Invited Paper

Parallel Method for Sparse Semisupervised Hyperspectral Unmixing

José M. P. Nascimento[a,b], José M. Rodríguez Alves[a], Antonio Plaza[c],
Vítor Silva[a,d], José M. Bioucas-Dias[a,e]

[a]Instituto de Telecomunicações, Lisbon, Portugal;
[b]Instituto Superior de Engenharia de Lisboa, Lisbon, Portugal;
[c]Hyperspectral Computing Laboratory, University of Extremadura, Cáceres, Spain;
[d] DEEC, University of Coimbra, Coimbra, Portugal;
[e]Instituto Superior Técnico, Technical University of Lisbon, Lisbon, Portugal

ABSTRACT

Parallel hyperspectral unmixing problem is considered in this paper. A semisupervised approach is developed under the linear mixture model, where the abundance's physical constraints are taken into account. The proposed approach relies on the increasing availability of spectral libraries of materials measured on the ground instead of resorting to endmember extraction methods.

Since Libraries are potentially very large and hyperspectral datasets are of high dimensionality a parallel implementation in a pixel-by-pixel fashion is derived to properly exploits the graphics processing units (GPU) architecture at low level, thus taking full advantage of the computational power of GPUs. Experimental results obtained for real hyperspectral datasets reveal significant speedup factors, up to 164 times, with regards to optimized serial implementation.

Keywords: Hyperspectral Imaging, Sparse Unmixing, Sparse Regression, Graphics Processing Unit, Parallel Methods, Spectral Libraries.

1. INTRODUCTION

Hyperspectral imagery is a continuously growing area in remote sensing applications. The spectral range extending from the visible region through the near-infrared and mid-infrared in hundreds of narrow contiguous bands, provides a very high spectral resolution, which allows the detection and the discrimination between different chemical elements of the observed image.[1–3] The main problem of hyperspectral images is that the spatial resolution can vary from a few to tens of meters, thus each pixel is a mixture of several spectrally distinct materials (also called *endmembers*).[1,4]

Hyperspectral unmixing is a source separation problem which amounts at estimating the number of endmembers, their spectral signatures and their abundance fractions (i.e., the percentage of each endmember).[5] Over the last decade, several algorithms have been developed to unmix hyperspectral datasets, from a geometrical and statistical point of view.[1] Among geometrical methods, there are some efficient algorithms from the computational point of view that assume the presence of pure pixels in the dataset, meaning that there is at least one spectral vector on each vertex of the data simplex.[1,6] Some popular algorithms taking this assumption are *vertex component analysis* (VCA),[7] the *automated morphological endmember extraction* (AMEE),[8] the *pixel purity index* (PPI),[9] the N-FINDR,[10–14] the *alternating volume maximization* (AVMAX), the *successive volume maximization* (SVMAX),[11] the *automatic target generation process* (ATGP),[15] and the *robust and recursive non-negative matrix factorization* (RRNMF).[16]

If the pure pixel assumption is not fulfilled, which is a more realistic scenario, the unmixing process is a rather challenging task, since some endmembers are not in the dataset. Some recent methods, in the vein of Craig's

Further author information: (Send correspondence to J. Nascimento) E-mail: zen@isel.pt

High-Performance Computing in Remote Sensing III, edited by Bormin Huang, Antonio J. Plaza, Zhensen Wu,
Proc. of SPIE Vol. 8895, 88950B · © 2013 SPIE · CCC code: 0277-786X/13/$18 · doi: 10.1117/12.2029206

work *minimum Volume Transform* (MVT)[17] which finds the smallest simplex that contain the dataset, are the *simplex identification via split augmented Lagrangian* (SISAL),[18] the *iterated constrained endmember* (ICE),[19] the *sparsity promoting iterated constrained endmember* (SPICE),[20] the *minimum-volume enclosing simplex algorithm* (MVES),[21] the *robust minimum volume enclosing simplex* (RMVES),[22] the *minimum-volume enclosing simplex algorithm* (MVES),[21] and the *alternating projected subgradients* (APS).[23]

A necessary condition for the correct identification of the mixing matrix is the existence of at least $p-1$ (p stands for the number of endmembers) spectral vectors on each facet of the data simplex. However, in highly mixed datasets, this assumption is not realistic, what has fostered research in the statistical front.[24–27]

A semi-supervised alternative approach to unmix hyperspectral datasets consists of assuming that the observed image signatures can be expressed in the form of linear combinations of a number of pure spectral signatures known in advance[28,29] (e.g., spectra collected on the ground by a field spectro-radiometer). Unmixing then amounts to finding the optimal subset of signatures in a spectral library that can best model each mixed pixel in the scene. In practice, this is a combinatorial problem which calls for efficient linear sparse regression techniques, since the number of endmembers participating in a mixed pixel is usually very small compared with the dimensionality of spectral libraries.[5]

In recent years, high-performance computing systems have become more widespread in remote sensing applications, namely, graphics processing units (GPUs) have evolved from expensive application specific units into highly parallel and programmable systems. Several state-of-the-art hyperspectral imaging algorithms have been shown to be able to benefit from this hardware and take advantage of the extremely high floating-point processing performance, compact size, huge memory bandwidth, and relatively low cost of these units, which make them appealing for onboard data processing.[30–35]

This paper proposes a parallel method designed for GPUs, to solve the constrained sparse regression problem. This method is based on the *spectral unmixing by splitting and augmented Lagrangian* (SUNSAL)[36] that estimates the abundance fractions using the alternating direction method of multipliers (ADMM),[37] which decomposes a difficult problem into a sequence of simpler ones, alleviating considerably the computational burden, and allowing a significant increase in processing speed. Additionally, this method is performed in a pixel-by-pixel fashion, thus, it is highly parallelizable, and can be effectively implemented by simply dividing large datasets onto smaller subsets without the need to establish an optimal size for them.

The remainder of the paper is organized as follows. Section 2 formulates the problem and presents the fundamentals of the sparse regression problem. Section 3 describes the proposed parallel method designed for GPU. Section 4 evaluates the acceleration of the proposed method from the computational point of view. Finally, Section 5 concludes the paper with some remarks.

2. SPARSE UNMIXING FORMULATION

Linear mixing model considers that a l-dimensional mixed pixel $\mathbf{y}$, of an hyperspectral image with l spectral bands, is a linear combination of endmember signatures weighted by the correspondent abundance fractions.

Let $\mathbf{A} \equiv [\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_m]$ denote a spectral library with m spectral signatures, each with l spectral bands, where it is assumed that the number of spectral signatures is much larger than the number of bands ($m \gg l$). Assuming that matrix $\mathbf{Y} \equiv [\mathbf{y}_1, \ldots, \mathbf{y}_n] \in \mathbb{R}^{l \times n}$ holds n observed spectral vectors and is given by

$$\mathbf{Y} \quad = \quad \mathbf{AS} + \mathbf{N}, \tag{1}$$

where $\mathbf{S} \equiv [\mathbf{s}_1, \ldots, \mathbf{s}_n] \in \mathbb{R}^{m \times n}$ is the abundance fraction matrix and $\mathbf{N} \equiv [\mathbf{n}_1, \ldots, \mathbf{n}_n] \in \mathbb{R}^{l \times n}$ is the additive noise. To be physically meaningful, abundance fractions are subject to the often called *abundance nonnegativity constraint* (ANC) and *abundance sum-to-one constraint* (ASC):[4,5]

$$\begin{aligned} \mathbf{S} &\geq 0\,, \\ \mathbf{1}_m^T \mathbf{S} &= \mathbf{1}_n^T, \end{aligned} \tag{2}$$

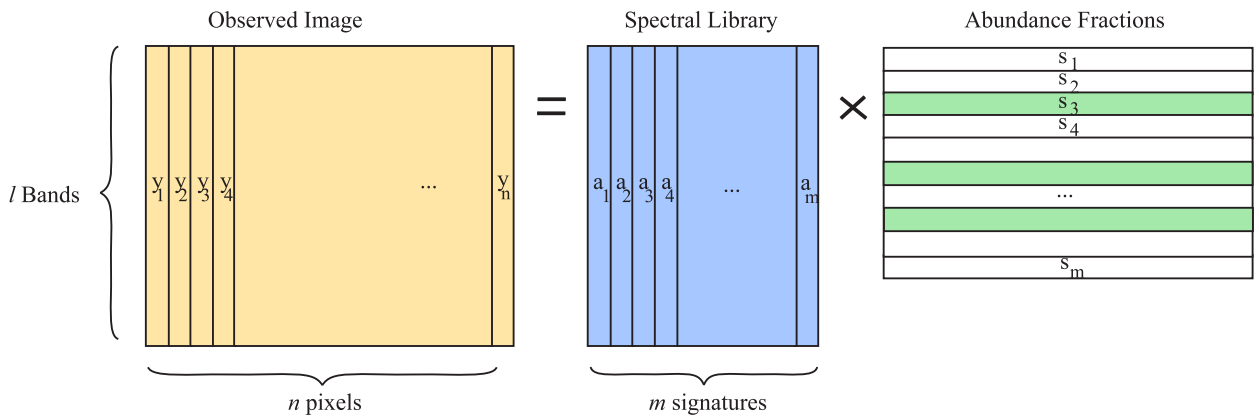

Figure 1. Graphical illustration of the sparsity problem. Material signatures of the considered spectral library $\mathbf{A}$ that contribute to the observed image are represented in green color, and non-active signatures of the considered spectral library $\mathbf{A}$ are represented in white color.

where the inequality $\mathbf{S} \geq 0$ is to be understood componentwise, $\mathbf{1}_m$, and $\mathbf{1}_n$ denote a $1 \times m$ and $1 \times n$ column vectors filled of 1's, respectively. Thus, the abundance fractions estimation problem can formulated as follows:

$$\begin{aligned} \min_{\mathbf{S}} \quad & \|\mathbf{Y} - \mathbf{AS}\|_F^2 + \lambda \sum_{k=1}^{m} \|\mathbf{s}^k\|_2 \\ \text{subject to :} \quad & \mathbf{S} \geq 0\,, \end{aligned} \tag{3}$$

where notation $\|(\cdot)\|_F$ stands for Frobenius norm, λ is a positive regularization parameter, and $\mathbf{s}^k$ denotes the k-th line of $\mathbf{S}$. The convex term $\sum_{k=1}^{m} \|\mathbf{s}^k\|_2$ promotes the sparsity among lines of $\mathbf{S}$, i.e., it promotes solutions of expression (3) with small number of nonzero lines of $\mathbf{S}$. This is further illustrated in Fig. 1, where the fact that only a few signatures contained in $\mathbf{A}$ are likely to contribute to the observed spectra $\mathbf{Y}$, $\mathbf{S}$ contains many zero values (represented in white color).

2.1 SUNSAL method

The optimization problem (3) can be posed in the convex optimization framework, where the convex term promotes the sparsity of $\mathbf{S}$. This problem is a particular case of the constrained $\ell_2 - \ell_1$ problems solved by SUNSAL, corresponding to the absence of the ℓ_1 term. SUNSAL algorithm uses the alternating direction method of multipliers (ADMM)[37] to solve (3). The pseudo-code is presented in **Algorithm 1** .

Algorithm 1 : SUNSAL

choose $\mu > 0$, $\mathbf{U}_0$, and $\mathbf{D}_0$.
$\mathbf{H} := \mathbf{A}^T\mathbf{Y}$
$\mathbf{B} := \mathbf{A}^T\mathbf{A} + \mu\mathbf{I}$
$\mathbf{c} := \mathbf{B}^{-1}\mathbf{1}_m \left(\mathbf{1}_m^T\mathbf{B}^{-1}\mathbf{1}_m\right)^{-1}$
$\mathbf{G} := \mathbf{B}^{-1} - \mathbf{c}\mathbf{1}_m^T\mathbf{B}^{-1}$
$k := 0$
repeat
  $\mathbf{R} := \mathbf{H} + \mu\left(\mathbf{U}_k + \mathbf{D}_k\right)$
  $\mathbf{S}_{k+1} := \mathbf{G}\mathbf{R} + \mathbf{c}\mathbf{1}_n^T$
  $\mathbf{V}_k := \mathbf{S}_{k+1} - \mathbf{D}_k$
  $\mathbf{U}_{k+1} := \max\{0, \mathbf{V}_k\}$
  $\mathbf{D}_{k+1} := \mathbf{D}_k - (\mathbf{S}_{k+1} - \mathbf{U}_{k+1})$
  $k := k + 1$
until stopping criterion is satisfied

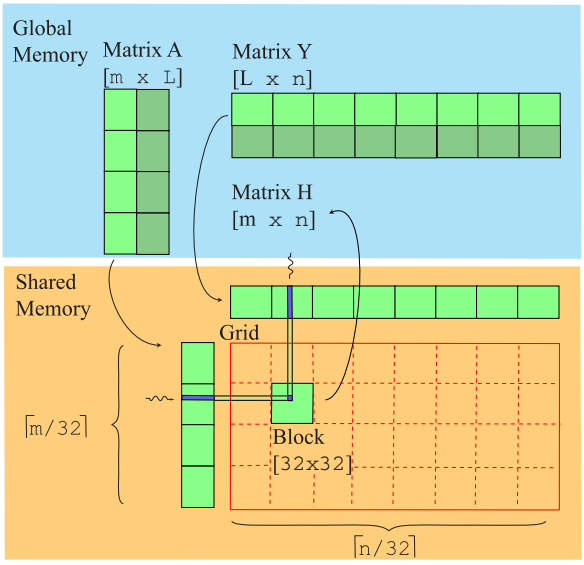

Figure 2. Illustration of parallel SUNSAL in the GPU: multiplication kernel.

3. SPARSE UNMIXING IMPLEMENTATION ON GPU

SUNSAL it is highly parallelizable, since all calculations can be performed in a pixel-by-pixel basis. The first step is to map the hyperspectral image and the spectral library into GPU global memory. Fig. 2 shows the schematic of the proposed implementation for the multiplication kernel used in line 2 and line 9 of Algorithm 1. The grid contains $\lceil m/32 \rceil \times \lceil n/32 \rceil$ blocks of 32×32 and the number of threads for each block is 1024, thus all pixels in the hyperspectral image are processed in parallel. The algorithm is constructed by a chain of kernels. The first kernel computes the matrix $\mathbf{H}$ (line 2 of Algorithm 1). In order to minimize the number of global memory accesses, matrices $\mathbf{A}$ and $\mathbf{Y}$ are partitioned into sub-blocks of 32×32, which is the size of the block, and transferred, step by step, to the shared memory. The result for each element of $\mathbf{H}$ is the sum of several partial products. Inside the loop, another kernel compute matrix $\mathbf{R}$ (see line 8 of Algorithm 1). This kernel launches as many threads as elements that are present in $\mathbf{R}$, where each thread computes an element of the $\mathbf{H} + \mu\,(\mathbf{U}_k + \mathbf{D}_k)$ and stores the result in the global memory. The kernel that computes the abundance estimates, $\mathbf{S}$, on each iteration (line 9 of Algorithm 1), is decomposed on two operations: first it computes the product of matrices $\mathbf{G}$ and $\mathbf{R}$, followed by the addition of matrix $\mathbf{c}\mathbf{1}_N^T$. The scheme herein used is similar to the first kernel used to compute matrix $\mathbf{H}$ whereas the kernels to update $\mathbf{V}$ and $\mathbf{D}$ follow the same rule has the kernel used to compute $\mathbf{R}$. Finally, $\mathbf{U}$ is updated by analyzing if each element of $\mathbf{V}$ is negative.

4. PERFORMANCE EVALUATION

In this section, the sequential and parallel sparse unmixing methods are tested on a subset of the AVIRIS Cuprite Nevada dataset, available online in reflectance units. The Cuprite site has been extensively used for remote sensing experiments over the past years, is well understood mineralogically, and has several exposed minerals of interest, all included in the USGS library considered in experiments.

The subset comprises a portion of 250×190 pixels with 224 spectral bands between 0.4 and 2.5 μm, with a spectral resolution of 10 nm. Prior to the analysis, bands 1-2, 105-115, 150-170, and 223-224 were removed due to water absorption and low SNR in those bands, leaving a total of 188 spectral bands. Fig. 3 shows band 30 (wavelength $\lambda = 647.7nm$) of the subset considered.

In order to compare the parallel and the sequential versions in terms of processing times, the sequential version of the method was implemented in C programming language running on one core of the Intel i7-2600 CPU, with 16 Gbyte memory and the parallel version was implemented in OpenCL programming language running on a GPU card equipped with a GTX-680 from Nvidia.

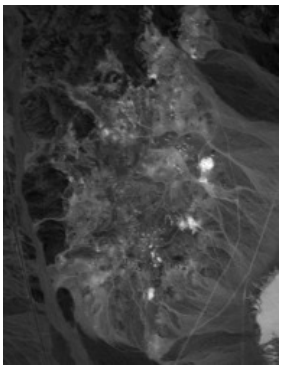

Figure 3. Band 30 (wavelength $\lambda = 647.7nm$) of the subimage of AVIRIS Cuprite Nevada dataset.

Table 1. Processing times and speedup for the Cuprite dataset ($l = 188$) and for the USGS library ($m = 342$).

	$n = 1000$	$n = 10000$	$n = 47750$
Sequential (CPU)	845	19 333	76 323
Parallel (GTX680)	36	124	465
Speedup	23.4	155.9	164.1

Table 1 illustrates the processing time and the speedup for the Cuprite dataset, for each column a different number of pixels is selected. The achieved speedup is approximately 164 with regards to the sequential version, which is quite remarkable taking into account that the sequential version has been carefully optimized. Note that, as expected, the acceleration factors are higher as the number of pixels to be processed is larger. The GPU total time presented in table 1 accounts for both the processing time and the time spent on host/device memory transfers, being the last one responsible for the major part.

Fig. 4, shows the estimated abundances of all pixels of the dataset, considering the library materials. It is worse nothing that most of the materials do not contribute to the mixture (represented in white color), thus, the matrix is sparse. Fig. 5, presents the abundance maps of four different materials which dominate the considered scene. The results herein presented are in accordance with the USGS abundance maps of the region.

5. CONCLUSIONS

In this paper a parallel semisupervised method for sparse hyperspectral unmixing is proposed. The method developed on GPU has a significant speedup when compared with regards to the sequential version. Experimental results, conducted using real hyperspectral datasets collected by the AVIRIS instrument and spectral libraries publicly available from U.S. Geological Survey (USGS), indicate the potential of parallel sparse unmixing methods on high performance computing environments in the task of accurately characterizing mixed pixels using library spectra. This opens new perspectives for spectral unmixing, since the abundance estimation process no longer depends on the availability of pure spectral signatures in the input data nor on the capacity of a certain endmember extraction algorithm to identify such pure signatures.

The method was designed on the multi-platform OpenCL programming language. As future work, we are currently exploring the role of multi-core CPU architectures in hyperspectral imaging and further developing an implementation using OpenMP that will lead to a comparison of GPUs versus multicore CPUs in this context.

ACKNOWLEDGMENTS

The authors would like to take this opportunity to gratefully thank Fundação para a Ciência e a Tecnologia and Instituto de Telecomunicações support under project PEst-OE/EEI/LA0008/2013.

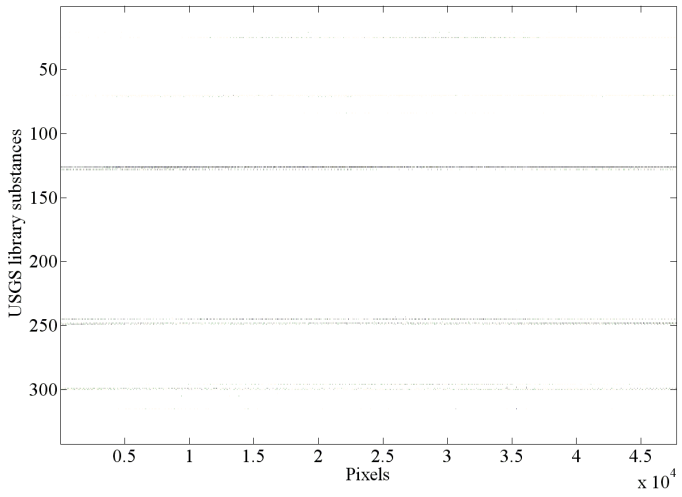

Figure 4. Estimated abundances of the USGS library materials for all pixels of the Cuprite dataset (gray scale where white and black colors represent 0 and 1, respectively).

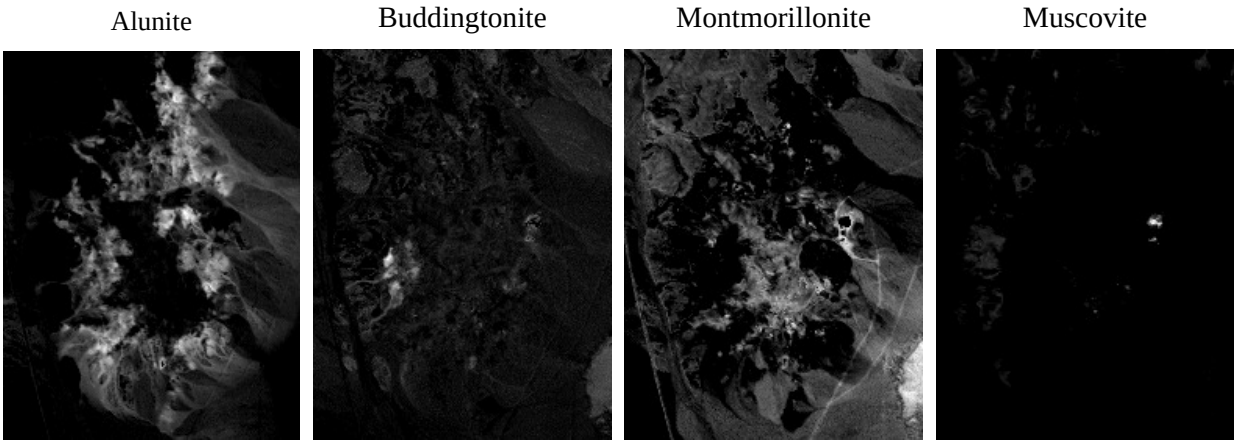

Figure 5. Estimated abundance maps of the dominant materials in the scene (gray scale where black and white colors represent 0 and 1, respectively).

REFERENCES

[1] Bioucas-Dias, J., Plaza, A., Dobigeon, N., Parente, M., Du, Q., Gader, P., and Chanussot, J., "Hyperspectral unmixing overview: Geometrical, statistical, and sparse regression-based approaches," *IEEE J. Sel. Topics Appl. Earth Observations Remote Sens.* **99**(1-16) (2012).

[2] Parente, M. and Plaza, A., "Survey of geometric and statistical unmixing algorithms for hyperspectral images," *Proceedings of the 2nd IEEE Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing (WHISPERS)* , 1–4 (Reykjavik, Iceland, Jun. 14-16, 2010).

[3] Bioucas-Dias, J. M. and Plaza, A., "Hyperspectral unmixing: geometrical, statistical, and sparse regression-based approaches," *Proceedings of SPIE, Image and Signal Processing for Remote Sensing XVI* **7830**, 1–15 (Toulouse, France, Sept. 20-23, 2010).

[4] Nascimento, J. M. P. and Bioucas-Dias, J. M., "Does Independent Component Analysis Play a Role in Unmixing Hyperspectral Data?," *IEEE Trans. Geosci. Remote Sensing* **43**(1), 175–187 (2005).

[5] Keshava, N. and Mustard, J., "Spectral Unmixing," *IEEE Signal Processing Mag.* **19**(1), 44–57 (2002).
[6] Keshava, N., Kerekes, J., Manolakis, D., and Shaw, G., "An Algorithm Taxonomy for Hyperspectral Unmixing," in [*Proc. of the SPIE AeroSense Conference on Algorithms for Multispectral and Hyperspectral Imagery VI*], **4049**, 42–63 (2000).
[7] Nascimento, J. M. P. and Bioucas-Dias, J. M., "Vertex Component Analysis: A Fast Algorithm to Unmix Hyperspectral Data," *IEEE Trans. Geosci. Remote Sensing* **43**(4), 898–910 (2005).
[8] Plaza, A., Martinez, P., Perez, R., and Plaza, J., "Spatial/Spectral Endmember Extraction by Multidimensional Morphological Operations," *IEEE Trans. Geosci. Remote Sensing* **40**(9), 2025–2041 (2002).
[9] Boardman, J., "Automating Spectral Unmixing of AVIRIS Data using Convex Geometry Concepts," in [*Summaries of the Fourth Annual JPL Airborne Geoscience Workshop, JPL Pub. 93-26, AVIRIS Workshop.*], **1**, 11–14 (1993).
[10] Winter, M. E., "N-FINDR: An Algorithm for Fast Autonomous Spectral End-member Determination in Hyperspectral Data," in [*Proc. of the SPIE conference on Imaging Spectrometry V*], **3753**, 266–275 (1999).
[11] Chan, T.-H., Ma, W.-K., Ambikapathi, A., and Chi, C.-Y., "A simplex volume maximization framework for hyperspectral endmember extraction," *IEEE Trans. Geosci. Remote Sensing* **49**(11), 4177 –4193 (2011).
[12] Liu, J. and Zhang, J., "A new maximum simplex volume method based on householder transformation for endmember extraction," *IEEE Trans. Geosci. Remote Sensing* **50**, 104 –118 (jan. 2012).
[13] Tao, X., Wang, B., and Zhang, L., "Orthogonal bases approach for the decomposition of mixed pixels in hyperspectral imagery," *IEEE Geosci. Remote Sensing Let.* **6**(2), 219 – 223 (2009).
[14] Chang, C.-I., Wu, C.-C., min Liu, W., and chieh Ouyang, Y., "A New Growing Method for Simplex-Based Endmember Extraction Algorithm," *IEEE Trans. Geosci. Remote Sensing* **44**(10), 2804–2819 (2006).
[15] Ren, H. and Chang, C.-I., "Automatic spectral target recognition in hyperspectral imagery," *IEEE Transactions on Aerospace and Electronic Systems* **39**(4), 1232–1249 (2003).
[16] Gillis, N. and Vavasis, S., "Fast and robust recursive algorithms for separable nonnegative matrix factorization," (2012). arXiv preprint arXiv:1208.1237.
[17] Craig, M. D., "Minimum-volume Transforms for Remotely Sensed Data," *IEEE Trans. Geosci. Remote Sensing* **32**, 99–109 (1994).
[18] Bioucas-Dias, J. M., "A Variable Splitting Augmented Lagrangian Approach to Linear Spectral Unmixing," in [*First IEEE GRSS Workshop on Hyperspectral Image and Signal Processing-WHISPERS'2009*], (2009).
[19] Berman, M., Kiiveri, H., Lagerstrom, R., Ernst, A., Dunne, R., and Huntington, J. F., "ICE: A Statistical Approach to Identifying Endmembers in Hyperspectral Images," *IEEE Trans. Geosci. Remote Sensing* **42**(10), 2085– 2095 (2004).
[20] Zare, A. and Gader, P., "Sparsity Promoting Iterated Constrained Endmember Detection in Hyperspectral Imagery," *IEEE Geosci. Remote Sensing Let.* **4**(3), 446 – 450 (2007).
[21] Chan, T.-H., Chi, C.-Y., Huang, Y.-M., and Ma, W.-K., "A Convex Analysis-Based Minimum-Volume Enclosing Simplex Algorithm for Hyperspectral Unmixing," *IEEE Trans. Signal Processing* **57**(11), 4418 – 4432 (2009).
[22] Ambikapathi, A., Chan, T.-H., Ma, W.-K., and Chi, C.-Y., "Chance-constrained robust minimum-volume enclosing simplex algorithm for hyperspectral unmixing," *IEEE Trans. Geosci. Remote Sensing* **49**(11), 4194 –4209 (2011).
[23] Zymnis, A., Kim, S.-J., Skaf, J., Parente, M., and Boyd, S., "Hyperspectral Image Unmixing via Alternating Projected Subgradients," in [*41st Asilomar Conferece on Signals, Systems, and Computer*], 4–7 (2007).
[24] Nascimento, J. M. P. and Bioucas-Dias, J. M., "Hyperspectral unmixing based on mixtures of dirichlet components," *IEEE Trans. Geosci. Remote Sensing* **50**(3), 863–878 (2012).
[25] Dobigeon, N., Tourneret, J.-Y., and Chang, C.-I., "Semi-Supervised Linear Spectral Unmixing Using a Hierarchical Bayesian Model for Hyperspectral Imagery," *IEEE Trans. Signal Processing* **56**(7), 2684–2695 (2008).
[26] Dobigeon, N., Moussaoui, S., Coulon, M., Tourneret, J.-Y., and Hero, A. O., "Joint Bayesian Endmember Extraction and Linear Unmixing for Hyperspectral Imagery," *IEEE Trans. Signal Processing* **57**(11), 4355 – 4368 (2009).

[27] Arngren, M., Schmidt, M. N., and larsen, J., "Bayesian Nonnegative Matrix Factorization with Volume Prior for Unmixing of Hyperspectral Images," in [*Machine Learning for Signal Processing, IEEE Workshop on (MLSP)*], (Sep 2009).

[28] Iordache, M. D., Bioucas-Dias, J., and Plaza, A., "Sparse unmixing of hyperspectral data," *IEEE Transactions on Geoscience and Remote Sensing* **49**(6), 2014–2039 (2011).

[29] Rogge, D. M., Rivard, B., Zhang, J., and Feng, J., "Iterative spectral unmixing for optimizing per-pixel endmember sets," *IEEE Trans. Geosci. Remote Sens.* **44**(12), 3725–3736 (2006).

[30] Christophe, E., Michel, J., and Inglada, J., "Remote sensing processing: From multicore to GPU," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* **4**(3), 643–652 (2011).

[31] Goodman, J. A., Kaeli, D., and Schaa, D., "Accelerating an imaging spectroscopy algorithm for submerged marine environments using graphics processing units," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* **4**(3), 669–676 (2011).

[32] Mielikainen, J., Huang, B., and Huang, A., "GPU-accelerated multi-profile radiative transfer model for the infrared atmospheric sounding interferometer," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* **4**(3), 691–700 (2011).

[33] Lee, C. A., Gasster, S. D., Plaza, A., Chang, C.-I., and Huang, B., "Recent developments in high performance computing for remote sensing: A review," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* **4**(3), 508–527 (2011).

[34] Plaza, A., Du, Q., Y.-L-Chang, and King, R. L., "High performance computing for hyperspectral remote sensing," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* **4**(3), 528–544 (2011).

[35] Sanchez, S., Paz, A., Martin, G., and Plaza, A., "Parallel unmixing of remotely sensed hyperspectral images on commodity graphics processing units," *Concurrency and Computation: Practice & Experience* **23**(13), 1538–1557 (2011).

[36] Bioucas-Dias, J. and Figueiredo, M., "Alternating direction algorithms for constrained sparse regression: Application to hyperspectral unmixing," in [*Hyperspectral Image and Signal Processing: Evolution in Remote Sensing (WHISPERS), 2010 2nd Workshop on*], 1 –4 (2010).

[37] Boyd, S., Parikh, N., Chu, E., Peleato, B., and Eckstein, J., "Distributed optimization and statistical learning via the alternating direction method of multipliers.," *Foundations and Trends in Machine Learning* **3**(1), 1–122 (2011).