepman, S. L. Ustin, A. J. Plaza, T. H. Painter, J. Verrelst, and S. Liang, "Earth system science related imaging spectroscopy—An assessment," *Remote Sens. Environ.*, vol. 113, pp. 123–137, Sep. 2009.

[5] C.-I. Chang, *Hyperspectral Imaging: Techniques for Spectral Detection and Classification*. New York, NY, USA: Kluwer, 2003.

[6] N. Keshava and J. F. Mustard, "Spectral unmixing," *IEEE Signal Process. Mag.*, vol. 19, no. 1, pp. 44–57, Jan. 2002.

[7] R. Heylen, M. Parente, and P. Gader, "A review of nonlinear hyperspectral unmixing methods," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 7, no. 6, pp. 1844–1868, Jun. 2014.

[8] C. C. Borel and S. A. W. Gerstl, "Nonlinear spectral mixing models for vegetative and soil surfaces," *Remote Sens. Environ.*, vol. 47, no. 3, pp. 403–416, 1994.

[9] J. M. Bioucas-Dias *et al.*, "Hyperspectral unmixing overview: Geometrical, statistical, and sparse regression-based approaches," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 5, no. 2, pp. 354–379, Apr. 2012.

[10] A. Zare and P. Gader, "Sparsity promoting iterated constrained endmember detection in hyperspectral imagery," *IEEE Geosci. Remote Sens. Lett.*, vol. 4, no. 3, pp. 446–450, Jul. 2007.

[11] A. Zare and P. Gader, "Hyperspectral band selection and endmember detection using sparsity promoting priors," *IEEE Geosci. Remote Sens. Lett.*, vol. 5, no. 2, pp. 256–260, Apr. 2008.

[12] C.-I. Chang and Q. Du, "Estimation of number of spectrally distinct signal sources in hyperspectral imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 42, no. 3, pp. 608–619, Mar. 2004.

[13] G. Martin and A. Plaza, "Spatial-spectral preprocessing prior to endmember identification and unmixing of remotely sensed hyperspectral data," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 5, no. 2, pp. 380–395, Apr. 2012.

[14] T. Kanungo, D. M. Mount, N. S. Netanyahu, C. D. Piatko, R. Silverman, and A. Y. Wu, "An efficient k-means clustering algorithm: Analysis and implementation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 7, pp. 881–892, Jul. 2002.

[15] M. E. Winter, "N-FINDR: An algorithm for fast autonomous spectral end-member determination in hyperspectral data," in *Proc. SPIE*, Oct. 1999, pp. 266–270.

[16] J. A. Richards and X. Jia, *Remote Sensing Digital Image Analysis: An Introduction*. Berlin, Germany: Springer, 2006.

[17] L. I. Jiménez *et al.*, "GPU implementation of spatial–spectral preprocessing for hyperspectral unmixing," *IEEE Geosci. Remote Sens. Lett.*, vol. 13, no. 11, pp. 1671–1675, Nov. 2016.

[18] C. Gonzalez, D. Mozos, S. Lopez, and R. Sarmiento, "A novel FPGA-based architecture for the estimation of the virtual dimensionality in remotely sensed hyperspectral images," *J. Real-Time Image Process.*, pp. 1–12, 2015.

[19] C. Gonzalez, D. Mozos, J. Resano, and A. Plaza, "FPGA implementation of the N-FINDR algorithm for remotely sensed hyperspectral image analysis," *IEEE Trans. Geosci. Remote Sens.*, vol. 50, no. 2, pp. 374–388, Feb. 2012.

[20] D. Lavenier, "FPGA implementation of the k-means clustering algorithm for hyperspectral images," Los Alamos Nat. Lab., Los Alamos, NM, USA, Tech. Rep., 2000.

[21] S. Barrachina *et al.*, "An integrated framework for power-performance analysis of parallel scientific workloads," in *Proc. 3rd Int. Conf. Smart Grids, Green Commun. IT Energy-Aware Technol.*, 2013, pp. 114–119.