A New Fast Algorithm for Linearly Unmixing Hyperspectral Images

Raúl Guerra, Lucana Santos, Sebastián López, *Senior Member, IEEE*, and Roberto Sarmiento

***Abstract*—Linear spectral unmixing is nowadays an essential tool to analyze remotely sensed hyperspectral images. Although many different contributions have been uncovered during the last two decades, the majority of them are based on dividing the whole process of linearly unmixing a given hyperspectral image into three sequential steps: 1) estimation of the number of endmembers that are present in the hyperspectral image under consideration; 2) extraction of these endmembers from the hyperspectral data set; and 3) calculation of the abundances associated with the endmembers induced in the previous step per each mixed pixel of the image. Although this *de facto* processing chain has proven to be accurate enough for unmixing most of the images collected by hyperspectral remote sensors, it is also true that it is not exempt of drawbacks, such as the fact that all the possible combinations of algorithms in order to fully unmix a hyperspectral image according to the aforementioned processing chain demand a formidable computational effort, which tends to be higher the better the performance of the designed unmixing chain is. This troublesome issue unfortunately prevents the use of hyperspectral imaging technology in applications under real-time constraints, in which hyperspectral images have to be analyzed in a short period of time. Hence, there is a clear need to face the challenge of fully exploiting the unquestionable benefits of the hyperspectral imaging technology for these applications, but concurrently overcoming the limitations imposed by the computationally complex nature of the processes involved. For this purpose, this paper introduces a novel algorithm named fast algorithm for linearly unmixing hyperspectral images (FUN), which is capable of fully unmixing a hyperspectral image with at least the same accuracy than state-of-the-art approaches while demanding a much lower computational effort, independent of the characteristics of the image under analysis. The FUN algorithm is based on the concept of orthogonal projections and allows performing the estimation of the number of endmembers and their extraction simultaneously, using the modified Gram–Schmidt method. The operations performed by the FUN algorithm are simple and can be easily parallelized. Moreover, this algorithm is able to calculate the abundances using very similar operations, also based on orthogonal projections, which makes it easier to achieve a hardware implementation to perform the entire unmixing process. The benefits of our proposal are demonstrated with a diverse set of artificially generated hyperspectral images and with the well-known AVIRIS Cuprite image, for which the proposed FUN algorithm is able to reduce in a factor of more than 31 times the time required for processing it, while providing a better unmixing performance than traditional methods.**

Manuscript received February 3, 2015; revised May 11, 2015; accepted June 10, 2015. This work was supported by the REBECCA (TEC2014-58036-C4-4-R) Project.

The authors are with the Institute for Applied Microelectronics, University of Las Palmas de Gran Canaria, Las Palmas de Gran Canaria 35017, Spain (e-mail: rguerra@iuma.ulpgc.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/TGRS.2015.2447573

***Index Terms*—Fast algorithm for linearly unmixing hyperspectral images (FUN) algorithm, linear hyperspectral unmixing, orthogonal projections, real-time applications.**

I. Introduction

LINEAR unmixing has rapidly become one of the most popular techniques to determine the content of a remotely sensed hyperspectral image [1]. It is based on the idea that each captured pixel in a hyperspectral image, which is composed of N_b spectral bands, can be represented as a linear combination of a set of p spectrally pure constituent spectra or endmembers e, weighted by an abundance factor a, which establishes the proportion of each endmember in the pixel under inspection.

This linear mixture model assumes that secondary reflections and scattering effects can be neglected from the data collection procedure, and hence, the measured spectra can be expressed as a linear combination of the spectral signatures of materials present in the mixed pixel. In those cases in which the impact of the secondary reflections or the scattering effects is relevant, it is necessary to resort to more complex nonlinear models, which demand *a priori* information about the geometry and physical properties of the observed objects, which results in a further increase of the computational complexity of the unmixing process.

The linear unmixing process is usually divided in three different steps: estimation of the number of endmembers, endmember extraction, and abundance calculation. In addition, there are additional operations that can be performed and might provide enhanced unmixing results, such as noise filtering or dimensional reduction of the data set. Many different algorithms have been proposed to solve the different steps of the unmixing process. For example, the virtual dimensionality algorithm [2] and the hyperspectral subspace identification by minimum error (HySIME) algorithm [3] are popularly employed to estimate the number of endmembers. These algorithms aim at finding the intrinsic dimensionality of the image data set, attending to different criteria such as the correlation between the bands or between the eigenvalues extracted from the data set. For the endmember extraction, the vertex component analysis (VCA) algorithm [4] and the N-finder algorithm [5] are frequently used. The N-finder algorithm selects the endmembers as the pixels of the image that form the simplex with the largest volume. This process requires solving $p \times p$ determinants, resulting in a high computational effort. On the other hand, the VCA algorithm extracts the endmembers as the pixels with the largest orthogonal projection, starting with a plane spanned by p random vectors that are sequentially replaced by the extracted

endmembers. This algorithm performs p iterations in which the orthogonal projection of each pixel of the hyperspectral image to the selected plane has to be computed. Although the VCA algorithm is faster than the N-finder algorithm, it is still computationally complex. Finally, the fully constraint least squares linear spectral unmixing (FCLSU) algorithm [6] is popular for calculating the abundances of each endmember, minimizing the square error in the linear approximation of the hyperspectral image, imposing the nonnegative constraint and the sum-to-one constraint for the abundances.

Each of the aforementioned algorithms is able to provide good results in the specific task of the unmixing process they were designed for. However, they present several disadvantages for real-time applications in which it is necessary to perform the complete unmixing process with all its steps.

First, the fact that it is necessary to utilize three different algorithms, one for each of the tasks, turns the unmixing process into a sequential chain in which the results of each step depend on the results of the previous step. This yields the propagation of errors from one step to the following one, and hence a loss of accuracy in the results.

The computational complexity represents also a disadvantage. Typical unmixing algorithms have a high computational burden and may also require complex matrix calculations such as the inverse calculation or the eigenvalues and eigenvectors estimation. For instance, in VCA and N-finder, it is necessary to compute the inverse of a matrix in every iteration.

Another disadvantage arises when trying to implement the unmixing process in hardware. The fact that it is necessary to use a different algorithm for each step of the process scales up the difficulty of designing a hardware implementation of the entire unmixing chain, making it almost impossible to achieve real-time performance.

In this paper, a new fast algorithm for linearly unmixing hyperspectral images, named FUN, is proposed with the aim of providing a solution to the aforementioned problems, achieving a high accuracy in the complete unmixing process. The FUN algorithm is based on the concept of orthogonal projections and allows performing the estimation of the number of endmembers and their extraction simultaneously, using simple operations, which can be also easily parallelized. Moreover, the FUN algorithm is able to calculate the abundances based on orthogonal projections, too. The operations performed by the FUN algorithm to calculate the abundances are very similar to the operations performed to extract the endmembers, which makes it easier to achieve a hardware implementation of the entire unmixing process.

Orthogonal projections have been widely utilized in the scientific literature for extracting the endmembers in a hyperspectral image. A well-known example is the orthogonal subspace projection (OSP) algorithm [7]. These algorithms obtain accurate results when the first endmember selected is actually an endmember, but the results get poorer if the algorithms fail at selecting the first endmember. This can be solved using an initialization method, which ensures that the first endmember selected is as near to be a pure signature as possible. Different initialization methods have been empirically proven and compared, and it has been observed that the pixel of the hyperspectral image with the largest orthogonal projection with respect to the centroid pixel is always one of the purest pixels of the image and can be selected as the first endmember, ensuring accurate results. Another difference of the FUN algorithm when compared with other algorithms that use orthogonal projections is its capability of estimating the number of endmembers while extracting them, as well as calculating the abundances, only performing a few additional calculations. These facts bring important advantages if we want to obtain a hardware implementation of the entire unmixing process with real-time performance and without complex matrix operations. Furthermore, the proposed algorithm yields accurate results, which are similar or even better than those obtained with the most relevant algorithms of the state of the art, at a much faster computation time. These facts are corroborated in this study by experimental results performed with both real and synthetic hyperspectral images.

This paper is organized as follows. Section II presents the process by which the estimation of the number of endmembers and the endmembers extraction are performed within the FUN algorithm, whereas Section III introduces how the proposed algorithm can be used for abundance estimation. Section IV shows the main computational advantages of the FUN algorithm, and Section V presents the experimental results. Finally, Section VI draws the conclusions.

II. Estimation of the Number of Endmembers and Their Extraction Within the FUN Algorithm

A hyperspectral image can be represented as $M = [r_1, r_2, \ldots, r_{N_p}]$, where N_p is the number of pixels, and r_i is the ith pixel represented as a spectral vector with N_b components, being N_b the number of bands in the image. In the linear mixing model, each pixel in M can be expressed as

$$r_i = \sum_{j=1}^{p} e_j \cdot a_{i,j} + n_i \tag{1}$$

where e_j represents the jth endmember signature, p is the number of endmembers in the image, and $a_{i,j}$ is the abundance of endmember e_j in the pixel r_i. The noise present in the pixel r_i is contained in the vector n_i.

The unmixing of a pixel in a hyperspectral image requires both the estimation of the number of endmembers p and the extraction of the endmembers, which conform a matrix $E = [e_1, e_2, \ldots, e_p]$. These two tasks are usually performed in two separate steps and executed by two different algorithms.

The FUN algorithm, which is presented in this paper, allows the simultaneous estimation of the number of endmembers and the endmembers extraction. Consequently, it outperforms the algorithms of the state of the art, offering savings in terms of execution time in the unmixing process and facilitating its hardware implementation.

The FUN algorithm selects as first endmember the pixel of the hyperspectral image with the largest orthogonal projection to the centroid pixel. Afterward, it sequentially performs the orthogonal projections of the hyperspectral image, in such a way that one endmember is found in each iteration.

Algorithm 1: Gram Schmidt Orthogonalization

Inputs: $E = [e_1, e_2, ..., e_n]$
for $i = 1$ **to** n **do**
  $q_i = e_i$;
  for $j = 1$ **to** $i - 1$ **do**
    $q_i = q_i - (e_i \circ u_j) \cdot u_j$;
  end
  $u_i = q_i / \|q_i\|$;
end
Outputs:
$Q = [q_1, q_2, ..., q_n]$ {Orthogonalized vector};
$U = [u_1, u_2, ..., u_n]$ {Orthonormalized vector};

After each projection, the algorithm evaluates if it is necessary to extract another endmember. For that purpose, the FUN algorithm estimates the percentage of information that cannot be represented with the endmembers already selected using a stop factor s. If the value obtained is smaller that an input parameter α, the algorithm finishes, thus obtaining the total number of endmembers present in the image plus the extracted endmembers. This is explained with more details in Section V-B.

The orthogonal projections of the hyperspectral image can be obtained using different methods. The FUN algorithm employs a modified version of the Gram–Schmidt method, which features low computational complexity and allows the reuse of previously computed information, speeding up the overall process, and without any complex matrix calculation.

A. Background: The Modified Gram–Schmidt Orthogonalization

The modified Gram–Schmidt method calculates the orthogonal projection of a vector e_i to a set of vectors $E = [e_1, e_2, \ldots, e_k]$ with $k < i$ by subtracting the portion of the vector e_i contained in the directions spanned by the vectors $E = [e_1, e_2, \ldots, e_k]$ with $k < i$.

Consequently, the modified Gram–Schmidt method allows orthogonalizing a set of independent vectors $E = [e_1, e_2, \ldots, e_n]$. The result is a set of orthogonal vectors $Q = [q_1, q_2, \ldots, q_n]$ and their normalized vectors $U = [u_1, u_2, \ldots, u_n]$. The Gram–Schmidt process works as shown in Algorithm 1.

In the pseudocode, "$\cdot$" represents the product of a vector and a scalar value, and the "$\circ$" operator is used for the scalar product between two vectors.

B. Initialization of the Algorithm: Exctraction of the First Endmember

The FUN algorithm has to be initialized with an endmember. Therefore, it is necessary to find a pure pixel in the image. In order to do that, different criteria can be applied. In this paper, we have analyzed three different methods to accurately extract a first endmember, since its accuracy greatly affects the performance of the FUN algorithm.

1) Initialization Based on the Maximum Variance: When observing a hyperspectral scene, one can see that, in some bands, the reflectance of the different pixels has a higher variance than in others. Identifying the band in which the pixels present the highest variance is useful to find a pure pixel. Considering the linear mixing model, we can see that the mixed pixels are a weighted average of the reflectance of the different endmembers and their corresponding abundances; thus, the reflectance of a mixed pixel is always a value that is between the maximum and minimum reflectance of the endmembers present in it.

Taking this into account, it can be observed that, in the band in which the differences between the pixels present in the scene are larger, the mixed pixels show intermediate values, whereas the more extreme values (maximum and minimum) correspond to pure pixels, and thus can be considered as endmembers. This implies that, in the band with the highest pixel variance, at least two pure pixels will always be identified, which correspond to those with the highest and lowest values. This is always fulfilled regardless of the number of endmembers and the number of mixed pixels.

Considering the aforementioned facts, the FUN algorithm can be initialized by finding first the band where the variance among the pixels is the highest and then picking the brightest pixel as the first endmember. We note that the pixel with the lowest value could be also selected, but it is not advisable, since it might be comparatively more affected by the noise.

2) Initialization Based on the Centroid Pixel: As with many geometrical unmixing algorithms, the FUN algorithm tries to select as endmembers the most different pixels or the most extreme ones within the captured data set. In order to select the first endmember according to this criterion, the average pixel, also named centroid, can be used, as it is done in the iterative error analysis algorithm [8].

Considering how the FUN algorithm works, it is straightforward to use the centroid pixel to initialize the unmixing process. First, the average pixel is computed. Once this pixel is obtained, the orthogonal projection of all the pixels of the hyperspectral image with respect to this pixel is computed using the modified Gram–Schmidt method. This orthogonal projection shows how much information of each pixel is not contained in the centroid pixel, showing how different each pixel is from the average. Finally, the FUN algorithm selects the pixel with the largest orthogonal projection as the first endmember.

3) Initialization Based on the Brightest Pixel: Some geometrical unmixing algorithms select the first endmember relying only on the brightness of the sensed pixels, selecting the brightest one as endmember. This is the case of the OSP algorithm [7].

C. Simultaneous Unmixing and Estimation of the Number of Endmembers

Once the first endmember has been selected, the FUN algorithm sequentially extracts new endmembers, by selecting the pixels with the largest orthogonal projections to the endmembers already extracted. Initially, the selected pixels are considered only candidates to be endmembers. Before these pixels are added to the list of extracted endmembers, the FUN algorithm estimates if it has really found a new endmember, or instead, if all the endmembers had been already found in the previous iteration, and hence, the extracted pixel is not really an endmember. This way, the FUN algorithm manages to extract

Algorithm 2: Process to extract the endmembers

Inputs: $M = [r_1, r_2, ..., r_{N_p}]$, $\alpha = 1$
1 $E := []$; {Endmembers matrix }
2 $Q := []$; {Endmembers' Gram Schmidt orthogonalization}
3 $U := []$; {Endmembers' Gram Schmidt orthonormalization}
4 $X := [x_1, x_2, ..., x_{N_p}] = M$ {Auxiliary copy of the hyperspectral cube}
5 $e_1 := x_j$ where x_j is the pixel of the hyperspectral image selected as first endmember according to the initialization criteria; {Select the first endmember}
6 $q_1 := e_1$;
7 $u_1 = q_1/\|q_1\|$;
8 $E := [e_1]$;
9 $Q := [q_1]$;
10 $U := [u_1]$;
11 $p := 1$; {Number of endmembers found}
12 exit := 0;
13 **while** *exit* $= 0$ **do**
14 **for** $j = 1$ **to** N_p **do**
15 $x_j = x_j - (x_j \circ u_p) \cdot u_p$;
16 $s_j = 100 \cdot \|x_j\|/\|r_j\|$;
17 **end**
18 **if** $\max(s) \leq \alpha$ **then**
19 exit $= 1$;
20 **else**
21 $j_{\max} := \arg\max(s_j)$;
22 $p := p + 1$;
23 $q_p := x_{j_{\max}}$;
24 $u_p := x_{j_{\max}}/\|x_{j_{\max}}\|$;
25 $e_p := r_{j_{\max}}$;
26 **end**
27 **end**
Outputs: p, {number of endmembers}
$E = [e_1, e_2, ..., e_p]$, {Endmembers}
$Q = [q_1, q_2, ..., q_p]$, {Orthogonalized endmembers}
$U = [u_1, u_2, ..., u_p]$, {Orthonormalized endmembers}

the endmembers and to estimate the number of endmembers in a single process.

In order to decide if a pixel is really an endmember, when a new endmember candidate is found, the FUN algorithm calculates the percentage of information that would be lost if that endmember was not added, by calculating a stop factor and comparing it with a parameter named α, which is given as an input to the algorithm and is defined by the user.

If the percentage of information lost is lower than α, the pixel is not considered an endmember, and the algorithm stops, since all the endmembers present in the image have been already found and extracted.

The pseudocode that describes the FUN algorithm is presented in Algorithm 2, where we consider that the noise in the input hyperspectral image (M) has been previously filtered. First of all, in line 5, the pixel with the largest orthogonal projection with respect to the centroid pixel is selected as the first endmember. In line 15, the orthogonal projection of each pixel to the endmembers already selected is calculated and saved in the matrix X. The stop factor s is calculated in line 16 for each pixel in the matrix X, obtaining a vector with the percentage of information of each pixel that cannot be represented with the endmembers already extracted. In line 18, the maximum value of the vector s, i.e., $s_{j_{\max}}$, is compared with the input parameter α. The algorithm finishes if $s_{j_{\max}} \leq \alpha$; otherwise, the pixel $r_{j_{\max}}$ is considered an endmember, and the algorithm continues.

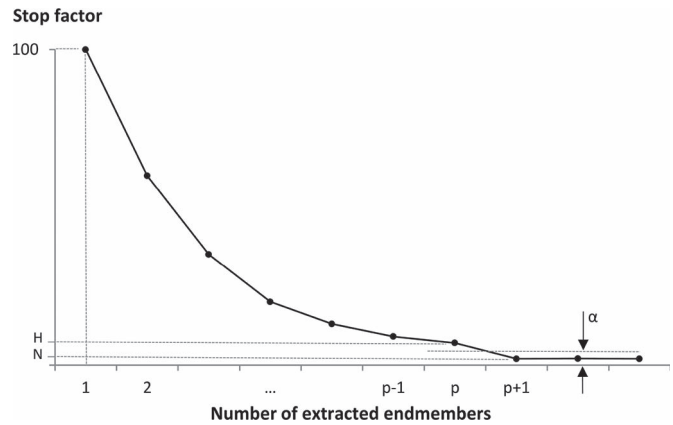

Fig. 1. Theoretical representation of the values that would be obtained for the stop factor as the number of endmembers extracted by the FUN algorithm increases, from 1 to $p + 1$, being p the real number of endmembers present in the hyperspectral image.

D. Stopping Condition

According to the linear mixing model, all the pixels in a hyperspectral image are linear combinations of the pure spectral signatures (endmembers) present in it.

Considering p as the number of endmembers, a hyperspectral image corresponds with a subspace of p dimensions, conformed by p independent vectors, which are the endmembers. Hence, the maximum possible number of orthogonal vectors present is likewise p, and as a consequence, the orthogonal projection of any other possible endmember candidate beyond the p present in the image will be null.

The FUN algorithm takes advantage of this fact to determine when it has finished extracting the p endmembers.

However, with the presence of noise, we observed that the orthogonal projection of new endmember candidates after p have been already extracted is not null but, instead, very small, and it decreases as the signal-to-noise ratio (SNR) increases.

In order to create a stopping condition attending to the aforementioned facts, a stop factor s is created and defined as the percentage of a pixel that cannot be represented as a linear combination of the endmembers already extracted according to the following equation:

$$s = 100 \cdot \frac{\|x_i\|}{\|r_i\|} \tag{2}$$

where $\|r_i\|$ represents the norm of the ith pixel, and $\|x_i\|$ is the norm of the Gram–Schmidt orthogonalization of r_i.

Fig. 1 depicts how the value of the stop factor changes as the endmembers are extracted.

As it can be observed, the stop factor for the first endmember is always 100, and it decreases smoothly as the number of endmember extracted increases, until the p endmembers present in the image are extracted. When trying to extract the $(p + 1)$th endmember, the stop factor gets significantly smaller. If we consider H the value of the stop factor of the pth endmember and N the value of the stop factor of the $(p + 1)$th endmember candidate, we observe that the ratio H/N increases with the SNR. We will be able to determine if the FUN algorithm has

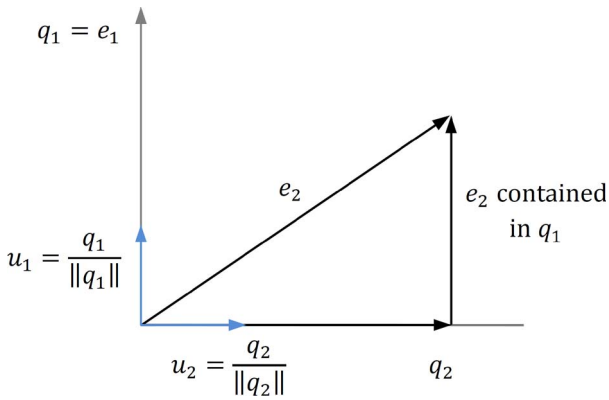

Fig. 2. Example of Gram–Schmidt orthogonalization of the endmember e_2 with respect to the endmember e_1; q_2 represents the amount of information in e_2 that is not contained in e_1; u_1 and u_2 are unitary vectors with the directions of q_1 and q_2, respectively.

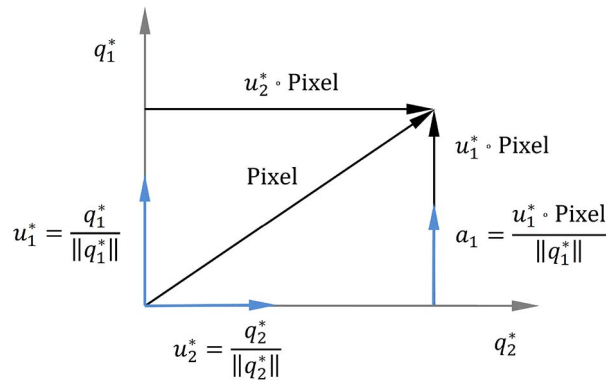

Fig. 3. Example of projection of a pixel in the direction of two endmembers. Vectors q^* and u^* refer to the vectors obtained with the FUN algorithm by orthogonalizing the endmembers using the modified Gram–Schmidt method. Using these vectors, the abundance of each endmember in the pixel can be obtained as shown in a_1, which represents the abundance of the endmember e_1 in this pixel.

finished extracting the endmembers by setting a value between H and N, which is denoted by α. If the stop value is smaller than α, the algorithm will stop.

According to this, α represents the percentage of information of the next pixel susceptible of being selected as endmember that will be considered as noise.

III. Abundance Estimation Within the FUN Algorithm

Considering the linear unmixing model shown in (1), once the p endmembers $E = [e_1, e_2, \ldots, e_p]$ have been found, it is necessary to estimate the abundances $(a_{i,j})$ of each endmember e_j in a pixel r_i, in order to complete the unmixing process.

To estimate the abundances, two physical constraints can be imposed to this linear model, namely, the abundance nonnegativity constraint, which forces the condition that $a_i > 0$ for all $1 < i < p$, and the abundance sum-to-one constraint, which establishes that $\sum_{i=1}^{p} a_i = 1$.

Here, we explain how the FUN algorithm estimates the abundances. The main advantage of the FUN algorithm when used to calculate the abundances is that it works only with the extracted endmembers $E = [e_1, e_2, \ldots, e_p]$, which are utilized to obtain a matrix Q^{**} that allows directly estimating the abundances of each pixel r_i by the operation $a_{i,:} = (Q^{**})^T \cdot r_i$. Therefore, the abundances matrix can be estimated as $A = (Q^{**})^T \cdot M$.

The $*$ and $**$ symbols are used to show that matrices U^*, Q^*, and Q^{**} are obtained as variations of the matrices U and Q, as shown in the following, but still preserve their properties of being formed by orthonormal and orthogonal vectors, respectively.

In order to obtain the matrix Q^{**}, which allows easily estimating the abundances, the Gram–Schmidt orthogonalization has to be applied p times. This way, the FUN algorithm is able to extract the exact information that makes each endmember different. In particular, when the Gram–Schmidt method is applied in order to orthogonalize a set of endmembers $E = [e_1, e_2, \ldots, e_p]$, the orthogonalization is performed by subtracting to each endmember the information that is contained in the endmembers already orthogonalized. Fig. 2 shows an example in which two vectors e_1 and e_2 are orthogonalized. It can be seen that the vectors resulting from the Gram–Schmidt process, i.e., q_2 and u_2, contain the information present in the endmember e_2, which is not part of endmember e_1. However, vectors q_1 and u_1 contain information that is part of e_2.

When the Gram–Schmidt method is applied to a set of p endmembers, the resulting vectors q_p and u_p will contain the information of the endmember e_p that is not contained in the rest of the endmembers $E = [e_1, e_2, \ldots, e_{p-1}]$. This information is what makes the endmember e_p different from the rest. However, the rest of the vectors in $Q = [q_1, q_2, \ldots, q_{p-1}]$ and $U = [u_1, u_2, \ldots, u_{p-1}]$ will contain information of more than one endmember. If the Gram–Schmidt method is applied p times to the set of endmembers, and the last endmember (e_p) is a different one each time, the corresponding q_p and u_p can be saved to the matrices $Q^* = [q_p^{*(1)}, q_p^{*(2)}, \ldots, q_p^{*(p)}]$ and $U^* = [u_p^{*(1)}, u_p^{*(2)}, \ldots, u_p^{*(p)}]$, in such a way that these matrices will contain the information that is exclusive to a specific endmember, i.e., the information in each endmember e_i that is not present in the rest of the endmembers.

The resulting orthogonalized and orthonormalized set of vectors, i.e., Q^* and U^*, both comprising the information exclusive to each endmember, can be used to project the pixels of the image in their directions, as shown in Fig. 3.

As shown in the example, by projecting the pixels of the hyperspectral image in the set of vectors Q^* and U^*, it is possible to obtain the portion of each pixel that can be represented in each of the directions $U^* = [u_p^{*(1)}, u_p^{*(2)}, \ldots, u_p^{*(p)}]$. This portion can be formally described as

$$\text{Portion}_i = u_p^{*(i)} \circ r_i \tag{3}$$

where r_i is a pixel, and $\circ$ is the scalar product; hence, Portion_i is a scalar value. Finally, the reconstructed pixels r_i can be obtained as the sum of these portions multiplied by their corresponding directions as

$$r_i = \sum_{i=1}^{p} \text{Portion}_i \cdot u_p^{*(i)}. \tag{4}$$

Algorithm 3: Process to obtain the abundances

Inputs: $M = [r_1, r_2, ..., r_{N_p}]$, p, $E = [e_1, e_2, ..., e_p]$
1 $E = [E, E]$;
2 $\text{Norms} = [], U = [], Q = [], U^* = [], Q^{**} = []$;
3 **for** $k = 2$ **to** $p + 1$ **do**
4 $\quad U_1 = E_k / \|E_k\|$;
5 $\quad$ **for** $j = 2$ **to** p **do**
6 $\quad\quad x = E_{k+j-1}$;
7 $\quad\quad$ **for** $i = 1$ **to** $j - 1$ **do**
8 $\quad\quad\quad q_j = x - (x \circ q_j) \cdot q_j$;
9 $\quad\quad$ **end**
10 $\quad\quad U_j = q_j / \|q_j\|$;
11 $\quad$ **end**
12 $\quad \text{Norms} = [\text{Norms}, \|q_p\|]$;
13 $\quad U^* = [U^*, u_p]$;;
14 **end**
15 **for** $i = 1$ **to** p **do**
16 $\quad Q_i^{**} = U_i^* / \text{Norms}_i$;
17 **end**
18 $A = (Q_i^{**})^T \cdot \text{IMG}$; {Abundances}
Outputs: Abundances

Taking into account that $u_p^{*(i)} = (q_p^{*(i)} / \|q_p^{*(i)}\|)$, the expression can be rewritten as

$$r_i = \sum_{i=1}^{p} \text{Portion}_i \cdot \frac{q_p^{*(i)}}{\left\| q_p^{*(i)} \right\|}. \tag{5}$$

We can assume that $q_p^{*(i)} \sim e_i$, since $q_p^{*(i)}$ comprises the information that makes e_i different from the rest of the endmembers. Hence, the expression in (5) can be rewritten as

$$r_i = \sum_{i=1}^{p} \text{Portion}_i \cdot \frac{q_p^{*(i)}}{\left\| q_p^{*(i)} \right\|} \sim \sum_{i=1}^{p} \text{Portion}_i \cdot \frac{e_i}{\left\| q_p^{*(i)} \right\|}. \tag{6}$$

Finally, the abundances of a pixel can be calculated as

$$A_i = \frac{\text{Portion}_i}{\left\| q_p^{*(i)} \right\|} = \frac{u_p^{*(i)} \circ \text{Pixel}_i}{\left\| q_p^{*(i)} \right\|}. \tag{7}$$

The vectors of the matrix U^* $[u_p^{*(1)}, u_p^{*(2)}, \ldots, u_p^{*(p)}]$ can be divided once by the norms of the corresponding vector $Q^* = [q_p^{*(1)}, q_p^{*(2)}, \ldots, q_p^{*(p)}]$, obtaining the matrix $Q^{**} = [q_p^{**(1)}, q_p^{**(2)}, \ldots, q_p^{**(p)}]$. The resulting vectors in Q^{**} are then multiplied by each pixel of the hyperspectral image, thus obtaining their abundances, instead of dividing each of the pixels of the matrix by the norms of the corresponding vectors $Q^* = [q_p^{*(1)}, q_p^{*(2)}, \ldots, q_p^{*(p)}]$. This way, the abundances of the hyperspectral image can be directly obtained as

$$A = (Q^{**})^T \cdot M. \tag{8}$$

The pseudocode in Algorithm 3 shows the process that has been described to calculate the abundances from the obtained endmembers by using the Gram–Schmidt orthogonalization. In lines 3–11, the Gram–Schmidt orthogonalization is performed p times, each finishing with a different endmember e_p.

As it can be observed, the FUN algorithm does not impose the nonnegative constraint and the sum-to-one constraint, and therefore, it can happen that the resulting abundances do not meet the aforementioned constraints. However, we have experimentally observed that the abundances estimated by the FUN algorithm are in fact very close to meet the constraints. Although some negative values appear, they are really close to zero, and the sum of the abundances is also very close to one. Taking this into account, it is possible to apply the constraints to the abundances estimated by the FUN algorithm, by setting all negative abundances to zero and then dividing all the abundance vectors by the sum of their terms. All these facts are better illustrated and demonstrated in the experimental results in Section V.

IV. Computational Complexity of the FUN Algorithm

Generally, in order to complete the entire unmixing process, different algorithms have to be used to estimate the number of endmembers, extract them, and calculate the abundances. The main advantage of the FUN algorithm is that a single core algorithm (Gram–Schmidt) can be employed to simultaneously obtain the number of endmembers and extract them from the image under analysis. Additionally, the same algorithm can be utilized to calculate the abundances. In addition to the obvious gains that this fact brings in terms of faster computational times, the FUN algorithm has the advantage that it uses simple mathematical operations that can be easily parallelized, avoiding complex matrix operations such as calculating the inverse or computing the eigenvalues and eigenvectors, operations that are present in most of the unmixing algorithms. This represents an important advantage, because it eases the hardware implementation of the algorithm [9]–[12].

The parallelization suitability of the FUN algorithm when compared with other algorithms can be seen in the fact that the FUN algorithm is capable of estimating the number of endmembers and extracting them in a single process, performing exactly the same operations on each pixel of the image in each iteration, without needing any information from neighboring pixels, thus avoiding creating data dependence.

Another advantage of the FUN algorithm appears when we want to employ it to calculate the abundances, because it only needs the endmember matrix as an input, which is then orthogonalized. After the orthogonalization, the obtained matrix is multiplied by the hyperspectral image pixels obtaining the abundances of each pixel directly. This way, the FUN algorithm uses considerably less data during the process than other algorithms, which need not only the endmember matrix but also the complete hyperspectral image.

V. Experimental Results

This section discloses the most significant results obtained by the FUN algorithm when it is utilized to perform the linear unmixing process. A set of artificially generated hyperspectral data and the real Cuprite hyperspectral image collected by the Airborne Visible/Infrared Imaging Spectrometer (AVIRIS) have been used as inputs to the algorithm. The performance of the FUN algorithm is evaluated by comparing the results with those obtained, for the same input hyperspectral images, with some of the most relevant algorithms of the state of the art, namely, the HySIME algorithm for the estimation of the number of endmembers, VCA for the endmember extraction, and

FCLSU for the calculation of the abundances. These algorithms have proven to yield very accurate results for the corresponding three stages of the unmixing process, as it can be concluded by the fact that they are considered a reference in many scientific studies in the field.

A. Synthetic Data Set Used to Perform the Simulations

Artificial hyperspectral images represent an excellent test bench in order to study the goodness of our proposal, because both their signature endmembers and their abundances are known in advance. In particular, the hyperspectral images used in this work were generated using three different MATLAB functions, and consequently, three different types of synthetic hyperspectral images were obtained. These three MATLAB functions allow creating hyperspectral images of a spatial size defined by the user, from p spectral signatures randomly selected from the U.S. Geological Survey (USGS) digital spectral library [13] that are then mixed according to abundance fractions, which are generated using different probabilistic distributions. In particular, Type-1 hyperspectral images are generated using the Gauss spheric distribution; Type-2 images are generated with the Legendre polynomial distribution, and finally, Type-3 images are generated with a properly tuned Dirichlet distribution.

In addition, a certain amount of Gaussian noise is added so that the generated images have a user-defined SNR.

B. Effect of Noise Filtering in the Performance of the FUN Algorithm

In the linear mixing model, each pixel of the image can be represented as the linear combination of the endmembers present in it, weighted by the corresponding abundances, plus a certain amount of noise. The FUN algorithm performs the unmixing process according to this linear mixing model assuming that the amount of noise in the image is neglectable, and hence, the endmembers can be directly extracted from the pixels in the hyperspectral image. The presence of a high level of noise in the scene would make the FUN algorithm obtain less accurate results, and as it was explained in Section II-D, filtering the noise has also an important impact in the stopping criterion.

In order to filter the noise, we utilize in this study the first stage of the HySIME algorithm, in which the noise is estimated and subtracted from the hyperspectral image. However, other noise filtering options are possible.

The benefits of the noise filtering and its influence in the stopping condition of the FUN algorithm are evaluated by performing simulations in which the endmembers are extracted with the FUN algorithm from hyperspectral images filtering and without previously filtering the noise. In these simulations, the number of endmembers to be extracted has been fixed to a number higher than the actual number of endmembers, and the stop factor, defined in Section II-D, is saved for all iterations of the FUN algorithm (see line 18 of Algorithm 2).

Figs. 4 and 5 show the different values taken by the stopping condition depending on the number of endmember extracted and the SNR of the image when the noise is filtered and when it is not, respectively.

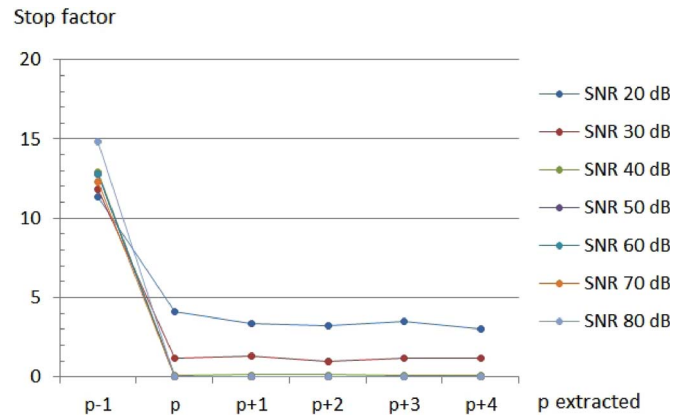

Fig. 4. Mean values of the stop factor depending on the number of endmembers extracted and the SNR of the hyperspectral image, filtering the noise of the image. Images with five endmembers.

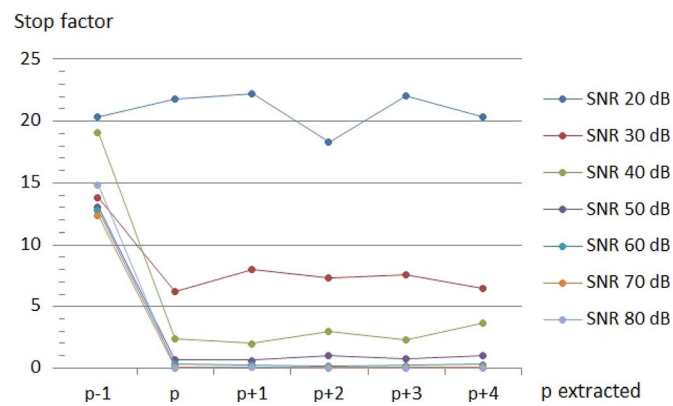

Fig. 5. Mean values of the stop factor depending on the number of endmembers extracted and the SNR of the hyperspectral image, without filtering the noise of the image. Images with five endmembers.

As it can be seen in the results, when the noise of the hyperspectral image is previously filtered, the value taken by the stop factor tends to be null if we try to extract more endmembers than the real number of endmembers present in the image, i.e., p. However, when we are extracting a number of endmembers smaller than p, the stop factor is significantly higher. This makes it possible to define parameter α, which can be used to determine when to stop the FUN algorithm, as it was described in Section II-D.

We observe a similar trend for the stop factor in those cases in which the noise is not previously filtered (see Fig. 5). However, in these cases, the values taken by the stop factor once all the endmembers have been extracted are not so close to zero, which makes it more difficult to determine the right value of α.

Considering the exposed facts, in the subsequent simulations exposed here, α has been always set to 1, and the noise in the hyperspectral image was always filtered before applying any algorithm for endmember extraction.

C. Accuracy of the First Endmember Extracted With the Different Initialization Methods

In order to evaluate the quality of the different initialization methods proposed in Section II-B, some simulations have been performed using synthetic hyperspectral images.

TABLE I
MEAN SPECTRAL ANGLE OBTAINED FOR THE FIRST ENDMEMBER EXTRACTED USING EACH OF THE THREE INITIALIZATION METHODS PROPOSED

Mean spectral angle (°)		
I1	I2	I3
0.815	0.759	0.851

These images have been created with 5, 10, and 15 endmembers with 224 spectral bands each, all taken from the USGS library, and different SNR values, which have been set to 20, 30, 40, 50, and 60 dB and a dimension of 90 000 pixels. Moreover, the type of image has been also varied, from Type 1 to Type 3, according to the different probabilistic distributions utilized to create the abundance maps, as it was explained in Section V-A. All this makes a total of 45 groups of ten different hyperspectral images each.

Ten simulations were performed for each group of synthetic hyperspectral images, selecting one pixel as first endmember with each of the three methods proposed in Section II-B. The accuracy of the pixel selected as endmember is evaluated by calculating the spectral angle, as described in

$$\mathrm{SA} = \arccos \frac{\mathrm{EE} \circ \mathrm{RE}}{\|\mathrm{EE}\| \cdot \|\mathrm{RE}\|} \tag{9}$$

where EE refers to the extracted endmembers, RE refers to the real endmembers present in the image, "$\circ$" represents the scalar product between the two endmembers vectors, and "$\cdot$" represents the product between two scalar values. The norm of the endmember vectors is represented as "$\|\|$."

The mean results obtained for the spectral angle for each initialization method are shown in Table I. Column headed as I1 refers to the spectral angle obtained with the first endmember extracted by selecting the pixel with the highest value in the band with the maximum variance, as described in Section II-B1. Column headed as I2 refers to the spectral angle obtained with the first endmember extracted using the centroid pixel, as shown in Section II-B2. Finally, column headed as I3 refers to the spectral angle obtained with the first endmember extracted as the brightest pixel of the hyperspectral image, as justified in Section II-B3.

According to the results obtained, it is concluded that the most accurate endmember is obtained computing the centroid, and hence, this method has been the one used in the rest of this paper.

D. Performance of the FUN Algorithm for Estimation of the Number of Endmembers

In order to evaluate the performance of the FUN algorithm when estimating the number of endmembers in the image, the results obtained by the FUN algorithm are compared with those of the HySIME algorithm targeting the same hyperspectral images. This bench of synthetic images is very similar to the one described in Section V-C.

Attending to the different configurable parameters, ten synthetics images were created for each *Type*, *SNR*, and p combination. The mean results obtained in the simulations performed, for each group of hyperspectral images, are computed and graphically displayed in Fig. 6, where the axis labeled as *Correct estimations* refers to the percentage of cases in which the number of endmembers estimated is exactly the number of endmembers present in the image (represented with circles), and the axis named *Mean error* shows the mean values of the differences between the number of endmembers obtained by the evaluated algorithms and the real number of endmembers present in the images (represented with bars).

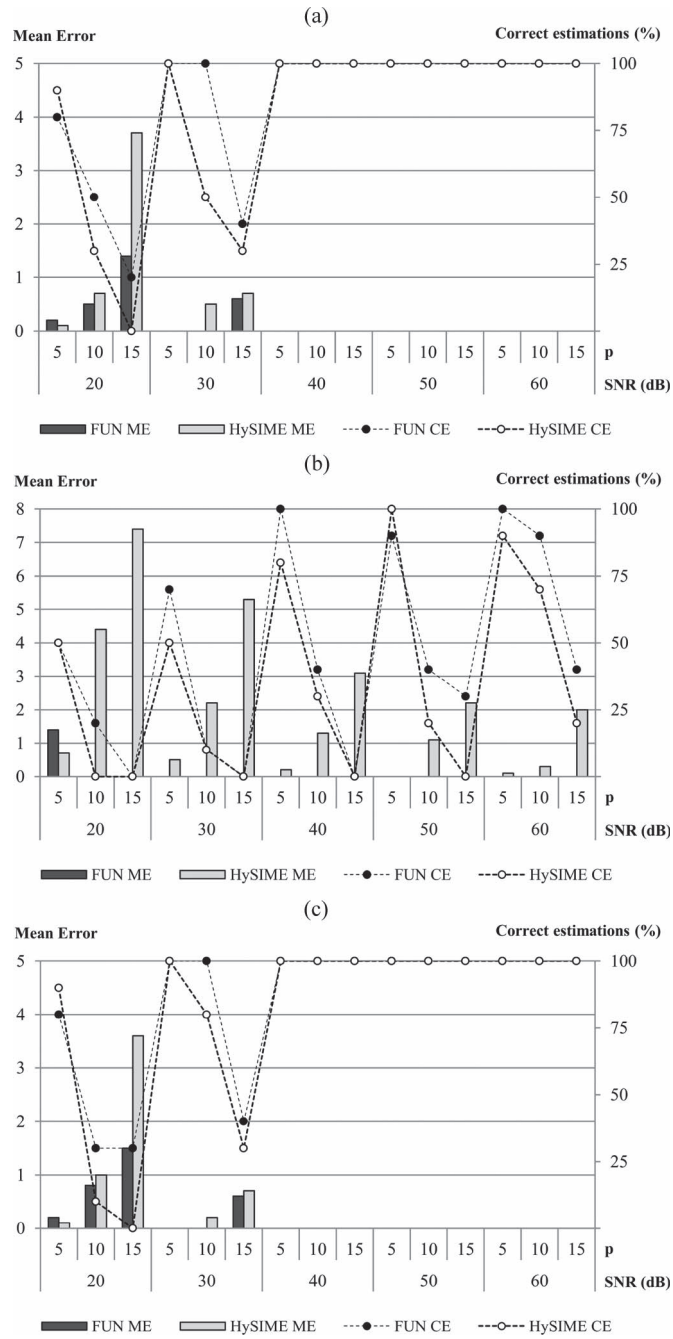

Fig. 6. Number of endmembers estimation. Percentage of correct estimations and mean error. (a) Image type 1. (b) Image type 2. (c) Image type 3.

It can be observed that the percentage of correct estimations is similar for both algorithms, but the mean errors obtained using the FUN algorithm are, in the vast majority of the cases, considerably smaller than the mean errors obtained with the HySIME algorithm. According to these facts, it can be

TABLE II
MEAN SPECTRAL ANGLE OBTAINED WITH THE FUN AND VCA ALGORITHMS

Image's characteristics			Mean spectral angle (°)	
SNR (dB)	p	Type	FUN	VCA
20	5	1	1.339	1.070
		2	3.258	2.329
		3	2.001	1.592
	10	1	2.167	1.954
		2	5.307	4.820
		3	1.997	1.927
	15	1	3.473	2.903
		2	7.748	7.083
		3	2.583	3.197
40	5	1	0.113	0.091
		2	1.702	1.728
		3	0.119	0.288
	10	1	0.273	0.208
		2	1.435	1.897
		3	0.331	0.314
	15	1	0.410	2.012
		2	3.286	7.090
		3	0.292	1.221
60	5	1	0.011	0.010
		2	0.701	0.697
		3	0.011	0.010
	10	1	0.023	0.043
		2	0.929	1.625
		3	0.030	0.019
	15	1	0.047	0.021
		2	1.059	9.325
		3	0.044	0.048

concluded that the FUN algorithm provides better results than HySIME when estimating the number of endmembers of a hyperspectral image.

E. Performance of the FUN Algorithm for the Endmembers Extraction Knowing the Number of Endmembers in Advance

Using a similar bench of synthetic images to the one described in Section V-C, we performed additional simulations in which we evaluated the quality of the endmembers extracted by the FUN algorithm. For this purpose, we explicitly informed the FUN algorithm of the number of endmembers present in the image, we extracted them, and then, we compared them with those obtained when the endmembers are extracted with VCA, by calculating the spectral angle, as described in (9).

Attending to the different configurable parameters, ten synthetics images were created for each *Type* and p combination. The mean results obtained in the simulations performed using these images are displayed in Table II.

Fig. 7 graphically shows the spectral angle obtained between the endmembers extracted by the FUN and VCA algorithms and the real endmembers present in the hyperspectral image. It can be seen that, in the images with a high amount of noise $(\mathrm{SNR} = 20)$, the results obtained using the FUN algorithm are very close to the ones obtained using the VCA algorithm. However, in the images with less noise, the results obtained by the FUN algorithm are much better than the results obtained by the VCA algorithm, improving as the number of endmembers increases. We can conclude that the FUN algorithm is as accurate as the VCA algorithm in terms of quality of the extracted endmembers, showing similar spectral angle values, which tend to be significantly better for the FUN algorithm as the number of endmembers increases.

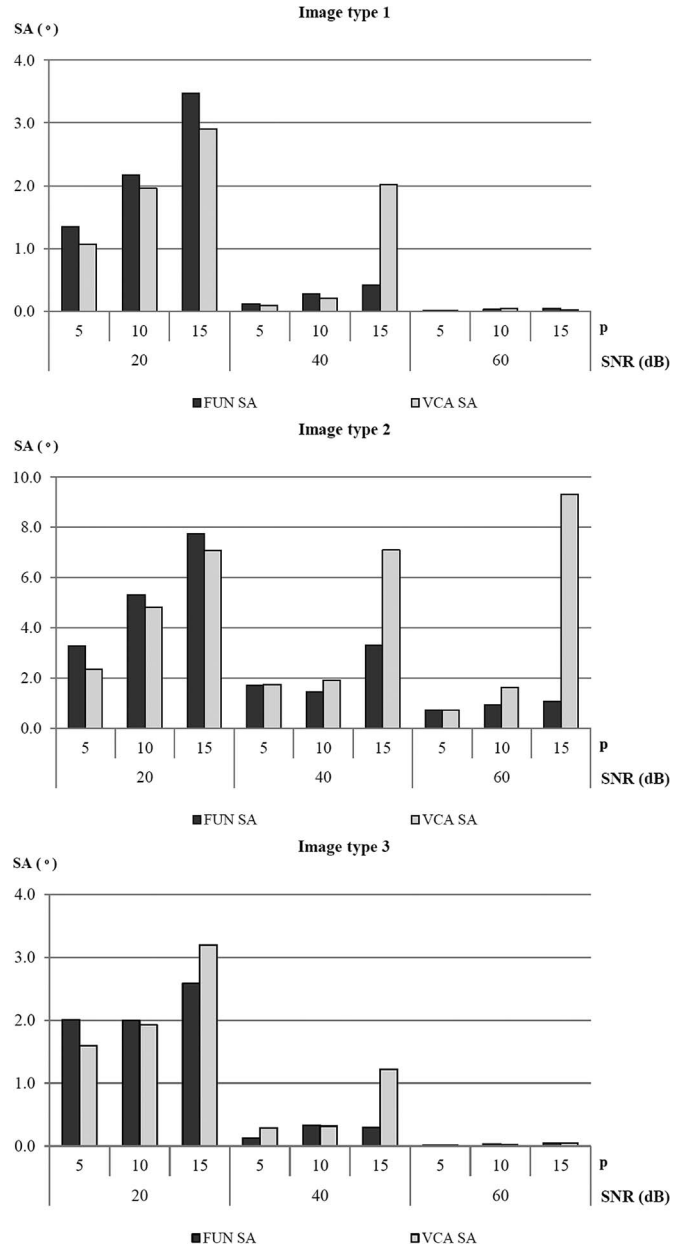

Fig. 7. Mean spectral angle obtained with the FUN and VCA algorithms using a set of different synthetic hyperspectral images, varying the signal-to-noise ratio (SNR), the number of endmembers in the scene (p), and the abundances distribution type.

F. Performance of the FUN Algorithm for Hyperspectral Unmixing

Finally, we evaluate the performance of the FUN algorithm for a typical use, i.e., when it is utilized for estimating the number of endmembers and extracting them simultaneously. The automatic stopping condition parameter α for the FUN algorithm has been set to $\alpha = 1$. The HySIME algorithm is utilized as a reference to evaluate the estimated number of

TABLE III
AVERAGE RESULTS OBTAINED FOR HYPERSPECTRAL UNMIXING

Image's characteristics		Estimation of the number of endmembers				Mean spectral angle (°)		
		Correct estimations (%)		Mean error				
SNR (dB)	p	FUN	HS	FUN	HS	FUN	VCA	NMC (%)
20	5	80	60	0.2	0.6	1.232	1.150	50
	10	40	0	0.9	1.8	1.656	3.985	30
	15	10	0	2.5	5.6	-	-	0
30	5	100	100	0	0	0.379	0.598	100
	10	80	40	0.3	0.9	0.575	0.834	40
	15	40	40	1.8	2.5	0.851	3.887	60
40	5	90	100	0.1	0	0.150	0.120	90
	10	70	80	0.3	0.2	0.470	0.865	90
	15	90	90	0.3	0.4	0.365	2.343	90
50	5	100	100	0	0	0.076	0.044	100
	10	90	90	0.1	0.2	0.162	0.127	90
	15	70	60	0.7	0.8	0.225	4.896	80
60	5	100	100	0	0	0.026	0.021	100
	10	70	80	0.5	0.2	0.029	0.074	70
	15	70	70	0.8	0.9	0.063	6.679	90

endmembers, and the VCA algorithm is used as reference to evaluate the quality of the extracted endmembers.

A new image bench, with the same configuration of the data sets used in the previous experiments, has been generated and used as input of the algorithms. The generated images have been grouped in sets of ten images, according to their characteristics, and the mean results obtained for the images in each group are shown in Table III. FUN columns contain the results obtained with the FUN algorithm, whereas the columns named HS contain the results obtained with the HySIME algorithm, and columns headed as VCA contain the results obtained with the VCA algorithm. The spectral angle values were only calculated when the numbers of endmembers estimated by the HySIME algorithm and by the FUN algorithm were the same. The number of simulations in which the HySIME and FUN algorithms estimated the same number of endmember is shown in the column named *Number of Matching Cases, NMC* (%).

Fig. 8 graphically shows the percentage of correct number of endmembers estimation and the mean error in these estimations, respectively, obtained by the FUN and HySIME algorithms. As it can be seen, the results obtained by the FUN algorithm in terms of the mean error obtained when estimating the number of endmembers are generally better than the results obtained by the HySIME algorithm. Fig. 8 also shows the mean spectral angle obtained by the FUN and VCA algorithms. As it can be seen, the results obtained by the FUN algorithm are very similar, or much better in some cases, than the results obtained by the VCA algorithm.

These results demonstrate the good performance of the FUN algorithm when compared with the HySIME and VCA algorithms, with the unquestionable gain that the FUN algorithm is able to perform two tasks in the same process, whereas to achieve the same with HySIME and VCA, it is necessary to execute the two algorithms sequentially. This represents an advantage for the FUN algorithm, which is able to speed up the hyperspectral unmixing process and will yield also hardware implementations of reduced complexity.

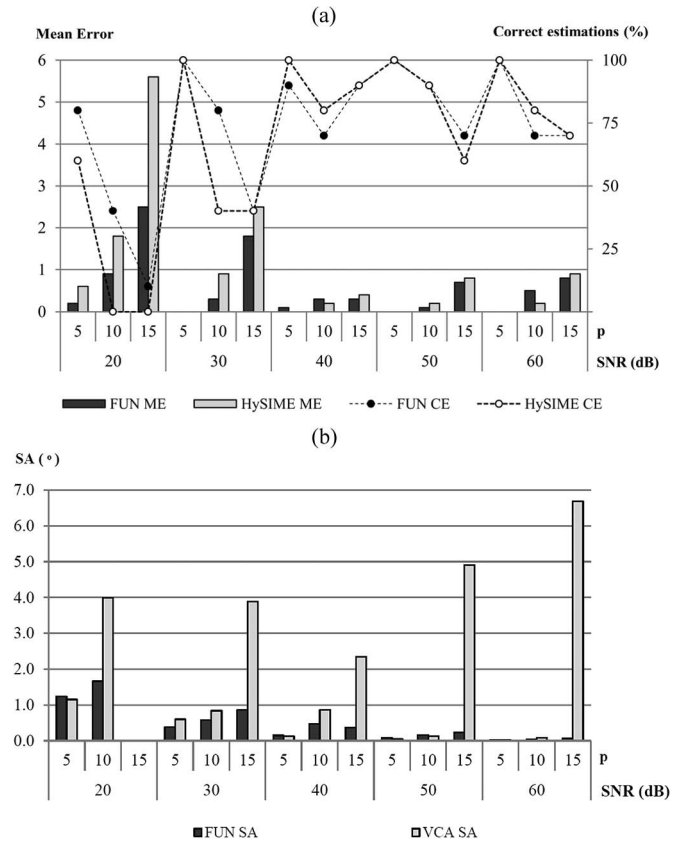

Fig. 8. (a) Number of endmembers estimation. Percentage of correct estimations and mean error obtained using the FUN and HySIME algorithms. (b) Endmembers extraction. Mean spectral angle obtained using the FUN and VCA algorithms.

G. Performance of the FUN Algorithm for Abundance Estimation

Once we have evaluated the performance of the FUN algorithm for the simultaneous estimation of the number of endmember and their extraction, we perform additional simulations in order to assess its usefulness for the estimation of the abundances of the extracted endmembers.

For that purpose, we generate a new set of synthetic hyperspectral images, with the same combinations of configuration parameters used in the previous experiments.

Considering that the number of endmembers is known in advance, the extraction of the endmembers is performed with the FUN algorithm, which is then utilized to estimate the abundances. The results are compared with those obtained with the FCLSU algorithm for the same images and endmembers.

In order to estimate the quality of the results, the root-mean-square error (RMSE) has been calculated in both cases, as follows:

$$\text{RMSE} = \frac{1}{N_p \cdot N_b} \cdot \sqrt{\sum_{i=1}^{N_b} \sum_{j=1}^{N_p} \left(M_{i,j} - M'_{i,j}\right)^2} \qquad (10)$$

where M refers to the real hyperspectral image, and M' refers to the reconstructed one, using the extracted endmembers and

TABLE IV
RMSE VALUES OBTAINED WITH THE FUN, FCFUN, AND FCLSU ALGORITHMS

Image's characteristics		Mean spectral angle (°)	RMSE		
SNR (dB)	p		FUN	FCFUN	FCLSU
20	5	1.989	0.0354	0.0449	0.0363
20	10	3.374	0.0330	0.0804	0.0404
20	15	3.521	0.0343	0.0944	0.0409
30	5	0.629	0.0110	0.0195	0.0111
30	10	1.898	0.0110	0.0282	0.0112
30	15	3.320	0.0109	0.0584	0.0126
40	5	0.215	0.0033	0.0071	0.0033
40	10	1.539	0.0037	0.0127	0.0037
40	15	1.235	0.0035	0.0232	0.0035
50	5	0.046	0.0010	0.0017	0.0011
50	10	1.208	0.0011	0.0100	0.0012
50	15	1.030	0.0011	0.0178	0.0012
60	5	0.290	0.0004	0.0016	0.0005
60	10	0.320	0.0003	0.0046	0.0004
60	15	0.438	0.0003	0.0082	0.0004

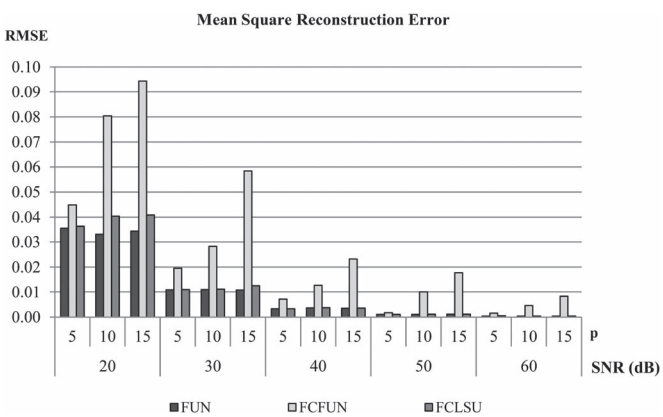

Fig. 9. Mean square reconstruction error (RMSE) obtained with the FUN, FCFUN, and FCLSU algorithms in the simulations performed with a set of synthetic hyperspectral images with different characteristics of noise and with different numbers of endmembers.

abundances; and N_p and N_b refer to the number of pixels and the number of bands in the hyperspectral image, respectively.

The spectral angle of the endmembers obtained by the FUN algorithm, which is used to calculate the abundances, has been also computed. The data have been grouped in sets of ten images, according to their characteristics, and the mean results are shown in Table IV, where RMSE is displayed for the FUN and FCLSU algorithms. Columns headed as FCFUN show the results obtained when the abundances are calculated with the FUN algorithm, and subsequently, the nonnegative constraint and the sum-to-one constraints are applied, as explained in Section III.

We can observe that the FUN algorithm is able to estimate the abundances with a similar accuracy to the one provided by the reference FCLSU algorithm. These facts are graphically displayed in Fig. 9, where it can be also seen that there is little or no difference in the accuracy of the results of the FUN algorithm with and without applying the nonnegative and sum-to-one constraints. This means that the constraints are mostly satisfied inherently.

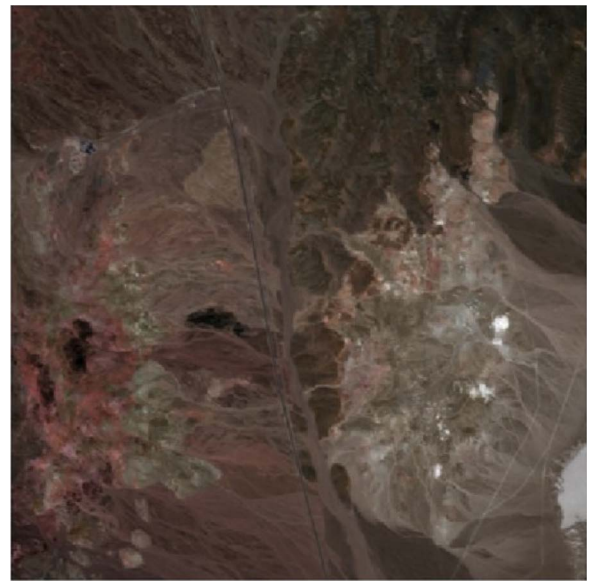

Fig. 10. AVIRIS Cuprite image.

H. Experiments on Real Hyperspectral Data

In order to test the proposed algorithm in a more realistic scenario, the AVIRIS Cuprite image has been also used in this work. This scene is well understood from the mineralogical point of view and consists of 224 spectral bands between 0.4 and 2.5 μm. Prior to the analysis, several bands have been removed due to water absorption and low SNR, resulting in a total of 188 spectral bands. Particularly, we have used in our experiments a portion of the full Cuprite image, with a spatial size of 350 $\times$ 350 pixels. Fig. 10 shows a representation of this image.

Ground truth information of the spectral signatures present in these images is available in the USGS library. The spectral signature of alumite, buddingtonite, calcite, kaolinite, and muscovite, which are present in the Cuprite images, are compared with those obtained with the FUN algorithm in order to evaluate its endmember extraction accuracy.

The FUN algorithm, with $\alpha = 1$ as input parameter, is utilized to perform the entire unmixing process and the abundance estimation. The obtained results are compared with HySIME, VCA, and FCLSU, each for the particular stage of the unmixing process they were designed for.

Moreover, the execution time of each of the algorithms has been computed as an indicator of the computational complexity. Ten simulations have been performed for this portion of the real AVIRIS Cuprite image. The results are shown in Table V, in which we also show the number of endmembers estimated and the spectral angle for the FUN and reference algorithms. Specifically, the spectral angle value is the sum of the spectral angles obtained by comparing the endmembers extracted by the FUN and VCA algorithms with the five available spectral signatures known in the AVIRIS Cuprite scene, which are alumite, buddingtonite, calcite, kaolinite, and muscovite.

Table V also shows the time taken by the FUN algorithm in the different stages of the unmixing process, namely, the noise filtering (column *NF*), the estimation of the number

TABLE V
COMPARISON OF THE RESULTS OBTAINED WITH THE CUPRITE DATA SET

P estimated		MSA		RMSE			Time consumed by the algorithms (s)							
							FUN			Reference algorithms			Total	
FUN	HS	FUN	VCA	FUN	FCFUN	FCLSU	N.F.	P+E	A	HS	VCA	FCLSU	FUN	RA
19	16	6.86	7.67	0.0018	0.0769	0.0067	3.8	2.7	0.1	4.4	3.4	199.6	6.5	207.5
19	16	6.86	7.43	0.0018	0.0769	0.0039	3.7	2.6	0.1	4.4	3.2	211.2	6.4	218.8
19	16	6.86	8.10	0.0018	0.0769	0.0073	3.7	2.6	0.1	4.5	3.1	199.2	6.5	206.8
19	16	6.86	7.63	0.0018	0.0769	0.0054	3.9	2.7	0.1	4.7	3.4	202.7	6.7	210.8
19	16	6.86	8.17	0.0018	0.0769	0.0058	3.7	2.7	0.1	4.7	3.3	188.6	6.5	196.6
19	16	6.86	7.72	0.0018	0.0769	0.0041	3.9	2.7	0.1	4.7	3.4	208.0	6.7	216.1
19	16	6.86	8.71	0.0018	0.0769	0.0073	3.9	2.8	0.1	4.6	3.3	167.0	6.8	174.9
19	16	6.86	8.00	0.0018	0.0769	0.0060	3.9	2.7	0.1	4.7	3.3	191.9	6.7	199.9
19	16	6.86	8.09	0.0018	0.0769	0.0049	3.9	2.7	0.1	4.5	3.3	203.2	6.7	211.0
19	16	6.86	8.59	0.0018	0.0769	0.0068	3.8	2.7	0.1	4.6	3.4	199.9	6.6	207.9

TABLE VI
MEAN RESULTS OBTAINED WITH THE CUPRITE DATA SET

P estimated		MSA		RMSE			Time consumed by the algorithms (s)							
							FUN			Reference algorithms			Total	
FUN	HS	FUN	VCA	FUN	FCFUN	FCLSU	N.F.	P+E	A	HS	VCA	FCLSU	FUN	RA
19	16	6.86	8.01	0.0018	0.0769	0.0058	3.8	2.7	0.1	4.6	3.3	197.1	6.6	205.0

TABLE VII
COMPARISON OF THE SPECTRAL SIGNATURES OBTAINED WITH THE FUN AND VCA ALGORITHMS

Algorithms	Spectral angle (°)				
	e_1	e_2	e_3	e_4	e_5
FUN	4.83	4.26	9.62	10.98	4.64
VCA	11.95	5.85	6.45	14.90	5.57

Reference spectral signatures	
Alunite	e_1
Buddingtonite	e_2
Calcite	e_3
Kaolinite	e_4
Muscovite	e_5

of endmember and simultaneous extraction $(P + E)$, and the abundance estimation (A). The column headed *Total* shows the time consumed by the algorithms to complete the unmixing process. Finally, column *RA* shows the total time taken by the reference algorithms to complete the unmixing process, as the combination of the HySIME, VCA, and FCLSU algorithms.

Table VI shows the mean results obtained for the simulations displayed in Table V, demonstrating the good performance of the FUN algorithm, since the spectral angle and the RMSE are smaller for the FUN algorithm than they are for the reference algorithms. Furthermore, the FUN algorithm has the additional advantage that performs the entire unmixing process more than 31 times faster than the reference algorithms.

Finally, one last simulation has been performed in order to show graphically the results obtained by both the proposed and reference algorithms. Table VII shows the spectral angle scores obtained with the FUN algorithm and with the VCA algorithm when comparing the respective extracted endmembers with respect to the real pure spectral signatures present in the scene, whereas Fig. 11 compares graphically the spectral signatures. Moreover, Fig. 12 shows the root square error (RSE) obtained for each pixel in the hyperspectral image shown in Fig. 11, using the proposed and FCLSU algorithms. The square error of each pixel has been calculated comparing the real Cuprite image with the reconstructed Cuprite one. The reconstruction of the image is done using the extracted endmembers and their calculated abundances with both methods, according to the linear mixing model. These square errors correspond with the RMSE shown in Table VIII. Thus

$$\mathrm{RSE}_j = \frac{1}{N_b} \cdot \sqrt{\sum_{i=1}^{N_b} \left(M_{i,j} - M'_{i,j}\right)^2}. \tag{11}$$

VI. CONCLUSION

In this paper, a new algorithm, named FUN, has been proposed to perform the unmixing process. This algorithm is based on orthogonal projections, using the modified Gram–Schmidt method to perform them. This methodology allows the reutilization of information and a high level of parallelization. Moreover, the algorithm does not perform complex matrix operations, such as the inverse of a matrix or the extraction of eigenvalues and eigenvectors, which makes easier its ulterior hardware implementation.

The FUN algorithm performs simultaneously the estimation of the number of endmembers and their extraction. Once the endmembers have been extracted, this algorithm is also capable of calculating the abundances by doing just some additional operations. In order to verify the well behavior of the FUN algorithm for the different parts of the unmixing process, and for different scenarios, some synthetic hyperspectral images have been generated. These images have been generated varying their abundances distribution, number of endmembers,

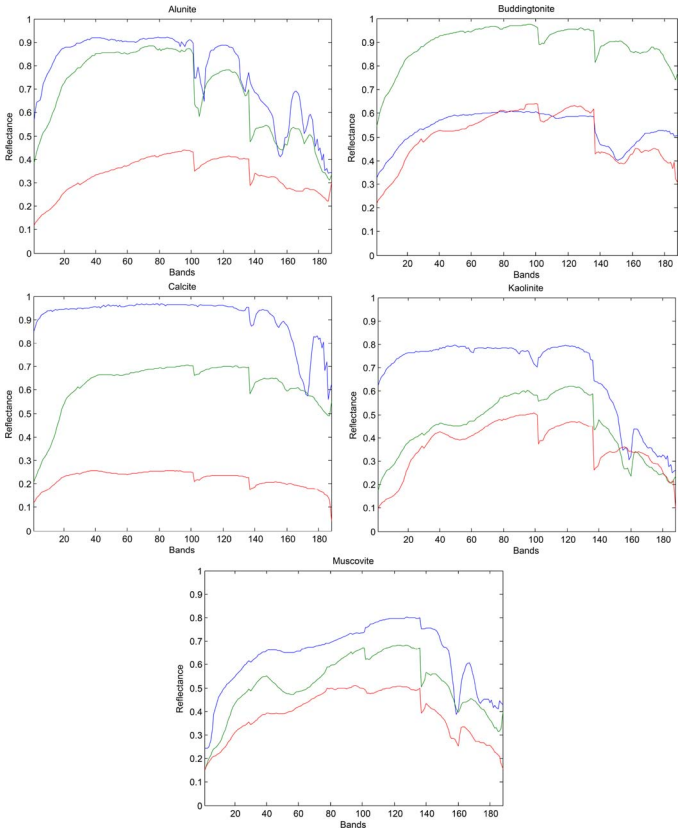

Fig. 11. Comparison of the five spectral signatures in the Cuprite scene (in blue) with respect to the endmembers extracted by the FUN (in green) and VCA (in red) algorithms. (From top to bottom, left to right) Alunite, buddingtonite, calcite, kaolinite, muscovite.

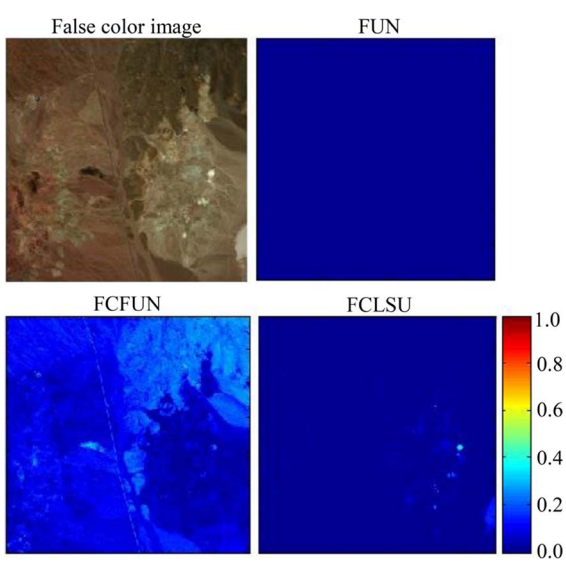

Fig. 12. (a) False color representation of the AVIRIS Cuprite image used. (b) RSE obtained with the FUN algorithm. (c) RSE obtained with the FCFUN algorithm. (d) RSE obtained with the FCLSU algorithm.

endmembers, and SNR. Moreover, in order to compare the results obtained with the FUN algorithm with some of the most relevant algorithms in the state of the art for the same tasks, the entire unmixing process has been also performed using the unmixing chain formed by the HySIME, VCA, and FCLSU algorithms for the same images.

TABLE VIII
RMSE OBTAINED WITH THE FUN, FCFUN, AND FCLSU ALGORITHMS

RMSE		
FUN	FCFUN	FCLSU
0.0018	0.0769	0.0058

The experimental results demonstrate that the accuracy obtained with the FUN algorithm when estimating the number of endmembers and extracting them is similar or better than the accuracy of the HySIME and VCA algorithms. Moreover, the accuracy of the abundances obtained with the FUN algorithm is also similar to the accuracy of the abundances calculated with FCLSU, whereas the FUN algorithm calculates them much faster.

Furthermore, in order to evaluate the FUN algorithm in a more realistic scenario, the AVIRIS Cuprite image has been also used. In this case, the proposed FUN algorithm has been able to reduce in a factor of more than 31 times the time required for processing it, while providing a better unmixing performance than traditional methods.

Hence, it is concluded that the FUN algorithm represents a more advantageous solution than the algorithms of the state of the art, since this new algorithm allows a much simpler and faster hardware implementation, with simpler matrix operations that can be easily parallelized, providing accurate results. This contribution significantly facilitates the implementation of the unmixing process for applications that require real-time performance.

REFERENCES

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

[2] 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.

[3] J. M. Bioucas-Dias and J. M. Nascimento, "Hyperspectral subspace identification," *IEEE Trans. Geosci. Remote Sens.*, vol. 46, no. 8, pp. 2435–2445, Aug. 2008.

[4] J. M. Nascimento and J. M. 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.

[5] M. E. Winter, "N-FINDR: An algorithm for fast autonomous spectral end-member determination in hyperspectral data," in *Proc. SPIE*, 1999, vol. 3753, pp. 266–275.

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

[7] C.-I. Chang, "Orthogonal Subspace Projection (OSP) revisited: A comprehensive study and analysis," *IEEE Trans. Geosci. Remote Sens.*, vol. 43, no. 3, pp. 502–518, Mar. 2005.

[8] R. Neville, K. Staenz, T. Szeredi, J. Lefebvre, and P. Hauff, "Automatic endmember extraction from hyperspectral data for mineral exploration," in *Proc. 21st Can. Symp. Remote Sens.*, 1999, pp. 21–24.

[9] S. Bernabe *et al.*, "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.

[10] S. Lopez *et al.*, "The promise of reconfigurable computing for hyperspectral imaging onboard systems: A review and trends," *Proc. IEEE*, vol. 101, no. 3, pp. 698–722, Mar. 2013.

[11] S. Bernabe, S. López, 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.

[12] S. Lopez, P. Horstrand, G. M. Callico, J. F. Lopez, and R. Sarmiento, "A novel architecture for hyperspectral endmember extraction by means of the modified vertex component analysis (MVCA) algorithm," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 5, no. 6, pp. 1837–1848, Dec. 2012.

[13] USGS Digital Spectral Library. [Online]. Available: http://speclab.cr.usgs.gov/spectral-lib.html

Raúl Guerra was born in Las Palmas de Gran Canaria, Spain, in 1988. He received the industrial engineer degree and the master's degree in telecommunications technologies from the University of Las Palmas de Gran Canaria, Las Palmas de Gran Canaria, in 2012 and 2013, respectively. He is currently working toward the Ph.D. degree in the Institute for Applied Microelectronics, University of Las Palmas de Gran Canaria, funded by the institute to do his research in the Integrated System Design Division.

His current research interests include the parallelization of algorithms for multispectral and hyperspectral images processing and hardware implementation.

Lucana Santos received the telecommunication engineer degree and the Ph.D. degree from the University of Las Palmas de Gran Canaria, Las Palmas de Gran Canaria, Spain, in 2008 and 2014, respectively.

Since then, she has conducted her research activity at the Integrated System Design Division with the Institute for Applied Microelectronics, University of Las Palmas de Gran Canaria. In 2011, she was a Visiting Researcher with the European Space Research and Technology Centre, Noordwijk, The Netherlands, where she did research on hardware architectures for hyperspectral and multispectral image compression on graphic processing units (GPUs) and field-programmable gate arrays (FPGAs), funded by the European Network of Excellence on High Performance and Embedded Architecture and Compilation. From 2012 to 2014, she actively participated in an industrial project in this field for Thales Alenia Space España. She is currently a member of the CCSDS Multispectral/Hyperspectral Data Compression working group sponsored by the European Space Agency. She has coauthored several scientific papers. Her current research interests include hardware architectures for onboard data processing, video coding standards, and reconfigurable architectures.

Dr. Santos has been a reviewer for major international journals in her research areas.

Sebastián López (M'08–SM'15) was born in Las Palmas de Gran Canaria, Spain, in 1978. He received the electronic engineer degree from the University of La Laguna, Santa Cruz de Tenerife, Spain, in 2001 and the Ph.D. degree in electronic engineering from the University of Las Palmas de Gran Canaria, Las Palmas de Gran Canaria, in 2006.

He is currently an Associate Professor developing his research activities in the Integrated Systems Design Division with the Institute for Applied Microelectronics, University of Las Palmas de Gran Canaria. He has authored or coauthored over 80 papers in international journals and conferences. His current research interests include real-time hyperspectral imaging, reconfigurable architectures, high-performance computing systems, and image and video processing.

Dr. López is an AdCom Member of the Spanish Chapter of the IEEE Geoscience and Remote Sensing Society. He is also a program committee member of different international conferences, including the *SPIE Conference on Satellite Data Compression, Communication, and Processing*; the IEEE WORKSHOP ON HYPERSPECTRAL IMAGE AND SIGNAL PROCESSING: EVOLUTION IN REMOTE SENSING; and the *SPIE Conference on High Performance Computing in Remote Sensing*. He acted as one of the program chairs in the last two aforementioned conferences for their 2014 editions and will be the Program Chair for the *SPIE Conference on High Performance Computing in Remote Sensing* in 2015. He was an Associate Editor of the IEEE TRANSACTIONS ON CONSUMER ELECTRONICS from 2008 to 2013. He is currently an Associate Editor of the IEEE JOURNAL OF SELECTED TOPICS IN APPLIED EARTH OBSERVATIONS AND REMOTE SENSING (JSTARS). He also currently serves as an active Reviewer of IEEE JSTARS; the IEEE TRANSACTIONS ON GEOSCIENCE AND REMOTE SENSING; the IEEE Geoscience and Remote Sensing Letters; the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY; the Journal of Real Time Image Processing; Microprocessors and Microsystems: Embedded Hardware Design (MICPRO); and the *IET Electronics Letters*, among others. Moreover, he was the Guest Editor of the special issue entitled "Design and Verification of Complex Digital Systems" that was published in 2011 in the *Microprocessors and Microsystems: Embedded Hardware Design* (MICPRO) journal, and he is one of the guest editors of the special issue entitled "Hyperspectral remote sensing," which will be published in IEEE JSTARS in 2015. He has been a recipient of regional and national awards for his Curriculum vitae (CV) during his electronic engineer degree studies.

Roberto Sarmiento received the Ph.D. degree in electronic engineering from the University of Las Palmas de Gran Canaria (ULPGC), Las Palmas de Gran Canaria, Spain, in 1991.

He is currently a Full Professor in the area of electronic engineering with the Telecommunication Engineering Faculty, ULPGC. He contributed to the birth of this center in 1989. He was the Dean of the Faculty from 1994 to 1998 and the Vice Chancellor for Academic Affairs and Staff from 1998 to 2003 with ULPGC. In 1993, he was a Visiting Professor with The University of Adelaide, Adelaide, S.A., Australia, and then with Edith Cowan University, Mount Lawley, Australia. He cofounded the Institute for Applied Microelectronics and has been the Director of the Integrated Systems Design Division of this institute. He has been awarded with four six-year research periods by the National Agency for the Research Activity Evaluation in Spain. He has participated in over 35 projects and research programs funded by public and private organizations, from which he has been leader researcher in 16 of them. He has conducted several agreements with companies for the design of high-performance integrated circuits, being the most remarkable the collaboration with Vitesse Semiconductor Corporation (California) and Thales Alenia Space (Spain). Since 1990, he has authored or coauthored over 40 journal papers and book chapters and over 120 conference papers. His research interests include multimedia processing and video coding standard systems, reconfigurable architectures, and real-time processing and compression of hyperspectral imaging.