A New Digital Repository for Hyperspectral Imagery With Unmixing-Based Retrieval Functionality Implemented on GPUs

Jorge Sevilla and Antonio Plaza, *Senior Member, IEEE*

***Abstract*—Over the last few years, hyperspectral image data have been collected for a large number of locations over the world, using a variety of instruments for Earth observation. In addition, several new hyperspectral missions will become operational in the near future. Despite the increasing availability and large volume of hyperspectral data in many applications, there is no common repository of hyperspectral data intended to distribute and share free hyperspectral data sets in the community. Quite opposite, the hyperspectral data sets which are available for public use are spread among different storage locations and exhibit significant heterogeneity regarding their format, associated meta-data (if any), or ground-truth information. The development of a standardized hyperspectral data repository is a highly desired goal in the remote sensing community. In this paper, we take a necessary first step toward the development of a completely open digital repository for remotely sensed hyperspectral data. The proposed system (available online for public use at: http://www.hypercomp.es/repository) allows uploading new hyperspectral data sets along with meta-data, ground-truth, analysis results, and pointers to bibliographic references describing the use of the data. The current implementation consists of a front-end which allows management of hyperspectral images through a web interface. The system is implemented on a parallel cluster system in order to guarantee storage availability and fast performance. The system includes a spectral unmixing-guided content-based image retrieval (CBIR) functionality which allows searching for images from the database using queries or available information such as spectral libraries. Specifically, for each new hyperspectral scene which is cataloged in our system, we extract the spectrally pure components (*endmembers*) and their associated fractional abundances, and then store this information as meta-data associated to the hyperspectral image. The meta-data can be used to efficiently retrieve images based on their information content. In order to accelerate the process of obtaining the meta-data for a new entry in the system, we develop efficient implementations of spectral unmixing algorithms on graphics processing units (GPUs). This paper particularly focuses on the software design of the system and provides an experimental validation of the unmixing-based retrieval functionality using both synthetic and real hyperspectral images.**

Manuscript received November 03, 2013; revised January 19, 2014; accepted March 25, 2014. Date of publication April 30, 2014; date of current version August 01, 2014. This work was supported in part by the CEOS-SPAIN Project AYA2011-29334-C02-02, in part by the Spanish Ministry of Science and Innovation, in part by the computing facilities of Extremadura Research for Advanced Technologies (CETA-CIEMAT), and in part by the European Regional Development Fund (ERDF). The CETA-CIEMAT belongs to the Spanish Ministry of Science and Innovation. West University of Timisoara supported the system implementation and provided computing resources that were used in the preliminary implementations of our system.

The authors are with the Hyperspectral Computing Laboratory, Department of Technology of Computers and Communications, University of Extremadura, E-10003 Cáceres, Spain (e-mail: jorgesece@unex.es; aplaza@unex.es).

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

Digital Object Identifier 10.1109/JSTARS.2014.2314601

***Index Terms*—Content-based image retrieval (CBIR), graphics processing units (GPUs), hyperspectral imaging, parallel computing, spectral unmixing.**

I. Introduction

THE incorporation of content-based image retrieval (CBIR) [1] techniques into remote sensing data repositories offers significant advantages from the viewpoint of effectively managing, storing, and retrieving large volumes of remotely sensed data [2]. Hyperspectral imaging (also known as imaging spectroscopy [3]) is a fast growing area in remote sensing. Hyperspectral scenes consisted of hundreds of images (at different wavelength channels) for the same area on the surface of the Earth, thus generating very large data volumes that need to be efficiently stored and managed. The interpretation of remotely sensed hyperspectral scenes is also an increasingly relevant research topic involving many different analysis techniques [4].

In order to provide an idea of available and future missions for Earth observation using hyperspectral instruments, Table I provides a summary of the main characteristics of eight hyperspectral instruments: two airborne (HYDICE [5] and AVIRIS [6]) and six spaceborne (HYPERION [7], EnMAP [8], PRISMA [9], CHRIS [10], HyspIRI [11], and IASI [12]). From this list, EnMAP, PRISMA, and HyspIRI are not yet operational. The spatial resolutions are generally higher for airborne instruments. The spectral coverage of HYDICE, AVIRIS, HYPERION, EnMAP, PRISMA, and HyspIRI corresponds to the visible, the near-infrared, and the shortwave infrared spectral bands (typically, from 0.4 to 2.5 nm), whereas CHRIS covers only the visible bands and IASI also covers the mid-infrared and the long-infrared bands. The number of spectral bands is approximately 200 for HYDICE, AVIRIS, HYPERION, EnMAP, PRISMA, and HyspIRI, with a spectral resolution of the order of 10 nm. The lowest number of bands is provided by CHRIS, with 63 bands and spectral resolutions of 1.3 and 12 nm (depending on the region of the spectrum). Quite opposite, instruments such as IASI provide up to 8461 spectral bands. In all cases, the spectral resolution is very high (offering a huge potential to discriminate materials).

Several factors make the analysis of hyperspectral data a complex and hard task, calling for sophisticated analysis methods. Among these factors, we emphasize the presence of spectral mixing effects that have been generally approached by identifying a set of spectrally pure signatures in the scene (called *endmembers* in spectral unmixing terminology) and their corresponding

TABLE I
MAIN CHARACTERISTICS OF SEVERAL (AVAILABLE AND NEW) HYPERSPECTRAL IMAGING INSTRUMENTS

Parameter	HYDICE	AVIRIS	HYPERION	EnMAP	PRISMA	CHRIS	HysplRI	IASI
Altitude (Km)	1.6	20	705	653	614	556	626	817
Spatial resolution (m)	0.75	20	30	30	5–30	36	60	V: 1–2 km H: 25 km
Spectral resolution (nm)	7–14	10	10	6.5–10	10	1.3–12	4–12	0.5 cm^{-1}
Coverage (μm)	0.4–2.5	0.4–2.5	0.4–2.5	0.4–2.5	0.4–2.5	0.4–1.0	0.38–2.5 & 7.5–12	3.62–15.5 (645-2760 cm^{-1})
Number of bands	210	224	220	228	238	63	217	8461
Data cube size (samples x lines x bands)	200 x 320 x 210	512 x 614 x 224	660 x 256 x 220	1000 x 1000 x 228	400 x 880 x 238	748 x 748 x 63	620 x 512 x 210	765 x 120 x 8461

abundance fractions in each (mixed) pixel of the scene [13]. An additional issue is the extremely high dimensionality and size of the data, resulting from the very fine spatial, spectral, and temporal resolutions currently provided by hyperspectral instruments (see Table I). This demands fast computing solutions that can accelerate the interpretation and efficient exploitation of hyperspectral data sets in various applications [14]–[16].

Although the amount and volume of hyperspectral image data have been significantly increased in recent years, with a large number of data sets already collected over different locations over the world and several new missions under development, the data sets which are available for public use are spread among different storage locations and present significant heterogeneity regarding the storage format, associated meta-data (if any), or ground-truth availability. As a result, only a few data sets are recurrently used to validate hyperspectral imaging applications, and available data are highly fragmented. Nowadays, it is estimated that a large fraction of collected hyperspectral data sets are never used, but simply stored in different databases. Even if the use of standardized benchmark data is quite interesting from the viewpoint of algorithm comparison, there is a need to increase the pool of benchmark hyperspectral data sets available to the community in order to allow a more appropriate selection of specific test data for different applications. This functionality is already available in other systems which are able to effectively provide remotely sensed data on-demand and with high retrieval performance [17]–[20]. At present, there is no common repository of hyperspectral data sets which can effectively distribute such data among potential users. Since the amount and volume of hyperspectral data are expected to significantly increase with the new missions described in Table I, a highly desirable objective in the hyperspectral imaging community is to develop new tools to effectively share large amounts of hyperspectral data together with their high-level associated information (e.g., ground-truth, analysis results, pointers to bibliographic references describing previous results on the data, etc.)

In this paper, we take a necessary first step toward the development of a completely open and shared digital repository for remotely sensed hyperspectral data with CBIR functionality. The system is implemented on a cluster of PCs [each one equipped with a graphic processing unit (GPU)] and takes advantage of spectral unmixing concepts to generate effective meta-data for image retrieval purposes, thus allowing potential users of the system to effectively retrieve images based on their content. Here, we use the information provided by spectral unmixing (i.e., the spectral endmembers and their associated abundances) as meta-data to assist users in the task of efficiently searching hyperspectral image instances in our repository. Each time a new hyperspectral image is stored in our system, a full spectral unmixing chain is run in order to automatically obtain the meta-data and catalog the scene. Additional information (if available) can also be introduced for each new scene and will be stored in the parallel file system of the cluster. Once the meta-data has been generated and the image has been stored in the database, it can be retrieved by a query provided by the user, such as spectral signatures of interest in a previously available library or the minimum abundance of the spectral materials that should be present in the retrieved scenes. In order to deal with the computational cost of extracting the information needed to catalog a new hyperspectral image in our system, we use an efficient GPU implementation of the full spectral unmixing chain to enhance the storage and retrieval process. The proposed system is experimentally validated in this work using both synthetic and real hyperspectral scenes already included in the system.

The remainder of the paper is organized as follows. Section II describes related work in the development of CBIR systems for hyperspectral image retrieval and also in the design of efficient GPU implementations of spectral unmixing algorithms. Section III describes the software implementation of the proposed system, which is composed of three main layers: 1) *client layer*, which defines the interactions between the user and the system through a web interface; 2) *server layer*, which manages the requests from end-users; and 3) *processing layer*, in charge of more complex processing tasks such as generation of meta-data or image retrieval. Section IV presents an experimental validation of the system. For this purpose, we use a collection of synthetic and real hyperspectral data sets which are already available in the system. Section V concludes the paper with some remarks and hints at plausible future research lines.

II. RELATED WORK

This section describes related work focused on the development of CBIR strategies and GPU implementations in the context of spectral unmixing applications. These topics are the main ingredients of our newly proposed system and therefore, we illustrate previous developments in those areas before introducing our system.

A. CBIR Systems for Hyperspectral Images

There have been several attempts toward the development of CBIR systems in the area of hyperspectral imaging. One of the most relevant ones was described in [21], which presents a spectral/spatial CBIR system for hyperspectral images. The authors use endmember induction algorithms to extract a set of image spectral features, and then compute spatial features as abundance image statistics. These two sources of information are then combined into a dissimilarity measure that guides the search for answers to database queries [22]. In this context, each hyperspectral image is characterized by a tuple given by the set of induced endmembers and the set of fractional abundance maps resulting from an unmixing process conducted using three stages [23]. For the estimation of the number of endmembers, the authors use the virtual dimensionality (VD) [24] method. For the endmember induction step, the authors use three different methods: 1) a fast implementation of the pixel purity index (PPI) algorithm [25], called fast iterative PPI (FIPPI) [26]; 2) the volume-based N-FINDR method [27]; and 3) the incremental lattice strong independence algorithm (ILSIA) [28]. For the estimation of the fractional abundances of inducted endmembers, the authors use a fully constrained least-squares unmixing (FCLSU) algorithm [29]. The authors validate their approach using synthetic data and a real hyperspectral data set (with 2878×512 pixels and 125 spectral bands, cut in patches of 64×64 pixels for a total of 360 patches).

A similar strategy is employed in [30], which presents a parallel heterogeneous CBIR system for efficient hyperspectral image retrieval using spectral mixture analysis. Here, the PPI algorithm performs endmember extraction and FCLSU estimates the abundances. A spectral signature matching algorithm guides the queries to the database. This algorithm first considers the spectral angle distance (SAD) [31] in order to retrieve images by means of an endmember-guided similarity criterion, and then the search is refined by analyzing the relative difference between the abundance fractions associated to the retrieved image and the example image for the search. Another contribution of [30] is an efficient implementation of the system for heterogeneous networks of computers, possibly distributed among different locations. This idea arised from the naturally distributed format of hyperspectral image databases. The system was tested using a collection of 154 hyperspectral data sets collected by the AVIRIS sensor over the World Trade Center area in New York, only a few days after the terrorist attacks of September 11, 2001. The implementation used a heterogeneous network of 16 workstations, and also the NASA Thunderhead cluster[1] with 256 CPUs, providing good results in terms of image retrieval accuracy and parallel performance.

B. GPU Implementations of Spectral Unmixing Techniques

In recent years, GPUs have evolved into highly parallel, multithreaded, many-core coprocessors with tremendous computational power and memory bandwidth [32]. The combined features of general-purpose supercomputing, high parallelism, high memory bandwidth, low cost, compact size, and excellent programmability are now making GPU-based desktop computers an appealing alternative to massively parallel systems made up of commodity CPUs. The exploding GPU capability has attracted more and more scientists and engineers to use it as a cost-effective high-performance computing platform in many applications, including hyperspectral imaging problems. In addition, GPUs can also significantly increase the computational power of cluster-based and distributed systems (e.g., clusters of GPUs are becoming an important architecture for supercomputing purposes.[2]).

Several efforts exploiting GPU technology can already be found in the hyperspectral unmixing literature, including [33] and references therein. Only in the area of spectral unmixing of hyperspectral data, there have been many developments already. A GPU-based implementation of an automated morphological endmember extraction (AMEE) algorithm for pure spectral signature identification was described in [34]. In this case, speedups on the order of $15\times$ were reported. The well-known PPI algorithm [35] has been implemented in GPUs using different strategies [36], [37]. A GPU-based real-time implementation of the vertex component analysis (VCA) algorithm [38] has also been recently reported in [39]. A full spectral unmixing chain [40], [41] comprising the automatic estimation of the number of endmembers using the VD [42] or the hyperspectral subspace identification with minimum error (HySime) [43], the identification of the endmember signatures using the N-FINDR algorithm [27], and quantification of endmember fractional abundances using unconstrained least-squares unmixing (ULS) [44] has been reported in [45], with speedups superior to $50\times$. A variation of this chain using the orthogonal subspace projection (OSP) [46] instead N-FINDR for endmember identification [47] was given in [48], achieving similar speedups and real-time unmixing results. Since ULS provides abundances that are not subject to constraints, a non-negative abundance estimation method called image space reconstruction algorithm (ISRA) [49], which was available in the form of a field programmable gate array (FPGA) implementation [50], has been recently implemented in multicore systems in [51]. Here, we will use a newly developed implementation of the ISRA algorithm for GPUs.

C. New Contributions of the Presented System

The digital repository that we present in this work is inspired by some of the aforementioned developments, but, at the same time, includes some new important features. These innovative contributions can be summarized as follows.

1) The presented system uses a full unmixing chain to perform the cataloguing and retrieval of hyperspectral images in the repository, as already discussed in [21] and [30]. However, the proposed implementation allows for more complex search criteria than the ones presented in previous contributions. Specifically, the queries in our system can be defined by the spectral information (provided by endmember signatures) and/or the spatial information (provided by abundances), in joint or separate fashion.

[1]http://science.gsfc.nasa.gov/606.1/docs/Specs.pdf.

[2]http://www.top500.org.

2) Another important contribution of our system is the possibility to use previously available spectral libraries as the main criteria in order to perform the query. In other words, our tool allows automatically loading a spectral library and using the spectra in such library in order to perform CBIR. The user may select a few spectra from the library or even the full library, using the selected spectral signatures as input to a query. Hence, the increased availability of open repositories of spectral libraries such as the SPECCHIO project[3] is a good complement to the system that we present in this contribution.
3) The processing modules included in our system comprise many well-established techniques in all parts of the full hyperspectral unmixing chain. Specifically, our system currently includes two methods for estimating the number of endmembers (VD and HySime), two algorithms for identifying the spectral signatures of the endmembers directly from the data (N-FINDR and OSP), and two methods for estimating the fractional abundances, unconstrained (ULS) and non-negatively constrained (ISRA). GPU implementations for all these methods are included in the system, thus allowing for fast cataloguing and metadata generation for new hyperspectral image scenes. It should be noted that the GPU implementation of ISRA was specifically developed as part of this contribution as it was not previously available.
4) Although the contribution [30] already discussed an efficient implementation of an unmixing-based CBIR system for hyperspectral imagery, this implementation was specifically developed for heterogeneous networks of workstations or clusters of computers, without taking advantage of hardware accelerators such as GPUs which are now widely available in modern clusters and supercomputers. In this regard, the proposed system expands the parallel features of the system in [30] and includes the use of GPU accelerators, thus increasing the computational performance significantly.
5) Last but not least, previous developments such as [21] or [30] were not fully available to the community. In this contribution, we present a fully open system, with an advanced user interface, and implemented on a large supercomputing facility: a cluster of GPUs available at the Center of Advanced Technologies in Extremadura (CETA-Ciemat), which is one of the most powerful clusters of GPUs in Spain. As a result, the CBIR system that we describe in this contribution is completely available for public use.[4] It contains several synthetic and real hyperspectral data sets that interested readers can use to conduct their own experiments and include additional hyperspectral data sets in the repository.

In the Section III, we describe the proposed system and its different layers. Experimental results with the synthetic and real hyperspectral data sets which are already included in the system will be reported in Sections III–V.

[3]http://www.specchio.ch.

[4]http://www.hypercomp.es/repository.

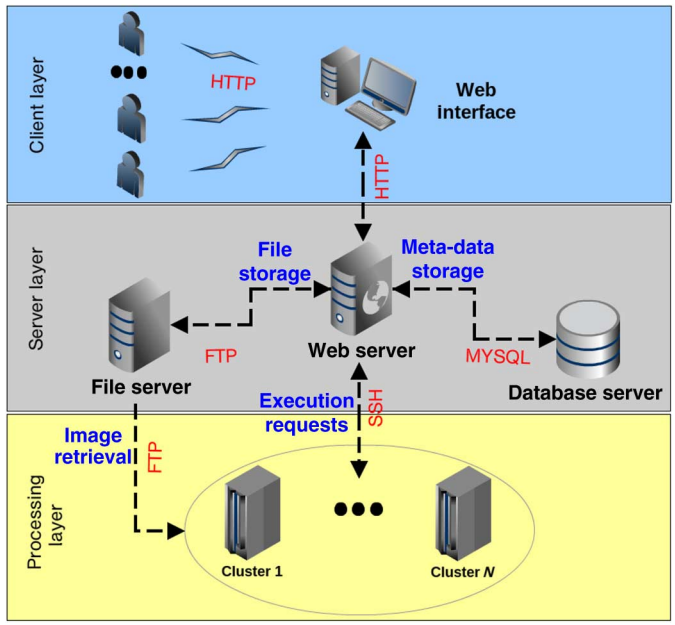


Fig. 1. Architecture of the proposed hyperspectral image repository.

III. Proposed CBIR System

The proposed system has been implemented as a web service composed of different layers. Its modular design provides quality of service and scalability and allows adding and/or modifying components without the need to modify the system. On the *client* layer, an intuitive web interface provides users with remote access to the system. On the *server* layer, the system provides data management and storage capabilities. On the *processing* layer, the system provides advanced processing functionalities for image cataloguing and retrieval. In order to accelerate the CBIR process, parallel processing functionalities and GPU implementations are also provided.

The remainder of this section, devoted to the description of our system, is organized as follows. In Section III-A, we describe the software architecture of the system and the different layers that compose it. In Section III-B, we describe the structure of the database that stores the hyperspectral data in our system. Section III-C describes how the queries to the database are carried out. Section III-D briefly outlines the parallel processing modules and GPU implementations available in our system, as well as their integration with the other components.

A. Software Architecture

As shown in Fig. 1, the software architecture of the proposed system is formed by different layers, which can be defined by their roles. The system follows a modular design in which the communication between layers is defined using standard data exchange formats and transfer protocols, so that any layer can be modified as long as it can communicate with the rest of the system. Our design has been carried out using free software tools such as `Symfony2`,[5] a full-stack web framework, while the adopted format for data exchange is JavaScript Object Notation

[5]http://symfony.com.

(JSON),[6] an open standard format that uses human-readable text to transmit data objects. In the following, we describe the different software layers that compose the system.

1) Client Layer: This layer defines the interactions between the user (through an Internet browser) and our system and is responsible for providing users secure remote access to the system. A web interface has been designed using HTML5,[7] the most widely used web programming language, and CSS3,[8] a standard style language that improves the HTML web appearance. In our software design of this layer, we also use jQuery[9] which is a JavaScript-based library designed to make responsive and dynamic websites and applications that are highly accessible. In addition, the interaction between the user and the web interface is captured by the events handlers of the jQuery libraries. The web interface transmits the request to the server layer via the hypertext transfer protocol (HTTP), an application protocol for distributed, collaborative, and hypermedia information systems, and the foundation of data communication for the World Wide Web. Most of the views are actually generated in the server, using Symfony2, a robust web development framework.

2) Server Layer: Our system is implemented as a web service. This means that its main purpose is to deliver web pages to a set of clients (i.e., the users of our system), while it also receives content from the clients. An advantage of this approach is that no additional software has to be installed on the client computer, since only a web browser is required. Many of the services provided by the system are managed and executed on the server layer, which is composed of several elements with different roles. As Fig. 1 shows, the web server handles web interface requests (via HTTP) and manages the system resources, such as meta-data storage and file data, in addition of handling algorithm executions. The server layer can be considered as the main engine of the system since it is in charge of managing and connecting the different components of our system. The server layer is also in charge of storing image meta-data, following a database schema that is described in Section III-A3. MySQL,[10] an open-source relational database manager, has been selected since it provides fast queries and low computational cost. This is a popular choice of database for use in web applications and has the advantage that many programming languages (such as C/C++, used for the development of our system) include libraries for accessing MySQL databases. On the other hand, a file storage server is also included in this layer for providing remote file access to any of the layers of the system. This module is also in charge of uploading and downloading file data, such as images and meta-data, via the file transfer protocol (FTP), a standard network protocol used to transfer files from one host to another such as, in our case, the intermet. The file server provides image data to the processing layer, which is described next.

3) Processing Layer: The processing layer is in charge of executing algorithms with high computational cost, mostly related with the cataloguing of new hyperspectral images in the system and the execution of queries for image retrieval. This layer relieves the web server load by providing high system availability. The processing layer is implemented in C/C++ using libraries to access parallel computing facilities, which efficiently execute requests coming from the web server. The execution requests are managed via secure shell (SSH), a network protocol for secure data communication. Currently, our processing layer supports efficient algorithm executions using a cluster of 44 GPUs at CETA-Ciemat[11] (called hereinafter CETA-GPU-Cluster). The resources of the cluster are managed by the SLURM[12] resource manager, selected because of its scalability, performance, and fault tolerance. Although not used in the current implementation, the system is also ready for multicluster support, as indicated in Fig. 1.

B. Database Structure

The structure of the database used to store hyperspectral images is illustrated in Fig. 2. The database has been carefully designed in order to store relevant information about the hyperspectral images which are stored in our system. In addition to standard information about each scene, such as number of samples, lines, bands, data type, byteorder, wavelength information or interleave, we also store additional information such as the endmembers and abundances associated to each scene (meta-data), as well as additional (optional) information such as the results and publications in which a certain hyperspectral scene has been addressed, or the results obtained for the scene by different algorithms. The meta-data are automatically generated by the system using unmixing algorithms that are used to catalog each scene, hence the procedure for uploading a new data set to the system only requires basic information about the scene. In addition, relevant information about previous analyses and experiments with each scene can also be stored in the database if available, while the unmixing results are used to generate meta-data that can be then used for retrieval purposes using different queries. In Section III-C, we describe the structure of the queries that can be carried out in order to retrieve hyperspectral scenes from the database.

C. Queries

Our CBIR system allows an end-user to perform queries to the hyperspectral image database described in Fig. 2. For each new hyperspectral data set, the spectral endmembers and their corresponding abundance maps can be obtained using a set of algorithms implemented in the system. These algorithms conform a full spectral unmixing chain made up of three steps: 1) the number of endmembers to be extracted from the image can be automatically calculated using the VD [24] or the hyperspectral subspace identification with minimum error (HySime) [43] methods; 2) the endmember signatures can be extracted using two different methods: N-FINDR or an OSP [46] method implemented using Gram–Schmidt (GS) orthogonalization [48]; and 3) abundance estimation can be either conducted using ULS

[6]http://www.json.org.

[7]http://www.w3.org/TR/html51.

[8]http://www.w3.org/Style/CSS.

[9]http://jquery.com.

[10]http://www.mysql.com.

[11]http://www.ceta-ciemat.es.

[12]https://computing.llnl.gov/linux/slurm.

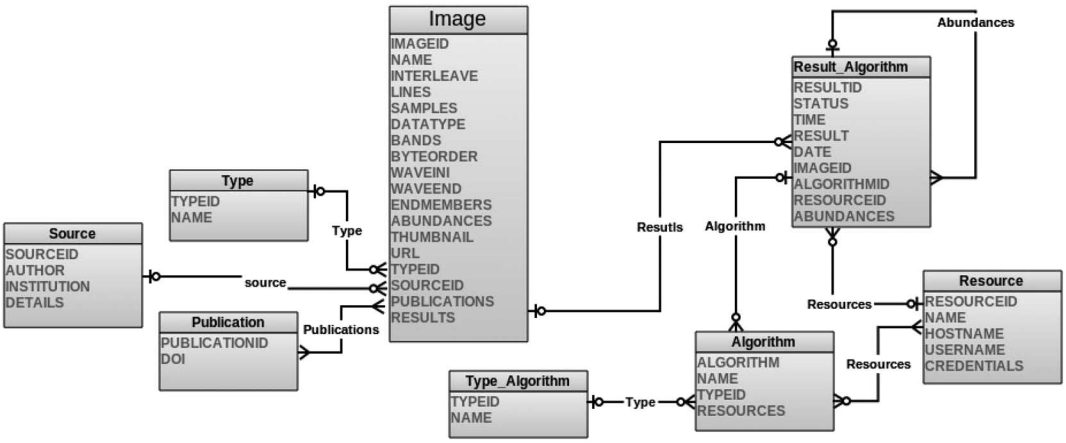


Fig. 2. Structure of the database used to store hyperspectral images in our system.

or ISRA. The information provided by endmember identification and abundance estimation is then used as meta-data, to catalog each image stored in the database. This allows for fast CBIR functionality, as the meta-data provide a compact representation of each scene in the database, and the full unmixing chain is implemented efficiently in parallel (exploiting also the GPUs available in the system). In the following, we describe the searching methodology including aspects such as distance metrics and matching algorithms used for retrieval purposes. This section concludes with a description of the searching strategy from the end-user's point of view, with particular emphasis on the web interface developed and the searching options available in the system.

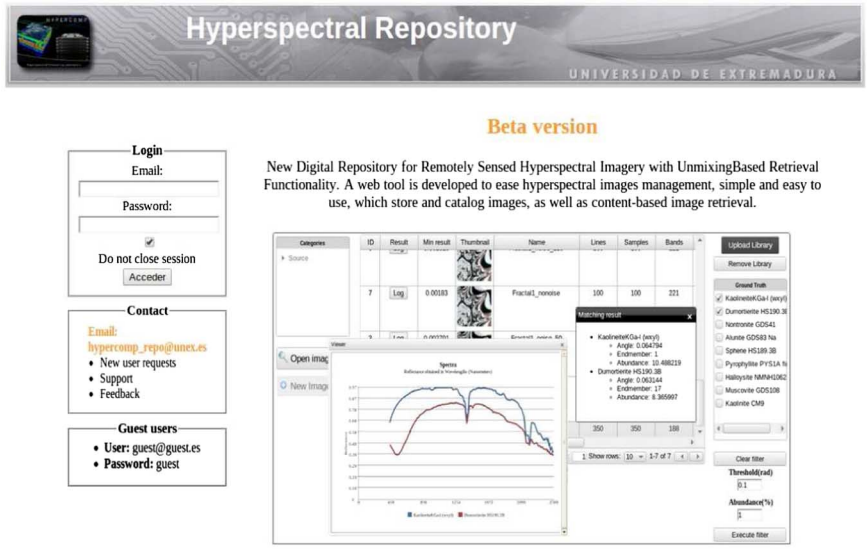
1) Searching Methodology: Two searching options are available in the proposed system. The first one relies on the SAD in order to retrieve hyperspectral images with endmembers that are similar to those available in a spectral library that can be loaded in the system. A particular issue that may arise in this kind of search is the fact that the wavelengths of the spectral library used as input can be different from the wavelengths of the hyperspectral images stored in the system. For this purpose, we have implemented a spectral convolution strategy that looks for wavelength values which are present in both the hyperspectral data and the input spectral library (with the possibility to include a tolerance threshold in the wavelength matching procedure). In most cases, the spectral resolution of the input signatures in the spectral library is much higher than the images stored in the database, and it is often possible to retrieve scenes with great accuracy as their associated wavelengths are a subset of those of the signatures in the spectral library.

On the other hand, the system also allows queries based on the specific abundance of a given endmember. For instance, we may not only look for scenes with a specific kind of vegetation (endmember), but also with a significant presence (abundance) of this kind of vegetation in the retrieved scene. For this purpose, we may set a minimum threshold for the abundance of a given endmember (or group of endmembers) in the retrieved scene. This is implemented by calculating the total abundance of each of the endmembers in the scene, summing all the relative contributions in each pixel and obtaining the total abundance coverage of an endmember in the scene. Then, we may impose a minimum abundance threshold that is used in the retrieval process. In this way, we can effectively perform image retrieval based on both endmember and abundance information (i.e., not only retrieving scenes that contain a certain endmember but also scenes that contain a certain amount of a given endmember). Since this information is available as meta-data for each scene, the retrieval process is quite fast and the most computationally expensive part is the generation of the meta-data itself, which is carried out in parallel as will be explained at the end of this section.

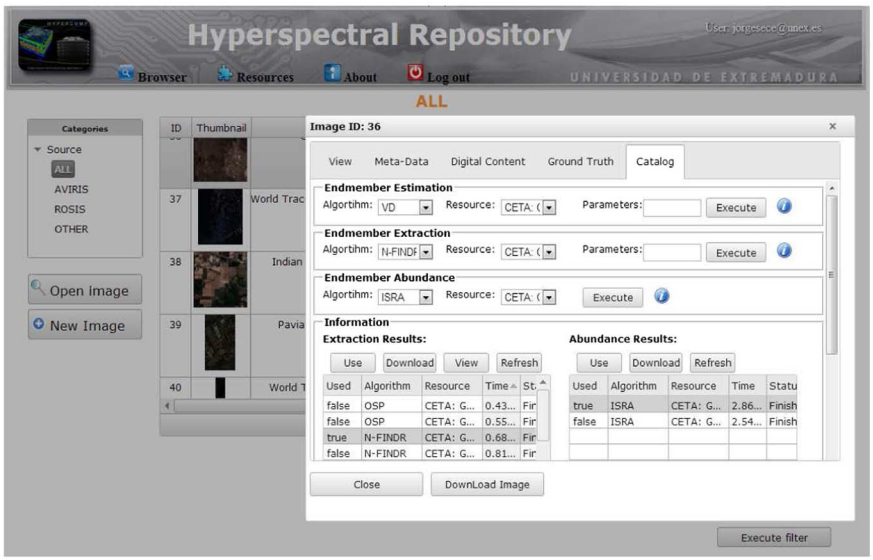
2) Search Procedure: From an end-user's point of view, a standard search procedure in our system can be summarized by the following steps.

1) *Initialization*. In this step, a spectral library of signatures used for the search is loaded in the system.
2) *Spectral convolution*. The system automatically performs a spectral convolution strategy that allows comparing the wavelengths of the input spectral library with the wavelenghts of the real hyperspectral data stored in the system.
3) *Signature comparison*. For a spectral signature (or set of signatures) available in the loaded spectral library, the system calculates the SAD with all the endmembers stored as meta-data for each hyperspectral scene in the system, and retrieves a number of matching scenes satisfying the specified criterion.
4) *Abundance filter*. As an optional step, the system allows defining a minimum abundance filter which is used as an additional condition to the signature comparison described in the previous step. In this case, the image is retrieved only if the matching endmember contains a total abundance in the scene that is higher than a minimum predefined abundance threshold.
5) *Sorting and visualization*. The retrieved images are shown to the user sorted from higher to lower spectral similarity (lowest to highest spectral angle).

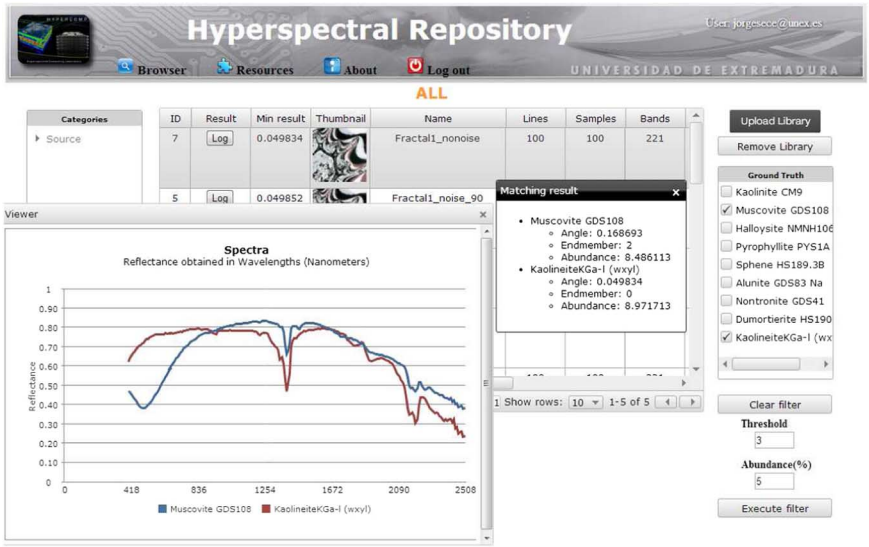
In the following, we provide a simple step-by-step example illustrating how to perform a simple hyperspectral image retrieval task in our system. Fig. 3(a) shows a general overview of our system, which allows guest access (this option does not allow uploading new images in the system). Full access to the system is also available upon request, including the image uploading functionality. The system is now fully operational, although still



(a)



(b)



(c)

Fig. 3. (a) General overview of the system. (b) Catalog panel of the system. (c) Example of a query.

in beta version, and allows any interested user to obtain a fully operational account.

Fig. 3(b) shows the catalog panel of the system, which allows for automatically extracting meta-data for each new hyperspectral scene that is uploaded in the system. The user can decide between two algoritms for estimating the number of endmembers (VD and HySime), two algorithms for identifying endmember signatures (N-FINDR and OSP-GS), and two algorithms for estimating the endmember abundance signatures (LSU and ISRA). The user can also decide in which computing resource the algorithms will be run. If the option CETA-GPU-CLUSTER is activated, the algorithms will be executed in the cluster of GPUs. After the algorithms have been executed, the results obtained are visualized and the user can decide the best combination of algorithms in order to catalog the hyperspectral scene. Once the procedure is completed, the hyperspectral image will be automatically cataloged using the meta-data obtained, and stored in the database structure described in Fig. 2.

Fig. 3(c) shows an example of a query, in which the United States Geological Survey (USGS) library[13] is loaded in our system and two specific spectral signatures (muscovite and kaolinite) are used to define the query. The spectral similarity threshold is set to 3 degrees and the minimum abundance is set to 5% (this means that we are looking for hyperspectral scenes containing at least 5% muscovite and 5% kaolinite). The system now provides information on the images retrieved. For the first one in the list, the system estimates 8.48% of muscovite and 8.97% of kaolinite. In this case, the spectral similarity scores are very high, with less than one degree in the spectral similarity test for both endmembers. As a result, the end-user can infer that this scene accurately satisfies the search criterion. Since there are other scenes retrieved, the end-user may decide to select another hyperspectral image retrieved by the query (the images are ordered according to the combined spectral similarity score resulting from the query).

D. Parallel Implementations

As mentioned above, the extraction of meta-data for a given hyperspectral image has been efficiently implemented in a cluster of GPUs called CETA-GPU-CLUSTER. The GPU implementations have been carried out using the compute unified device architecture (CUDA) developed by NVidia. The architecture of a GPU can be abstracted as a set of multiprocessors (MPs), in which each MP is characterized by a single instruction multiple data (SIMD) architecture, i.e., in each clock cycle, each processor executes the same instruction but operating on multiple data streams. Each processor accesses a local shared memory and also local cache memories in the MP, while the MPs have access to the global GPU (device) memory. GPUs can be, therefore, abstracted in terms of a *stream model*, under which all data sets are represented as streams (i.e., ordered data sets). Algorithms are constructed by chaining so-called kernels which operate on entire streams and which are executed by an MP, taking one or more streams as inputs and producing one or more streams as outputs. Thereby, data-level parallelism is exposed to hardware,

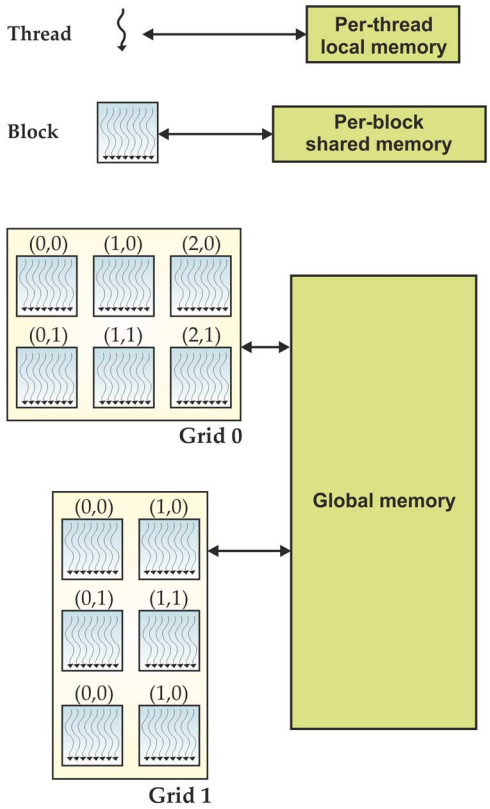


Fig. 4. Different levels of memory in the GPU for the thread, block, and grid concepts.

and kernels can be concurrently applied without any sort synchronization. The kernels can perform a kind of batch processing arranged in the form of a grid of blocks, where each block is composed by a group of threads that share data efficiently through the shared local memory and synchronize their execution for coordinating accesses to memory (see Fig. 4). It should be noted that all implementations have been carried out using only one of the GPUs available in the cluster, since the complexity of the scenes currently stored in the system allows completing the cataloguing in a few seconds only. However, the system allows using the full cluster of 44 GPUs available if needed, using a hybrid implementation based on `MPI`[14] and CUDA. Additional details about the GPU implementations of the algorithms currently used by the system for cataloguing purposes can be found in [52] and [53].

IV. Experimental Results

The performance of the proposed unmixing-based CBIR system has been evaluated from two different perspectives: 1) its ability to retrieve hyperspectral images of interest from the set of cataloged ones available in the system; and 2) the efficiency in cataloguing and retrieving hyperspectral images. Experiments have been conducted using both synthetic images (in a fully controlled environment) and also representative real hyperspectral images. The remainder of the section is organized as follows. First, we describe the synthetic and real hyperspectral data sets

[13]http://speclab.cr.usgs.gov.

[14]http://www.mcs.anl.gov/research/projects/mpi.

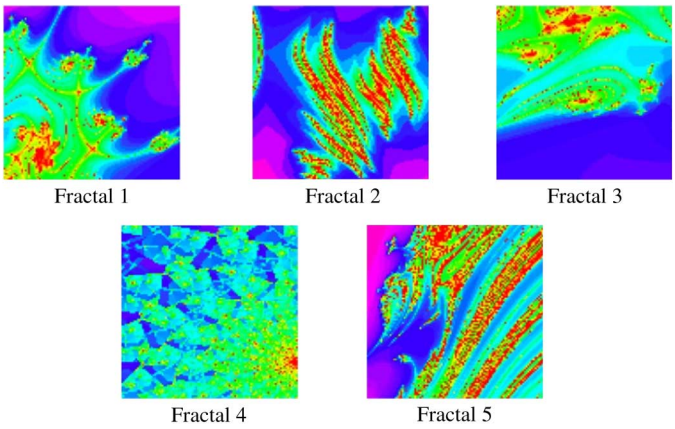


Fig. 5. Fractal images used in our simulations.

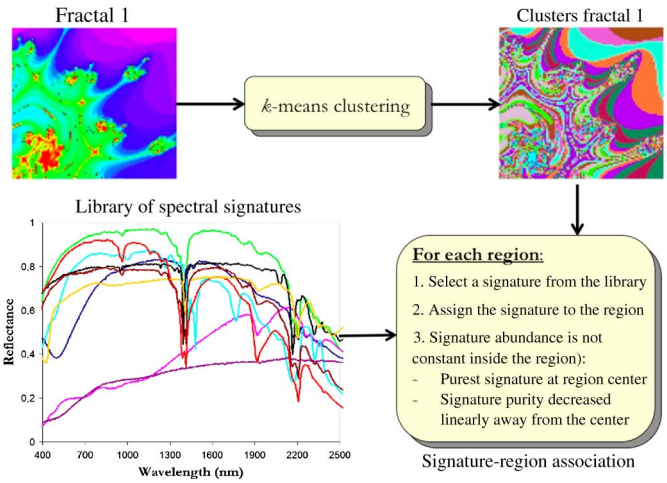


Fig. 6. Procedure used for generating a synthetic hyperspectral image from a fractal image.

used in our experiments. Then, we perform an evaluation of the system from the viewpoint of retrieval accuracy. Finally, we perform an evaluation from the viewpoint of computational efficiency.

A. Hyperspectral Data

1) Synthetic Data: The main reason for using synthetic data in our evaluation of retrieval accuracy is that these kind of images can be generated in a fully controlled environment. As a result, algorithm accuracy can be effectively validated and tested. In this work, we have used a set of synthetically generated images using fractals. We have selected fractals because they can simulate naturally occurring patterns in nature. For illustrative purposes, Fig. 5 displays the five fractal images used in our simulations. The procedure for generating one of our simulated images is depicted in Fig. 6, in which a fractal image is used to simulate spatial patterns. The k-means clustering algorithm is adopted to select a set of clusters from the fractal image. Then, a procedure starts which assigns a set of spectral signatures from a spectral library to each region resulting from the clustering step mentioned before. A crucial step in the simulation procedure is how to assign a spectral signature to each cluster. For this purpose, we have implemented an automatic procedure that follows a simple strategy, in which $p = 9$ signatures are first assigned to spatially disjoint regions belonging to different clusters. The remaining regions are then assigned spectral signatures in an automatic way, ensuring that: 1) spatially adjacent clusters always have different signatures associated to them; and 2) there is a balance among the overall number of pixels in the image which are associated to each spectral signature. Inside each region, the abundance proportions of spectral signatures have been generated following a procedure that tries to imitate reality as much as possible, i.e., those pixels closer to the borders of the regions are more heavily mixed, while the pixels located at the center of the regions are more spectrally pure in nature. This is accomplished by linearly mixing the signature associated to each cluster with those associated to neighboring clusters, making sure that the most spectrally pure signature remains at the center of the region while signature purity decreases linearly away from the center to the borders of the regions. With the aforementioned procedure, the simulated regions exhibit the following properties.

1) All the simulated pixels inside a region are mixed, and the simulated image does not contain completely pure pixels. This increases the complexity of the unmixing problem and simulates the situation often encountered in real-world analysis scenarios, in which completely pure pixels are rarely found.
2) Pixels close to the borders of the region are more heavily mixed than those in the center of the region.
3) If the simulated region is sufficiently large, the pixels located at the center can exhibit a degree of purity of 99% of a certain endmember. However, if the size of the simulated region is small, the degree of purity of pixels at the center of the region can decrease until 95% of a certain endmember, while pixels located in the region borders are generally more heavily mixed.

To conclude the simulation process, zero-mean Gaussian noise was added to the scenes in different signal-to-noise ratios (SNRs) of 10:1, 30:1, 50:1, 70:1, 90:1, and 110:1 to simulate contributions from ambient and instrumental sources, following the procedure described in [54]. For illustrative purposes, Fig. 6 shows the spectra of the United States Geological Survey (USGS) library[15] used in the simulation of one of the synthetic scenes (labeled as "Fractal 1"). In total, we considered five different fractal images (first simulated without noise, i.e., with $\mathrm{SNR} = \infty$) and then six different versions of each scene corrupted with different noise levels, which gives a total of 30 synthetic images, all of them available in our system for public use.

2) Real Data: In addition to the 30 synthetic images described in Section IV-A, our repository has currently 12 additional real hyperspectral images for a total of 42 hyperspectral images and total space of about 2 GB. It is worth noting that the repository is ready to receive additional scenes from end-users so that the database can grow. For the experiments that will be described in Section IV-B, we have considered three well-known hyperspectral images with reference information (and which have been widely used in recent hyperspectral imaging literature) in order to substantiate both the retrieval accuracy

[15]http://www.speclab.cr.usgs.gov.

TABLE II
SAD (DEGREES) AND ROOT MEAN RECONSTRUCTION ERROR (RMSE) IN ABUNDANCE ESTIMATION OBTAINED USING NINE USGS MINERAL SPECTRA (THOSE USED TO CONSTRUCT THE FRACTAL SYNTHETIC SCENES USED IN OUR EXPERIMENTS) AS INPUT TO A QUERY ON OUR DATABASE OF HYPERSPECTRAL SCENES

USGS signature	Similarity metric	Signal to noise ratio						AVIRIS Cuprite
		10:1	30:1	50:1	70:1	110:1	No noise	
Kaolinite CM9	SAD	14.443	3.128	2.844	1.153	1.639	1.764	5.251
	RMSE	0.185	0.174	0.173	0.172	0.173	0.174	–
Muscovite GDS108	SAD	15.261	1.701	0.236	0.169	0.169	0.169	3.618
	RMSE	0.197	0.171	0.176	0.173	0.171	0.173	–
Halloysite NMNH106236	SAD	25.436	2.947	0.366	0.279	0.394	0.277	17.334
	RMSE	0.229	0.125	0.127	0.144	0.164	0.127	–
Pyrophyllite PYS1A	SAD	14.233	3.061	0.318	10.055	0.048	0.048	8.194
	RMSE	0.231	0.219	0.222	0.249	0.223	0.219	–
Sphene HS189.3B	SAD	27.533	7.156	0.766	12.246	16.53196	0.348	6.634
	RMSE	0.274	0.223	0.226	0.281	0.288	0.224	–
Alunite GDS83 Na	SAD	15.054	1.529	0.157	0.127	0.126	0.065	8.304
	RMSE	0.166	0.163	0.162	0.163	0.163	0.162	–
Nontronite GDS41	SAD	16.124	1.605	0.223	0.136	0.134	0.135	13.615
	RMSE	0.158	0.128	0.128	0.133	0.126	0.129	–
Dumortierite HS190.3B	SAD	35.379	1.658	0.518	0.506	0.507	0.507	3.083
	RMSE	0.166	0.140	0.140	0.141	0.141	0.141	–
KaolineiteKGa-l (wxyl)	SAD	19.936	1.592	0.182	0.051	0.050	0.050	10.215
	RMSE	0.179	0.163	0.163	0.164	0.161	0.163	–

The similarity results obtained for the five synthetic scenes (on average) with different noise ratios and for the AVIRIS Cuprite scene are reported. In all cases, the scenes were cataloged using a spectral unmixing chaing given by VD+N-FINDR+ISRA.

and parallel performance of the proposed CBIR system. Since the images comprise different analysis scenarios, sizes, and properties, our selection is expected to be sufficiently heterogeneous to provide an evaluation of the system from different perspectives.

1) The first scene used in our experiments is the AVIRIS Indian Pines data set, which comprises 145 lines, 145 samples, and 220 spectral channels between 400 and 2500 nm, and a total size of around 9 MB. This scene has been widely used as a benchmark in classification applications, and contains detailed ground-truth in the form of a ground-truth map with 16 mutually exclusive classes.
2) The second scene used in our experiments is the AVIRIS Cuprite data set, comprising 350 lines, 350 samples, and 188 spectral channels between 400 and 2500 nm and a total size of around 50 MB. The spectral signatures of the minerals comprised by this scene are available in the USGS spectral library. This scene has been widely used as a benchmark in spectral unmixing applications [55].
3) The third scene used in our experiments is the AVIRIS World Trade Center data set, comprising 512 lines, 614 samples, and 224 spectral channels between 400 and 2500 nm and a total size of around 140 MB. This scene, which has been widely used as a benchmark in target and anomaly detection applications, comprises reference information available in several forms.[16]

B. Evaluation of Image Retrieval Accuracy

In order to illustrate the performance of our CBIR system, we specifically address a case study in which the synthetic images described in Section IV-A are used to substantiate retrieval accuracy using different noise conditions. For illustrative purposes, we also use the AVIRIS Cuprite scene to evaluate retrieval accuracy with real hyperspectral data. The two metrics that we have used in our experiments (and which are included in the presented system) are the SAD in endmember comparison and the root-mean-square error (RMSE) in the estimated abundance fractions [31]. These two metrics are widely used in the hyperspectral unmixing literature and constitute the main evaluation metrics included in our system.

[16]http://www.speclab.cr.usgs.gov/wtc.

TABLE III
PROCESSING TIMES (IN SECONDS) AND SPEEDUPS ACHIEVED FOR THE GPU IMPLEMENTATION OF VD, N-FINDR, AND ISRA ALGORITHMS USED TO CATALOG THREE REAL HYPERSPECTRAL SCENES AVAILABLE IN OUR CBIR SYSTEM

		VD			N-FINDR				ISRA		
		Initialization	VD	Total	Initialization	PCA	N-FINDR	Total	Initialization	ISRA	Total
AVIRIS Indian Pines	CPU	0.063	3.248	3.311	0.021	1.742	0.937	2.700	0.025	44.253	44.278
		± 0.057	± 0.252	± 0.252	± 0.005	± 0.021	± 0.260	± 0.256	± 0.003	± 0.513	± 0.514
	GPU	0.498	0.027	0.525	0.512	0.021	0.245	0.778	0.447	0.432	0.879
		± 0.084	± 0.001	± 0.084	± 0.0136	± 0.008	± 0.140	± 0.140	± 0.089	± 0.001	± 0.089
	Speedup	–	120.296	1.395	–	82.952	3.824	3.470	–	102.437	14.745
AVIRIS Cuprite	CPU	0.340	14.476	14.816	0.125	9.069	2.434	11.628	0.149	120.650	120.799
		± 0.158	± 2.134	± 2.221	± 0.020	± 0.064	± 0.440	± 0.480	± 0.021	± 0.355	± 0.355
	GPU	0.579	0.127	0.706	0.583	0.075	0.134	0.792	0.535	2.183	2.718
		± 0.147	± 0.001	± 0.148	± 0.009	± 0.002	± 0.032	± 0.032	± 0.098	± 0.001	± 0.098
	Speedup	–	113.094	5.399	–	120.920	18.164	14.682	–	55.268	24.855
AVIRIS WTC	CPU	0.934	57.629	58.563	0.381	35.210	16.422	52.013	0.521	885.336	885.857
		± 0.573	± 8.678	± 8.913	± 0.496	± 0.199	± 5.925	± 5.919	± 0.066	± 45.971	± 46.006
	CPU	0.683	0.279	0.962	0.710	0.216	0.837	1,763	0.679	4.460	5.139
		± 0.084	± 0.001	± 0.169	± 0.010	± 0.006	± 0.907	± 0.908	± 0.126	± 0.014	± 0.131
	Speedup	–	202.919	18.064	–	163.009	19.620	29.502	–	198.506	119.131

The table reports the mean values and the standard deviations measured across 10 algorithm executions.

Table II shows the SAD and RMSE results obtained after using a query based on the nine USGS spectral signatures that were used to construct the fractal synthetic images: KaoliniteKGA-l(wxyl), Dumortierite HS190.3B, Nontronite GDS41, Alunite GDS83 Na, Sphene HS189.3B, Pyrophyllite PYS1A, Halloysite NMNH10623, Muscovite GDS108, and Kaolinite CM9. The algorithms used to catalog the scenes were VD, N-FINDR, and ISRA. As shown in Table II, the SAD and RMSE scores obtained for the synthetic scenes generally decrease as the amount of simulated noise is lower, with SAD scores well below 10° for SNR levels of 30:1 or below. Since the worst case for the SAD is 90°, this is considered to be a good similarity score. In other words, the SAD metric reveals that our system can effectively retrieve the scenes containing endmembers which are highly similar, spectrally, to those used in the set of input signatures used to launch the query. The results reported in Table II for the RMSE metric follow a similar pattern. These results could not be obtained for the AVIRIS Cuprite scene since the ground-truth abundances are difficult to obtain in real scenarios, but in this case we used the RMSE between the original and the reconstructed scene using the endmembers and abundances derived by the considered unmixing chain. The overall reconstruction error obtained in this case was very low (0.158) [56], indicating that this strategy can also be used as a retrieval criterion including abundance information in the case that no ground-truth information is available.

C. Evaluation of Parallel Performance

The computational performance of the proposed CBIR system has been evaluated using the CETA-GPU-CLUSTER with 44 NVidia TESLA C2050[17] GPUs, each of which features 448 streaming processor cores with 1.15 GHz, with single precision floating point performance of 1.03 Tflops, double precision floating point performance of 515 Gflops, total dedicated memory of 3 GB, and memory bandwidth of 144 GB/s. Each GPU is connected to a multicore CPU of type Quad Core Intel Xeon at 2.26 GHz with four physical cores, and 24 GB of DDR3 SRAM memory. The GPU is mounted on a Bullx R422.[18] Before describing our results, it is important to emphasize that our GPU versions provide exactly the same results as the serial versions of the algorithms, implemented using the `gcc` (gnu compiler default) with optimization flag –O3. The serial algorithms were executed in one of the available cores of the system. Since the hyperspectral images contained in our repository can all fit the video memory of a single GPU (3 GB), we report the processing results obtained in a single GPU, although our system is ready to use the full cluster with 44 GPUs in parallel if needed. This is mainly because the current volume of stored hyperspectral data makes computations manageable with a single GPU. Since the computational requirements of the system can be currently managed with one GPU unit, we have decided to provide results based on the utilization of a single GPU. In the future, if the system grows as we expect with the addition of new data sets from external users, we may need to resort to a multi-GPU implementation. This is perfectly feasible, since the system has been designed with this configuration in mind, although currently we only need to use one GPU device. Another important consideration is that working in a single GPU reduces communication time, particularly if the scenes can be allocated into a single GPU memory. However, the communication overheads for a full multi-GPU implementation could be significant and it would be necessary to fully test our implementation in this scenario. In any event, we expect the searching part to scale

[17]http://www.nvidia.co.uk/object/product_tesla_C2050_C2070_uk.html.

[18]http://www.bull.com/catalogue/details.asp?tmp=bxs-rack-fr&opt=ns-r422e02&dt=ft&cat=bullx.

TABLE IV
PROCESSING TIMES (IN SECONDS) AND SPEEDUPS ACHIEVED FOR THE GPU IMPLEMENTATION OF HYSIME, OSP-GS, AND ULS ALGORITHMS USED TO CATALOG THREE REAL HYPERSPECTRAL SCENES AVAILABLE IN OUR CBIR SYSTEM

		HYSIME			OSP-GS			ULS		
		Initialization	HYSIME	Total	Initialization	OSP-GS	Total	Initialization	ULS	Total
AVIRIS Indian Pines	CPU	0.300	13.124	13.154	0.039	0.552	0.591	0.035	0.29	0.325
		± 0.025	± 3.164	± 3.164	± 0.002	± 0.064	± 0.062	± 0.003	± 0.050	± 0.001
	GPU	0.450	0.194	0.643	0.523	0.009	0.532	0.479	0.017	0.496
		± 0.368	± 0.013	± 0.375	± 0.400	± 0.001	± 0.400	± 0.342	± 0.001	± 0.542
	Speedup	–	63.610	64.122	–	61.333	1.111	–	17.059	0.655
AVIRIS Cuprite	CPU	0.512	63.610	63.122	0.198	2.225	2.423	0.160	1.500	1.660
		± 0.134	± 0.231	± 0.274	± 0.015	± 0.016	± 0.022	± 0.019	± 0.012	± 0.031
	GPU	0.567	0.907	1.464	0.564	0.024	0.588	0.493	0.054	0.547
		± 0.385	± 0.008	± 0.389	± 0.202	± 0.001	± 0.202	± 0.306	± 0.001	± 0.507
	Speedup	–	70.132	43.799	–	92.708	4.121	–	27.778	3.035
AVIRIS World Trade Center	CPU	0.349	240.027	240.376	0.495	18.620	19.115	0.445	16.455	16.900
		± 0.336	± 64.367	± 64.337	± 0.040	± 0.019	± 0.055	± 0.056	± 0.069	± 0.103
	GPU	0.678	3.245	3.923	0.612	0.106	0.718	0.651	0.232	0.883
		± 0.312	± 0.011	± 0.325	± 0.136	± 0.001	± 0.136	± 0.385	± 0.001	± 0.385
	Speedup	–	73.968	61.274	–	175.660	26.622	–	70.927	19.139

The table reports the mean values and the standard deviations measured across 10 algorithm executions.

properly with the GPU number since our search strategy is based on comparing the spectral endmembers and the abundance fractions using standard distance metrics. The most challenging part would be the cataloguing of a new scene using different endmember extraction techniques, as it will depend on the considered endmember extraction strategy, but in this case the cataloguing will take place only once (at the beginning) and the results would be then immediately available for searching purposes.

In the following, we analyze the processing times (in seconds) used by our system in the process of cataloguing three real hyperspectral scenes using two different unmixing chains. In each experiment, 10 runs were performed and the mean values were reported. We include the initialization times in our experiments since, as mentioned before, these times are generally higher in a GPU cluster rather than in single GPUs' devices, which is due to the fact that the communications between the nodes of the cluster introduce a slight delay in the process. Table III shows the processing times of the first unmixing, based on VD for identification of the number of endmembers, N-FINDR for endmember signature finding, and ISRA for nonnegative abundance estimation. In the case of N-FINDR, a dimensionality reduction of the original scene using PCA is conducted, hence we report the times for the PCA and for the N-FINDR algorithm in this case. Furthermore, Table IV shows the timing results obtained by a second unmixing chain made up of HySime for finding the number of endmembers, OSP-GS for extracting the endmember signatures, and ULS for unconstrained abundance estimation. In both Tables III and IV, we display the CPU and GPU times (including the initialization required in the GPU cluster) and the speedup of the GPU version over the CPU one. As shown by the Tables III and IV, using only one GPU of the cluster can already significantly accelerate the cataloguing process for the three considered scenes, which takes only a few seconds in all cases and with significant speedups. The initialization times (retained in our results in order to give an idea of the performance of the system in a GPU cluster, even if only one GPU is really used for the calculations) are not significant. This means that, once a new hyperspectral scene has been uploaded into our system, its associated meta-data can be efficiently generated in automatic fashion for subsequent retrieval. It is also worth noting that the results of the cataloguing can be dynamically selected, i.e., the end-user may decide to use any algorithm combination to catalog a scene. In this regard, Tables III and IV show two among many possibilities that can be derived by different combinations of the available blocks to conform different unmixing chains. Also, as Tables III and IV show, the speedups obtained increase with the hyperspectral image size.

V. CONCLUSIONS AND FUTURE LINES

In this paper, we have presented a new digital repository for hyperspectral image data that allows uploading and retrieving images through a CBIR functionality based on spectral unmixing concepts. The current version is implemented as a web service, which allows remote user access through a web interface while a server is in charge of managing the repository database and performing algorithm executions. The most computationally expensive operations (such as meta-data generation) are efficiently implemented in parallel. We have used well-known algorithms in the spectral unmixing community in order to generate suitable meta-data for retrieval purposes, based on relatively complex queries including both endmember and abundance information. Our main focus on this work has been on describing the different software layers of the system, as well as on conducting a detailed evaluation based on synthetic and real data. The synthetic data sets allowed us to evaluate the

performance of the system in a fully controlled scenario, while the real image data sets provided a practical illustration of the system with widely used hyperspectral data sets. The proposed system is fully available for public use and represents a first step toward a standardized hyperspectral data repository data intended to distribute and share hyperspectral data sets in the community. The proposed system is expected to increase the value of the data acquired by available and new airborne/satellite hyperspectral imaging instruments, and to improve the availability of the data and its associated information. As future extension of the system, we will implement multicore versions of the algorithms used to implement the system and explore in more details its multi-GPU functionality. Specifically, a multicore implementation would perhaps exhibit less restrictions in terms of communication overhead than a GPU cluster, due to the availability of shared memories. In both cases (multi-GPU and multicore), our system is expected to adapt quite well to these architectures since the searching part can be efficiently performed in parallel and the cataloguing part (in spite of the fact that it would depend on the considered endmember extraction implementation in multiple processors) would only need to be used once for each scene. In addition, we are planning on including other algorithms such as sparse unmixing techniques, which could be useful to improve the queries based on large spectral libraries.

Acknowledgment

The authors gratefully thank the associate editor who handled our manuscript and the two anonymous reviewers, who provided outstanding comments and suggestions that greatly helped us to significantly improve the technical content and presentation of our manuscript.

References

[1] A. W. M. Smeulders, M. Worring, S. Santini, A. Gupta, and R. Jain, "Fast dimensionality reduction and simple PCA," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 12, pp. 1349–1380, Dec. 2000.

[2] A. Plaza and C.-I. Chang, *High Performance Computing in Remote Sensing*. New York, NY, USA: Taylor & Francis, 2007.

[3] A. F. H. Goetz, G. Vane, J. E. Solomon, and B. N. Rock, "Imaging spectrometry for Earth remote sensing," *Science*, vol. 228, pp. 1147–1153, 1985.

[4] A. Plaza, J. Benediktsson, J. Boardman, J. Brazile, L. Bruzzone, G. Camps-Valls, J. Chanussot, M. Fauvel, P. Gamba, A. Gualtieri *et al.*, "Recent advances in techniques for hyperspectral image processing," *Remote Sens. Environ.*, vol. 113, pp. S110–S122, 2009.

[5] J. McKeown, D. M. S. Cochran, S. Ford, J. McGlone, J. Shufelt, and D. Yocum, "Fusion of HYDICE hyperspectral data with panchromatic imagery for cartographic feature extraction," *IEEE Trans. Geosci. Remote Sens.*, vol. 37, no. 3, pp. 1261–1277, May 1999.

[6] 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, M. R. Olah, and O. Williams, "Imaging spectroscopy and the airborne visible/infrared imaging spectrometer (AVIRIS)," *Remote Sens. Environ.*, vol. 65, no. 3, pp. 227–248, 1998.

[7] E. Middleton, S. Ungar, D. Mandl, L. Ong, S. Frye, P. Campbell *et al.*, "The Earth observing one (EO-1) satellite mission: Over a decade in space," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 6, no. 2, pp. 243–256, Apr. 2013.

[8] K. Segl, L. Guanter, C. Rogass, T. Kuester, S. Roessner, H. Kaufmann *et al.*, "EeteS: The EnMAP end-to-end simulation tool," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 5, no. 2, pp. 522–530, Apr. 2012.

[9] C. Galeazzi, A. Sacchetti, A. Cisbani, and G. Babini, "The PRISMA program," in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, 2008, vol. 4, pp. 105–108.

[10] M. Barnsley, J. Settle, M. Cutter, D. Lobb, and F. Teston, "The PROBA/CHRIS mission: A low-cost smallsat for hyperspectral multiangle observations of the Earth surface and atmosphere," *IEEE Trans. Geosci. Remote Sens.*, vol. 42, no. 7, pp. 1512–1520, Jul. 2004.

[11] M. Abrams, D. Pieri, V. Realmuto, and R. Wright, "Using EO-1 Hyperion data as HyspIRI preparatory data sets for volcanology applied to Mt Etna, Italy," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 6, no. 2, pp. 375–385, Apr. 2013.

[12] H. Wu, L. Ni, N. Wang, Y. Qian, B.-H. Tang, and Z.-L. Li, "Estimation of atmospheric profiles from hyperspectral infrared IASI sensor," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 6, no. 3, pp. 1485–1494, Jun. 2013.

[13] J. Bioucas-Dias, A. Plaza, N. Dobigeon, M. Parente, Q. Du, P. Gader *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.

[14] A. Plaza, J. Plaza, A. Paz, and S. Sánchez, "Parallel hyperspectral image and signal processing," *IEEE Signal Process. Mag.*, vol. 28, no. 3, pp. 119–126, May 2011.

[15] A. Plaza, Q. Du, Y.-L. Chang, and R. King, "High performance computing for hyperspectral remote sensing," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 4, no. 3, pp. 528–544, Sep. 2011.

[16] C. Lee, S. Gasster, A. Plaza, C.-I. Chang, and B. Huang, "Recent developments in high performance computing for remote sensing: A review," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 4, no. 3, pp. 508–527, Sep. 2011.

[17] N. Chen, Z. Chen, L. Di, and J. Gong, "An efficient method for near-real-time on-demand retrieval of remote sensing observations," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 4, no. 3, pp. 615–625, Sep. 2011.

[18] D. Brunner, G. Lemoine, F.-X. Thoorens, and L. Bruzzone, "Distributed geospatial data processing functionality to support collaborative and rapid emergency response," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 2, no. 1, pp. 33–46, Mar. 2009.

[19] M. Stasolla and P. Gamba, "Spatial indexes for the extraction of formal and informal human settlements from high-resolution SAR images," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 1, no. 2, pp. 98–106, Jun. 2008.

[20] S. S. Durbha, R. L. King, S. K. Amanchi, S. Bheemireddy, and N. H. Younan, "Standards-based middleware and tools for coastal sensor web applications," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 3, no. 4, pp. 451–466, Dec. 2010.

[21] M. A. Veganzones and M. Grana, "A spectral/spatial CBIR system for hyperspectral images," *IEEE J Sel Topics Appl. Earth Observ. Remote Sens.*, vol. 5, no. 2, pp. 488–500, Apr. 2012.

[22] M. Grana and M. A. Veganzones, "An endmember-based distance for content based hyperspectral image retrieval," *Pattern Recognit.*, vol. 45, no. 9, pp. 3472–3489, 2012. [Best Papers of Iberian Conference on Pattern Recognition and Image Analysis (IbPRIA 2011).]

[23] M. A. Veganzones, J. O. Maldonado, and M. Grana, "On content-based image retrieval systems for hyperspectral remote sensing images," in *Computational Intelligence for Remote Sensing, ser. Studies in Computational Intelligence*, vol. 133. Grana M. and Duro R. J., Eds. Berlin, Germany: Springer-Verlag, 2008, pp. 125–144.

[24] 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.

[25] J. W. Boardman, F. A. Kruse, and R. O. Green, "Mapping target signatures via partial unmixing of AVIRIS data," in *Proc. JPL Airborne Earth Sci. Workshop*, 1995, pp. 23–26.

[26] C.-I. Chang and A. Plaza, "A fast iterative algorithm for implementation of pixel purity index," *IEEE Geosci. Remote Sens. Lett.*, vol. 3, no. 1, pp. 63–67, Jan. 2006.

[27] M. E. Winter, "N-FINDR: An algorithm for fast autonomous spectral endmember determination in hyperspectral data," in *Proc. SPIE*, 1999, vol. 3753, pp. 266–277.

[28] M. A. Veganzones, "Contributions to hyperspectral image processing from lattice computing and computational intelligence," Ph.D. dissertation, Univ. Basque Country, Spain, 2012, pp. 1–202.

[29] D. Heinz and C.-I. Chang, "Fully constrained least squares linear mixture analysis for material quantification in hyperspectral imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 39, no. 3, pp. 529–545, Mar. 2001.

[30] A. Plaza, J. Plaza, and A. Paz, "Parallel heterogeneous CBIR system for efficient hyperspectral image retrieval using spectral mixture analysis," *Concurr. Comput. Pract. Exper.*, vol. 22, no. 9, pp. 1138–1159, 2010.

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

[32] J. Nickolls and W. J. Dally, "The GPU computing era," *IEEE Micro*, vol. 30, no. 2, pp. 56–69, Mar./Apr. 2010.
[33] A. Plaza, Q. Du, Y.-L. Chang, and R. King, "Foreword to the special issue on high performance computing in earth observation and remote sensing," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 4, no. 3, pp. 503–507, Sep. 2011.
[34] J. Setoain, M. Prieto, C. Tenllado, A. Plaza, and F. Tirado, "Parallel morphological endmember extraction using commodity graphics hardware," *IEEE Geosci. Remote Sens. Lett.*, vol. 4, no. 3, pp. 441–445, Jul. 2007.
[35] J. W. Boardman, F. A. Kruse, and R. O. Green, "Mapping target signatures via partial unmixing of AVIRIS data," in *Proc. JPL Airborne Earth Sci. Workshop*, 1995, pp. 23–26.
[36] A. Plaza, J. Plaza, and H. Vegas, "Improving the performance of hyperspectral image and signal processing algorithms using parallel, distributed and specialized hardware-based systems," *J. Signal Process. Syst.*, vol. 61, pp. 293–315, 2010.
[37] X. Wu, B. Huang, A. Plaza, Y. Li, and C. Wu, "Real-time implementation of the pixel purity index algorithm for endmember identification on GPUs," *IEEE Geosci. Remote Sens. Lett.*, vol. 11, no. 5, pp. 955–959, May 2014.
[38] J. Nascimento and J. Bioucas-Dias, "Vertex component analysis: A fast algorithm to unmix hyperspectral data," *IEEE Trans. Geosci. Remote Sens.*, vol. 43, no. 4, pp. 898–910, Apr. 2005.
[39] A. Barberis, G. Danese, F. Leporati, A. Plaza, and E. Torti, "Real-time implementation of the vertex component analysis algorithm on GPUs," *IEEE Geosci. Remote Sens. Lett.*, vol. 10, no. 2, pp. 251–255, Mar. 2013.
[40] S. Sanchez and A. Plaza, "Fast determination of the number of endmembers for real-time hyperspectral unmixing on GPUs," *J. Real-Time Image Process.*, vol. 9, pp. 1–9, 2012 [Online]. Available: http://dx.doi.org/10.1007/s11554-012-0276-3
[41] S. Sanchez, R. Ramalho, L. Sousa, and A. Plaza, "Real-time implementation of remotely sensed hyperspectral image unmixing on GPUs," *J. Real-Time Image Process.*, vol. 9, pp. 1–15, 2012 [Online]. Available: http://dx.doi.org/10.1007/s11554-012-0269-2
[42] 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.
[43] J. Bioucas-Dias and J. Nascimento, "Hyperspectral subspace identification," *IEEE Trans. Geosci. Remote Sens.*, vol. 46, no. 8, pp. 2435–2445, Aug. 2008.
[44] C.-I. Chang, *Hyperspectral Imaging: Techniques for Spectral Detection and Classification*. New York, NY, USA: Kluwer/Plenum, 2003.
[45] S. Sánchez, A. Paz, G. Martin, and A. Plaza, "Parallel unmixing of remotely sensed hyperspectral images on commodity graphics processing units," *Concurr. Comput. Pract. Exper.*, vol. 23, no. 13, pp. 1538–1557, 2011.
[46] J. C. Harsanyi and C.-I. Chang, "Hyperspectral image classification and dimensionality reduction: An orthogonal subspace projection," *IEEE Trans. Geosci. Remote Sens.*, vol. 32, no. 4, pp. 779–785, Jul. 1994.
[47] S. Bernabe, S. Lopez, A. Plaza, and R. Sarmiento, "GPU implementation of an automatic target detection and classification algorithm for hyperspectral image analysis," *IEEE Geosci. Remote Sens. Lett.*, vol. 10, no. 2, pp. 221–225, Mar. 2013.
[48] S. Bernabe, S. Sanchez, A. Plaza, S. Lopez, J. A. Benediktsson, and R. Sarmiento, "Hyperspectral unmixing on GPUs and multi-core processors: A comparison," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 6, no. 3, pp. 1386–1398, Jun. 2013.
[49] M. Daube-Witherspoon and G. Muehllehner, "An iterative image space reconstruction algorithm suitable for volume etc.," *IEEE Trans Med. Imag.*, vol. 5, no. 2, pp. 61–66, Jun. 1986.
[50] C. Gonzalez, J. Resano, A. Plaza, and D. Mozos, "FPGA implementation of abundance estimation for spectral unmixing of hyperspectral data using the image space reconstruction algorithm," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 5, no. 1, pp. 248–261, Feb. 2012.
[51] A. Remon, S. Sanchez, S. Bernabe, E. S. Quintana-Orti, and A. Plaza, "Performance versus energy consumption of hyperspectral unmixing algorithms on multi-core platforms," *EURASIP J. Adv. Signal Process.*, vol. 68, no. 1, pp. 1–16, 2013.
[52] J. Sevilla and A. Plaza, "A new digital repository for remotely sensed hyperspectral imagery on GPUs," in *Proc. Int. Symp. Symbolic Numer. Algorithms. Sci. Comput. (SYNASC'13), Workshop. High Perform. Comput. Sci. Probl.*, 2013, vol. 1, pp. 121–125.
[53] J. Sevilla and A. Plaza, "Unmixing-based retrieval system for remotely sensed hyperspectral imagery on GPUs," in *Proc. Int. Conf. Comput. Math. Methods Sci. Eng. (CMMSE'13)*, 2013, vol. 1, pp. 615–619.
[54] J. Harsanyi and C. Chang, "Hyperspectral image classification and dimensionality reduction: An orthogonal subspace projection approach," *IEEE Trans. Geosci. Remote Sens.*, vol. 32, no. 4, pp. 779–785, Jul. 1994.
[55] J. Chanussot, M. M. Crawford, and B.-C. Kuo, "Foreword to the special issue on hyperspectral image and signal processing," *IEEE Trans. Geosci. Remote Sens.*, vol. 48, no. 11, pp. 3871–3876, Nov. 2010.
[56] G. Martin and A. Plaza, "Region-based spatial preprocessing for endmember extraction and spectral unmixing," *IEEE Geosci. Remote Sens. Lett.*, vol. 8, no. 4, pp. 745–749, Jul. 2011.

Jorge Sevilla received the computer engineering degree and the M.Sc. degree from the University of Extremadura, Badajoz, Spain, in 2010 and 2012, respectively, where he is currently pursuing the Ph.D. degree.

He has been a Research Intern with the National Institute of Nuclear Physics, Catania, Italy. He has also been a Visiting Researcher with the West University of Timisoara, Timisoara, Romania. His research interests include hyperspectral image analysis, efficient implementations, development of digital repositories, and content-based image retrieval. He has served as a reviewer for several international journals and conferences.

Dr. Sevilla is a member of the Hyperspectral Computing Laboratory (HyperComp).

Antonio Plaza (M'05–SM'07) received the computer engineering degree, the M.Sc., and the Ph.D. degrees in 1997, 1999, and 2002, respectively.

He is an Associate Professor (with accreditation for Full Professor) with the Department of Technology of Computers and Communications, University of Extremadura, Badajoz, Spain, where he is the Head of the Hyperspectral Computing Laboratory (HyperComp). He was the Coordinator of the Hyperspectral Imaging Network, a European project with total funding of 2.8 MEuro. He authored more than 400 publications, including more than 110 JCR journal papers (66 in IEEE journals), 20 book chapters, and over 240 peer-reviewed conference proceeding papers (94 in IEEE conferences). He has guest edited seven special issues on JCR journals (three in IEEE journals). He has been a Chair for the IEEE Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing (2011).

Dr. Plaza is a recipient of the recognition of Best Reviewers of the IEEE Geoscience and Remote Sensing Letters in 2009 and a recipient of the recognition of Best Reviewers of the IEEE Transactions on Geoscience and Remote Sensing in 2010, a journal for which he has served as an Associate Editor in 2007–2012. He is also an Associate Editor for IEEE Access and the IEEE Geoscience and Remote Sensing Magazine and was a member of the Editorial Board of the IEEE Geoscience and Remote Sensing Newsletter in 2011–2012, and a member of the steering committee of the IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing in 2012. He served as the Director of Education Activities for the IEEE Geoscience and Remote Sensing Society (GRSS) in 2011–2012, and is currently serving as President of the Spanish Chapter of IEEE GRSS since November 2012. He is currently serving as the Editor-in-Chief of the IEEE Transactions on Geoscience and Remote Sensing Journal since January 2013.