

# A Fully Constrained Linear Spectral Unmixing Algorithm Based on Distance Geometry

Hanye Pu, Wei Xia, *Student, IEEE*, Bin Wang, *Senior Member, IEEE*, and Geng-Ming Jiang, *Member, IEEE*

**Abstract**—Under the linear spectral mixture model, hyperspectral unmixing can be considered as a convex geometry problem, in which the endmembers are located in the vertices of simplex enclosing the hyperspectral data set and the barycentric coordinates of observation pixels with respect to the simplex correspond to the abundances of endmembers. Based on distance geometry theory, in this paper we propose a new approach for abundance estimation of mixed pixels in hyperspectral images. With the endmember signatures, which is known *a priori* or can be obtained from the endmember extraction algorithms, the proposed method automatically estimates the abundances of endmembers at each pixel using convex geometry concepts and distance geometry constraints. In the algorithm, denoting the pairwise distances with Cayley–Menger matrix makes it easy to calculate the barycentric coordinates of the observation pixels. Another characteristic of this algorithm is that the optimal estimated points of observation pixels as well as the least distortion in geometric structure of original data set can be obtained with the distance geometry constraint. Simultaneously, the use of barycenter of simplex builds an accurate and efficient method to estimate endmembers with zero abundance and, as a result, the subsimplex containing the estimated points is obtained. A comparative study and analysis based on Monte Carlo simulations and real data experiments is conducted among the proposed algorithm and three state-of-the-art algorithms: fully constrained least squares (FCLS), FCLS computed using constrained sparse unmixing by variable splitting and augmented Lagrangian, and simplex-projection unmixing (SPU). The experimental results show that the proposed algorithm always provides the best unmixing accuracy and when the number of endmembers is not very large the algorithm has a lower computational complexity.

**Index Terms**—Abundance estimation, affine endmember-hull, barycentric coordinate, distance geometry constraint, endmember-simplex, exterior point, hyperspectral unmixing, interior point.

## I. INTRODUCTION

**D**UE TO the rapid development of hyperspectral imaging technology, hyperspectral image processing has become a fast growing technique in the field of image processing during the last several decades. Hyperspectral imagery makes use of as many as hundreds of contiguous spectral bands across

from ultraviolet to short-wave infrared regions to expand the capability of multispectral sensors [1]. A lot of applications of hyperspectral images, such as target detection, environmental monitoring, mineral exploration, food safety, and so on, are explored [1]–[5]. However, because of the relatively coarse spatial resolution of remote sensing sensors and mixing effects in surfaces, a single pixel may cover more than one constituent material within the instantaneous field of view of the sensor, and the pixel is known as mixed pixel [6], [7]. It is the common presence of mixed pixels that makes it hard to apply the traditional pixel level methods for hyperspectral imagery in certain domains such as identification and detection of ground targets [8], [9].

Unmixing the mixed pixels into a collection of constituent materials and their corresponding proportions, known as hyperspectral unmixing, is one of the most useful tasks in hyperspectral image processing, particularly when it is used in subpixel level analysis. To deal with this problem, a series of mathematical or physical mixture models are proposed for the interpretation of the mixed pixels in the hyperspectral images [10]. Among them, the linear spectral mixture model (LSMM), which is conventionally and effectively used in hyperspectral unmixing, is the one with the assumption that only single reflection of the endmembers is present within a pixel and the multiple scattering among distinct endmembers is negligible [5], [6], and [10]. Hyperspectral unmixing based on LSMM is referred to as linear spectral unmixing (LSU). Generally, the LSU methods consist of two procedures, endmember extraction, and abundance estimation. Some algorithms for endmember extraction or abundance estimation or both procedures simultaneously are proposed over the past decades [5], [6]. In this paper, we focus exclusively on the abundance estimation.

Abundance estimation is to infer the proportions of different endmembers at each pixel with the endmember signatures known or extracted by endmember extraction algorithms (EEAs). The present approaches for abundance estimation can be roughly categorized into statistical- and geometrical-based [2]. A number of variations of the least squares concepts are adopted in the statistical-based algorithms to reflect the unique circumstances associated with hyperspectral data [11]–[15]. Neglecting the physical meaning of abundance, the unconstrained method, e.g., unconstrained orthogonal subspace projection approach [16], is simple and straightforward but the difficulty is how to state explanations for the cases with the negative abundances and total abundances higher than 100%. Generally, the fractional abundances are subject to the sum-to-one and nonnegativity constraints.

Manuscript received January 15, 2012; revised January 29, 2013; accepted February 9, 2013. Date of publication April 9, 2013; date of current version December 12, 2013. This work was supported in part by the National Natural Science Foundation of China under Grant 61071134 and Grant 41271012, the Innovation Program of Shanghai Municipal Education Commission under Grant 13ZZ005, and the Research Fund for Doctoral Program of Higher Education of China under Grant 20110071110018.

The authors are with the Key Laboratory of Wave Scattering and Remote Sensing Information, Fudan University, Shanghai 200433, China and also with the Department of Electronic Engineering, Fudan University, Shanghai 200433, China (e-mail: wangbin@fudan.edu.cn).

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.2013.2248013

To overcome the hurdles in the unconstrained method above, the partially constrained approaches with either sum-to-one constraint or nonnegativity constraint are proposed, such as sum-to-one constrained least squares (SCLS) method [17], nonnegativity constrained least squares (NCLS) [18] method and so on. Because of the existence of closed form solution for the sum-to-one constraint, the SCLS algorithm has a lower computational complexity but a worse result than the NCLS algorithm. Although the partially constrained approaches have low computational demands, suboptimal approximations to actual mixing fractions may be yielded as two constraints are carried out in sequence and not simultaneously [12], [19]. Therefore, several studies have sought to develop efficient algorithms to solve the fully constrained problem [11], [12]. Because of the loss of closed form solution for the nonnegative constraint, some numerical methods like quadratic programming are adopted to incorporate the two constraints. However, high computational complexity follows with the iterative operations involved in the optimization or maximization procedure, which is not suitable for large-scale data [20].

A computationally efficient method named fully constrained least squares (FCLS) is proposed in [12]. Considering the hyperspectral unmixing as least-squares regression with the sum-to-one and nonnegativity constraints, the FCLS algorithm applies a standard active set method [18] for the nonnegativity constraint. A geometrical interpretation of FCLS is provided in [11]. In theory, the FCLS algorithm can get the optimal solution of constrained least squares. However, the iterative operations in the FCLS algorithm are inevitable. As an alternative version, the fuzzy sets (FSs) algorithm [21] avoids the iterative procedure by giving the analytic solution to the FCLS algorithm in terms of FS operations. However, when the number of endmembers is larger, a larger numerical error will occur. Another drawback of the FCLS and FSs is overfitting. In spite of the fact that the least squares solutions always fit the original data well, the fitting is not always desired especially when the noise level of the data is high [11], [22]. Based on the alternating direction method of multipliers (ADMM) [23], sparse unmixing by variable splitting and augmented Lagrangian (SUnSAL) and constrained SUnSAL (C-SUnSAL) [24] decompose the difficult optimization problems arising in spectral unmixing into a sequence of simpler ones.

Considering hyperspectral unmixing to be a convex geometry problem, geometrical-based methods deem that hyperspectral observation pixels are enclosed in a simplex whose vertices correspond to the endmembers and the abundances of the endmembers, at each pixel, are just equal to the barycentric coordinates with respect to the simplex [25]–[30]. For the barycentric coordinates, which are first introduced by Möbius as a special kind of homogeneous coordinates with respect to the vertices of a simplex [31], the aforementioned two constraints are satisfied in a natural way. The difficulty for the simplex-based algorithms is to find the estimated points of noisy observation pixels in the simplex spanned by endmembers. It can also be converted into a quadratic programming problem, in which the iterative algorithm is involved [27]. Using the projection onto the canonical simplex, the algorithms in

[28] and [29] give a fast and fully constrained result. However, the result is also suboptimal for hyperspectral unmixing.

As a geometrical-based algorithm, the simplex-projection unmixing (SPU) algorithm [30] is a recently proposed supervised unmixing algorithm, giving almost the same optimal solution with FCLS. The SPU algorithm proposes a way to find the orthogonal projection of a point onto the simplex plane and considers the projection as the estimated point in the simplex spanned by endmembers. A definition of bissective cone is introduced to obtain an efficient method to identify the endmembers with zero abundance by combining the incenter of simplex. Avoiding the optimization or maximization procedures, the SPU algorithm, which has a recursive implementation, has a low computational complexity for a small number of endmembers. However, when the number of endmembers is high, the computational complexity of SPU algorithm is still high because of the recursive implementation, and the algorithm also suffers the overfitting problem.

In this paper, we present a new abundance estimation algorithm based on convex geometry theory, which explicitly takes into account the geometric structure of data set. Different from the other simplex geometry-based algorithms, the proposed algorithm takes advantage of the pairwise distances between observation pixels and endmembers in the form of Cayley–Menger matrix, which is an important concept in the distance geometry [32]. Following up on the study of distance geometry, a new formula for calculating barycentric coordinates is proposed in this paper, which is fit for the calculation of barycentric coordinates with respect to the low-dimensional simplex in the high-dimensional data space. Considering the geometry relationship between observation pixels and endmembers, we introduce the distance geometry constraint and design a fast and efficient recursive method to get the final abundances of observation pixel. As a result, the unmixing procedure can be divided into three parts:

- 1) calculating the barycentric coordinates using the new formula;
- 2) location estimation with an algorithm based on distance geometry constraint;
- 3) affine subspace determination by an algorithm based on the barycenter of simplex.

Finally, an optimal result with the least distortion in geometric structure of original data is obtained. Moreover, the proposed method makes full use of the characteristic of Hermitian matrix, e.g., symmetry and nonsingularity, to reduce the computational complexity.

The rest of this paper is organized as follows. In Section II, we give a brief description of the LSMM. Some important concepts of convex geometry theory are also introduced in this Section. The proposed algorithm is then introduced in Section III. The algorithm is composed of three parts: barycentric coordinates calculation algorithm based on Cayley–Menger matrix described in Section III-A location estimation algorithm for exterior points in Section III-B and subspace determination based on interior point in Section III-C. The holistic summarization of the proposed algorithm will be given in Section III-D. in the form of pseudocode. In addition,

we will analyze the computational complexity of the proposed algorithm in Section III-E. The performance evaluations using simulated and real hyperspectral data are presented in Section IV. Finally, some conclusions are drawn in Section V.

For ease of illustration, we give explicitly the notations that we will keep unchanged in this paper. Column vectors and matrices are expressed by lowercase and uppercase boldface characters, e.g.,  $\mathbf{c}$  and  $\mathbf{C}$ , respectively, and italics like  $c$  denote scalars. The superscripts  $p$  in vector  $\mathbf{c}^p$  and square matrix  $\mathbf{C}^p$  indicate their dimensionality, respectively, whereas  $c^p$  is the  $p$ th power of  $c$ . The transpose and determinant of a matrix  $\mathbf{C}$  are  $\mathbf{C}^T$  and  $\det(\mathbf{C})$ , respectively. Following the MATLAB notation,  $\text{diag}(\mathbf{A})$  is the column vector containing all the diagonal elements of square matrix  $\mathbf{A}$ . The symbol  $\mathbf{C}_{ij}$  stands for the element of matrix  $\mathbf{C}$  in the  $i$ th row and the  $j$ th column.  $|c|$  is the absolute value of  $c$ , whereas  $\|\bullet\|_2$  is the Euclidean 2-norm.

## II. THEORETICAL BACKGROUND

Here, we first briefly review the LSMM, which is widely used in hyperspectral unmixing. Next, we introduce two basic concepts of convex geometry that are used to discuss the convex geometry theory for hyperspectral data.

### A. Linear Spectral Mixture Model

According to LSMM, the observation pixel in the hyperspectral image can be expressed as linear combination of some pure materials (endmembers) and their fractional proportions (abundances) as follows

$$\mathbf{x} = \sum_{i=1}^p s_i \mathbf{e}_i + \mathbf{n} = \mathbf{E}\mathbf{s} + \mathbf{n} \quad (1)$$

where  $\mathbf{x}$  is an  $L \times 1$  column vector that describes the spectrum of the observation pixel,  $\mathbf{E}$  given by  $\mathbf{E} = (\mathbf{e}_1, \mathbf{e}_2, \dots, \mathbf{e}_p)$  is a  $L \times p$  matrix containing  $p$  endmember signatures,  $p$ -dimensional column vector  $\mathbf{s}$  composed of abundance coefficients  $s_i$  ( $i = 1, 2, \dots, p$ ) is the abundances of endmembers at the observation pixel,  $L$ -dimensional error vector  $\mathbf{n}$  models the error and noise effects, and  $L$ ,  $N$ , and  $p$  are the number of bands, pixels, and endmembers, respectively. In LSMM, we consider the term  $\mathbf{n}$  as zeros-mean, additive Gaussian noise vector, which is a reasonable and widely used assumption in designing hyperspectral unmixing algorithms [1], [2], [5], [6].

Because of physical reasons, the abundances in (1) have to meet the two constraints, namely, abundance nonnegativity constraint (ANC) that means that the fractional abundances within a pixel cannot be negative and abundance sum-to-one constraint (ASC) that means that the sum of the fractional abundances within a pixel must be one, respectively

$$s_i \geq 0, \quad \sum_{i=1}^p s_i = 1, \quad (i = 1, 2, \dots, p). \quad (2)$$

For convenience, (1) can be written in the matrix form

$$\mathbf{X} = \mathbf{E}\mathbf{S} + \mathbf{N} \quad (3)$$

where  $\mathbf{X}$  is the  $L \times N$  matrix whose columns are the observation pixels, each column of  $p \times N$  matrix  $\mathbf{S}$  is the abundances of corresponding observation pixel, and  $\mathbf{N}$  is a  $L \times N$  noise matrix.

### B. Two Convex Geometric Concepts

As our approach is fundamentally based on convex geometry, we employ two important geometric concepts, namely, affine hull and simplex by introducing two definitions about a collection of points.

*Definition 1 (Affine Hull [33]):* Considering point set  $V = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_k\}$  in general position in  $\mathbf{R}^L$ , i.e.,  $\{\mathbf{x}_2 - \mathbf{x}_1, \mathbf{x}_3 - \mathbf{x}_1, \dots, \mathbf{x}_k - \mathbf{x}_1\}$  are affine independent, the affine hull (plural: hulls) spanned by them can be defined as

$$\text{aff}\{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_k\} = \left\{ \mathbf{x} = \sum_{i=1}^k t_i \mathbf{x}_i \mid \sum_{i=1}^k t_i = 1, t_i \in \mathbf{R} \right\}. \quad (4)$$

The affine hull is the smallest flat containing  $V$ .

*Definition 2 (Simplex [34]):* Considering point set  $V = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_k\}$  in general position in  $\mathbf{R}^L$ , i.e.,  $\{\mathbf{x}_2 - \mathbf{x}_1, \mathbf{x}_3 - \mathbf{x}_1, \dots, \mathbf{x}_k - \mathbf{x}_1\}$  are affine independent, the simplex (plural: simplices) spanned by them can be expressed in the form

$$\Delta \mathbf{x}_1 \mathbf{x}_2 \dots \mathbf{x}_k = \left\{ \mathbf{x} = \sum_{i=1}^k t_i \mathbf{x}_i \mid 0 \leq t_i \leq 1, \sum_{i=1}^k t_i = 1, t_i \in \mathbf{R} \right\}. \quad (5)$$

The dimension of  $\Delta \mathbf{x}_1 \mathbf{x}_2 \dots \mathbf{x}_k$  is  $k-1$  and we can call it  $(k-1)$ -simplex for short. The points in  $V$  are called the vertices of the simplex, whereas the simplex spanned by a  $(k-2)$ -point subset of  $V$  is a face of the simplex, said to be the face opposite to the remaining vertex.

Fix a point  $\mathbf{x} \in \text{aff}\{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_k\}$ . There exist unique scalars  $t_1, t_2, \dots, t_k$  satisfying (4) and we call the ordered multiple  $(t_1, t_2, \dots, t_k)$  barycentric coordinate of  $\mathbf{x}$  with respect to the  $(k-1)$ -simplex  $\Delta \mathbf{x}_1 \mathbf{x}_2 \dots \mathbf{x}_k$  with  $\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_k$  as base points [35].

For a collection of affine independent points  $\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_k$  in  $\mathbf{R}^L$ , according to the position relationship between the observation points and simplex  $\Delta \mathbf{x}_1 \mathbf{x}_2 \dots \mathbf{x}_k$ , we divide them into three types:

- 1) the first type is called interior points, the points inside the simplex (including inside the faces of simplex), which are given by  $\mathbf{x} \in \Delta \mathbf{x}_1 \mathbf{x}_2 \dots \mathbf{x}_k$ ;
- 2) the second type is named exterior points I, the points outside the simplex but inside the affine hull spanned by vertices, denoted by  $\mathbf{x} \notin \Delta \mathbf{x}_1 \mathbf{x}_2 \dots \mathbf{x}_k$ , and  $\mathbf{x} \in \text{aff}\{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_k\}$ ;
- 3) the final type is named exterior points II, including the points outside the affine hull, i.e.,  $\mathbf{x} \notin \text{aff}\{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_k\}$ .

A geometric illustration of affine hull and simplex in 2-D case with vertex set  $\{\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3\}$  is shown in Fig. 1. The affine hull  $\text{aff}\{\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3\}$  is the hyperplane containing the

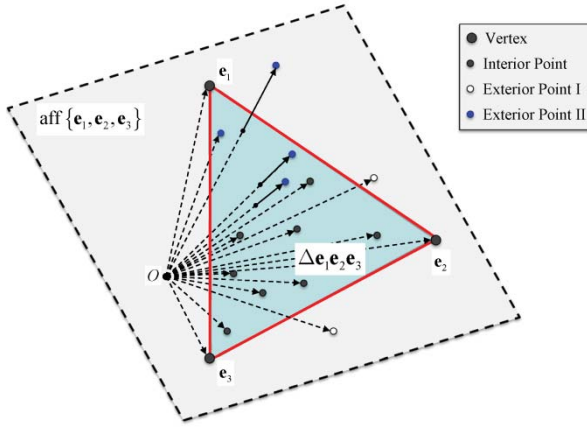


Fig. 1. Geometric illustration for affine hull and simplex in 2-D case ( $k = 3$ ).

three vertices, and the simplex  $\Delta e_1 e_2 e_3$  is the triangle on the hyperplane.

### C. Convex Geometry for Hyperspectral Data

From convex geometry theory, in the hyperspectral data set under LSMM all the data points should lie in a simplex spanned by endmembers [25], which we name endmember-simplex for convenience. Accordingly, we call the affine hull spanned by endmembers affine endmember-hull.

If a noise-free hyperspectral data set satisfies the LSMM, all the observation pixels are in the  $(p-1)$ -dimensional endmember-simplex, i.e., they are the interior points of the simplex. However, when the data are polluted by noise, some interior points will slide out of the simplex and become exterior points I or II as shown in Fig. 1.

For the interior points, we can use the barycentric coordinates for the endmember-simplex as their fractional abundances and the two constraints in (2) are satisfied well naturally [25]. Because of the failure to nonnegativity, the barycentric coordinates of exterior points I cannot be used as the estimated abundances directly. Regarding the exterior points II, which slide out of the affine endmember-hull, it is impossible to express them as linear combination of endmembers. Hence, the appropriate estimated points inside the simplex must be found to replace the exterior points I and II.

For exterior points II, our method is designed to find the optimal estimated points in the affine endmember-hull by solving a distance geometry constraint problem. If the estimated points lie in the endmember-simplex, we will get their abundances by a barycentric coordinate calculation algorithm. Otherwise, the estimated points will become the exterior points I for the endmember-simplex. For exterior points I, under a reasonable assumption, a fast and appropriate method is designed to judge the endmembers whose abundances are zeros. Then, these endmembers will be removed from the endmember-set to get the endmember-subsimplex. Therefore, the primary exterior points I will become the exterior points II for the endmember-subsimplex, and a new recursive process will start to search for the abundances of the remaining

endmembers. In the next section, we will explain the proposed algorithm in detail.

## III. PROPOSED ALGORITHM

Here, we propose an abundance estimation algorithm based on the distance geometry to tackle the problems discussed above. The aim of the algorithm is to find the optimal estimated points of exterior points in the endmember-simplex or endmember-subsimplex and compute the barycentric coordinates of both estimated points and interior points, which are considered as the fractional abundances. The proposed algorithm has a recursive implementation.

### A. Cayley–Menger Matrix-Based Barycentric Coordinate Calculation Algorithm

For an observation pixel inside the affine endmember-hull, the barycentric coordinate with respect to the endmember-simplex can be obtained from the distances between it and the endmembers. Based on this idea, here, we present a barycentric coordinate calculation method using the Cayley–Menger matrix.

A Cayley–Menger matrix with dimension  $(p+1) \times (p+1)$  formed by endmember-set  $\{e_1, e_2, \dots, e_p\}$  can be defined as [32]

$$\mathbf{C}^{p+1}(e_1, e_2, \dots, e_p) = \begin{pmatrix} 0 & 1 & 1 & 1 & \dots & 1 \\ 1 & 0 & d_{1,2}^2 & d_{1,3}^2 & \dots & d_{1,p}^2 \\ 1 & d_{2,1}^2 & 0 & d_{2,3}^2 & \dots & d_{2,p}^2 \\ 1 & d_{3,1}^2 & d_{3,2}^2 & 0 & \dots & d_{3,p}^2 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & d_{p,1}^2 & d_{p,2}^2 & d_{p,3}^2 & \dots & 0 \end{pmatrix} \quad (6)$$

where  $d_{i,j} = \|e_i - e_j\|$  is the Euclidean distance between two endmembers. The determinant of  $\mathbf{C}^{p+1}$  is known as Cayley–Menger determinant, which is widely used in distance geometry theory to deal with Euclidean geometry in spaces where distance is defined and invariant [32].

As a Hermitian matrix,  $(p+2)$ -dimensional matrix  $\mathbf{C}^{p+2}$  formed by  $(p+1)$  points, e.g.,  $p$  endmembers and an observation pixel  $\mathbf{q}$ , can be rewritten as a  $2 \times 2$  partitioned matrix:

$$\mathbf{C}^{p+2} = \begin{pmatrix} \mathbf{C}^{p+1} & \mathbf{c}^{p+1} \\ (\mathbf{c}^{p+1})^T & 0 \end{pmatrix} \quad (7)$$

where  $\mathbf{c}^{p+1} = (1 \ (\mathbf{d}^p)^T)^T$ ,  $\mathbf{d}^p = (d_{1,q}^2, d_{2,q}^2, \dots, d_{p,q}^2)^T$  contains the square distances from  $\mathbf{q}$  to endmembers. Then, using the expansion in cofactors of determinant, we can obtain

$$\det(\mathbf{C}^{p+2}) = -(\mathbf{c}^{p+1})^T (\mathbf{C}^{p+1})^{-1} \mathbf{c}^{p+1} \det(\mathbf{C}^{p+1}). \quad (8)$$

The sign of  $\det(\mathbf{C}^{p+2})$  is  $(-1)^{p+1}$  and when the simplex vanishes,  $\det(\mathbf{C}^{p+2}) = 0$  [32], so we have  $c = (\mathbf{c}^{p+1})^T (\mathbf{C}^{p+1})^{-1} \mathbf{c}^{p+1} \geq 0$ .

Herein we introduce the Theorem 1, which gives a method to calculate the barycentric coordinates of observation pixels with respect to the endmember-simplex.

*Theorem 1:* Let  $\Delta \mathbf{e}_1 \mathbf{e}_2 \cdots \mathbf{e}_p$  be the endmember-simplex spanned by endmember-set  $\{\mathbf{e}_1, \mathbf{e}_2, \dots, \mathbf{e}_p\}$ . For the observation point  $\mathbf{q}$  inside the affine hull  $\text{aff}\{\mathbf{e}_1, \mathbf{e}_2, \dots, \mathbf{e}_p\}$ , the barycentric coordinate  $\mathbf{s}$  with respect to the endmember-simplex which is given by  $(s_1, s_2, \dots, s_p)^T$  satisfies

$$\begin{pmatrix} \varsigma \\ \mathbf{s} \end{pmatrix} = (\mathbf{C}^{p+1})^{-1} \begin{pmatrix} 1 \\ \mathbf{d} \end{pmatrix} \quad (9)$$

where  $\mathbf{C}^{p+1}$  (6) is the Cayley–Menger matrix corresponding to the endmember-set, the scalar  $\varsigma = -\mathbf{s}^T \mathbf{d}$  and the  $p$ -dimensional column vector  $\mathbf{d} = (d_{1,q}^2, d_{2,q}^2, \dots, d_{p,q}^2)^T$  is the square distance from the point  $\mathbf{q}$  to the vertices of endmember-simplex with  $d_{i,q} = \|\mathbf{q} - \mathbf{e}_i\|$  ( $i = 1, 2, \dots, p$ ).

The proof of Theorem 1 can be found in Appendix A.

Equation (9) only involves the distances between the observation point and endmembers. In addition, the nonsingularity and symmetry of  $\mathbf{C}^{p+1}$  make it easy to calculate the barycentric coordinate using the inverse of matrix, which reduces the computational complexity remarkably.

Let us emphasize that the Cayley–Menger matrix-based barycentric coordinate calculation (CMBCC) algorithm can only be used for the interior points and the exterior points I. For the former, the two constraints in (2) are satisfied and the fractional abundances can be obtained using (9) directly, whereas for the latter, because of the failure to the nonnegativity constraint in (2), the solutions to (9) cannot be considered as the abundances directly. For the points out of the affine endmember-hull, i.e., exterior points II, the solutions obtained by CMBCC do not make sense. So another subalgorithm is needed to obtain their estimated points inside the affine endmember-hull, which are the exterior points I or II. The details of the algorithm are in the next Subsection.

### B. Distance Geometry Constraint-Based Localization Estimation Algorithm

From (9), with the given endmember-set, the key to compute the barycentric coordinate of observation pixel  $\mathbf{q}$  is to compute the  $p$ -dimensional square Euclidean distance vector  $\mathbf{d}^p$ . However, for the observation pixel out of the affine endmember-hull, as the  $\mathbf{d}^p$  cannot be used in (9) directly, an estimated vector of  $\mathbf{d}^p$ , which is denoted as  $\hat{\mathbf{d}}^p$ , must be calculated beforehand.

We define the vector  $\boldsymbol{\varepsilon}^p$  as

$$\boldsymbol{\varepsilon}^p = \hat{\mathbf{d}}^p - \mathbf{d}^p \quad (10)$$

where the vector  $\boldsymbol{\varepsilon}^p$  is an error in the square distance vector.

Thus, a method is developed to estimate the optimal substitution  $\hat{\mathbf{d}}^p$  in the endmember-hull with the least error  $\boldsymbol{\varepsilon}^p$ . We consider it as a distance geometry constraint problem that describes the geometric relations between the observation pixel and the endmembers as quadratic algebraic constraints.

*Theorem 2 (Distance Geometry Constraint (Theorem 112.1 in [32])):* Considering a  $k$ -tuple of points  $(\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_k)$  in  $n$ -dimensional Euclidean space with  $k \geq n + 1$ , the

rank of the  $k + 1$  dimensional Cayley–Menger matrix  $\mathbf{C}^{k+1}(\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_k)$  [defined analogously to (6)] is at most  $n + 1$ .

The proof of Theorem 2 is not given here. The interested reader is encouraged to consult [32].

Let  $\mathbf{C}^{p+2}$  be the  $(p + 2)$ -dimensional Cayley–Menger matrix obtained from the optimal estimated pixel  $\hat{\mathbf{q}}$  in the endmember-hull and endmembers which lie in the  $p$ -dimensional Euclidean space. The distance geometry constraint in Theorem 2 implies that the rank of  $\mathbf{C}^{p+2}$  is at most  $p + 1$ , i.e.,  $\det(\mathbf{C}^{p+2}) = 0$ . Using (8) we can get

$$(\mathbf{c}^{p+1})^T (\mathbf{C}^{p+1})^{-1} \mathbf{c}^{p+1} = 0 \quad (11)$$

with  $\mathbf{c}^{p+1} = (1 \ (\hat{\mathbf{d}}^p)^T)^T$ ,  $\hat{\mathbf{d}}^p = (\hat{d}_{1,q}^2, \hat{d}_{2,q}^2, \dots, \hat{d}_{p,q}^2)^T$ , and  $\hat{d}_{i,q} = \|\mathbf{e}_i - \hat{\mathbf{q}}\|$ .

The inverse of Hermitian matrix  $\mathbf{C}^{p+1}$  can be expressed as

$$(\mathbf{C}^{p+1})^{-1} = \begin{pmatrix} \rho & (\boldsymbol{\delta}^p)^T \\ \boldsymbol{\delta}^p & \mathbf{D}^p \end{pmatrix}. \quad (12)$$

Combining (11) and (12), we can obtain

$$(\mathbf{c}^{p+1})^T (\mathbf{C}^{p+1})^{-1} \mathbf{c}^{p+1} = (\boldsymbol{\varepsilon}^p)^T \mathbf{A} \boldsymbol{\varepsilon}^p - 2 (\boldsymbol{\varepsilon}^p)^T \mathbf{b} - c = 0 \quad (13)$$

with the following defined matrix notations:

$$\begin{cases} -\mathbf{D}^p = \mathbf{A} \\ \boldsymbol{\delta}^p + \mathbf{D}^p \mathbf{d}^p = \mathbf{b} \\ (1 \ (\mathbf{d}^p)^T) \begin{pmatrix} \rho & (\boldsymbol{\delta}^p)^T \\ \boldsymbol{\delta}^p & \mathbf{D}^p \end{pmatrix} \begin{pmatrix} 1 \\ \mathbf{d}^p \end{pmatrix} = c \end{cases} \quad (14)$$

where  $\mathbf{A}$  is a positive-semidefinite Hermitian matrix [36] and the nonnegative scalar  $c$  denotes the residual volume error.

Therefore, the distance geometry constraint problem can be converted into the following quadratic equality constrained convex formulation:

$$\boldsymbol{\varepsilon}^p = \arg \min_{\boldsymbol{\varepsilon}^p} \|\boldsymbol{\varepsilon}^p\|_2 \quad \text{s.t.} \quad (\boldsymbol{\varepsilon}^p)^T \mathbf{A} \boldsymbol{\varepsilon}^p - 2 (\boldsymbol{\varepsilon}^p)^T \mathbf{b} - c = 0. \quad (15)$$

Here, we should point out that, with the least squares problem which minimizes the squared distance errors, the geometric relations between the observation pixel and the endmembers can be kept approximately in the search for the optimal estimated point.

The optimal solution to (15) can be obtained using the Lagrange multiplier methodology by minimizing the Lagrangian function

$$L(\boldsymbol{\varepsilon}^p, \eta) = \frac{1}{2} (\boldsymbol{\varepsilon}^p)^T \boldsymbol{\varepsilon}^p - \frac{1}{2} \eta \left( (\boldsymbol{\varepsilon}^p)^T \mathbf{A} \boldsymbol{\varepsilon}^p - 2 (\boldsymbol{\varepsilon}^p)^T \mathbf{b} - c \right) \quad (16)$$

where  $\eta$  is the Lagrange multiplier. Differentiating (16) with respect to  $\boldsymbol{\varepsilon}^p$  gives

$$\boldsymbol{\varepsilon}^p - \eta \mathbf{A} \boldsymbol{\varepsilon}^p + \eta \mathbf{b} = 0 \quad (17)$$

which yields

$$\boldsymbol{\varepsilon}^p(\eta) = -\eta (\mathbf{I} - \eta \mathbf{A})^{-1} \mathbf{b}. \quad (18)$$

Finding  $\boldsymbol{\varepsilon}^p$  in (18) encompasses two difficulties as follows: the estimation of the Lagrange multiplier  $\eta$  which achieves

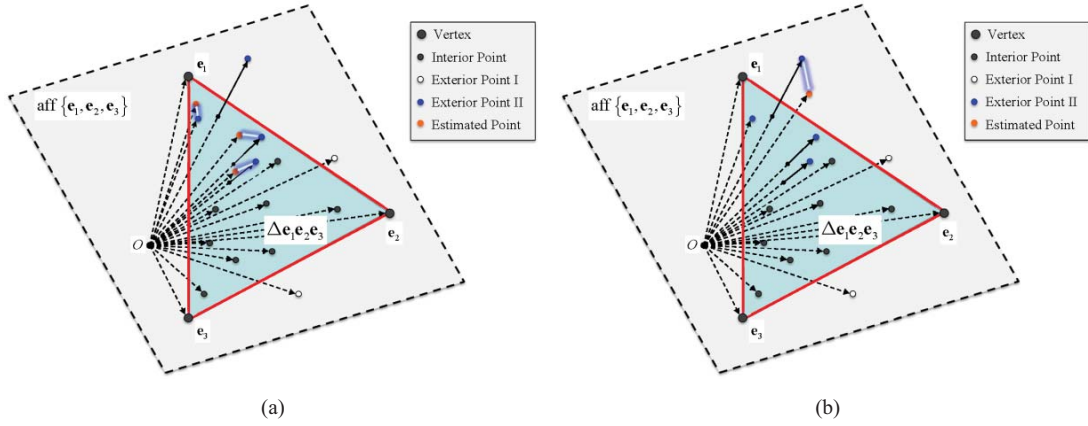


Fig. 2. Geometric illustration for DGCLE. (a) Case when estimated point is interior point. (b) Case when estimated point is exterior point I.

(17) and the computational load of computing the inverse of the matrix. Next, a technique based on eigendecomposition and Newton–Raphson method is employed to determine  $\varepsilon^p$ .

For positive-semidefinite Hermitian matrix  $\mathbf{A}$ , its spectral decomposition can be expressed as

$$\mathbf{A} = \sum_{j=1}^p \lambda_j \mathbf{u}_j \mathbf{u}_j^T = \mathbf{U} \mathbf{\Sigma} \mathbf{U}^T \quad (19)$$

where the columns of  $p$ -dimensional unitary matrix  $\mathbf{U}$  are the orthonormal eigenvectors  $\{\mathbf{u}_1, \mathbf{u}_2, \dots, \mathbf{u}_p\}$  of  $\mathbf{A}$ , and the diagonal elements of the diagonal matrix  $\mathbf{\Sigma}$ ,  $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_p \geq 0$ , are called the corresponding eigenvalues.

Therefore, (18) can be rephrased as

$$\varepsilon_p(\eta) = -\eta \left( \sum_{j=1}^p \frac{\mathbf{u}_j^T \mathbf{b}}{1 - \eta \lambda_j} \mathbf{u}_j \right). \quad (20)$$

It then follows (20) that the  $\eta$  can be obtained as the solution of the nonlinear equation:

$$f(\eta) = \eta \sum_{j=1}^p \left[ \frac{1}{(1 - \eta \lambda_j)^2} + \frac{1}{(1 - \eta \lambda_j)} \right] \|\mathbf{b}^T \mathbf{u}_j\|^2 - c = 0 \quad (21)$$

$f(\eta)$  is a monotonically increasing function when  $\eta \geq 0$ . Because  $f(0) = -c \leq 0$  and  $f(x) \rightarrow +\infty$  when  $\lambda \rightarrow 1/\lambda_1$ , there exists a unique solution  $\eta^* \in (0, 1/\lambda_1)$ . By combining the monotonicity of the function, we can obtain a tighter upper and lower bound on the solution  $\eta$  to (21)

$$\frac{1}{\lambda_1} \left( 1 - \frac{1}{\sqrt{1 + \lambda_1 \zeta}} \right) \leq \eta \leq \min \left( \frac{1}{2} \zeta, \frac{1}{\lambda_1} \right) \quad \text{and} \quad \eta \neq \frac{1}{\lambda_1} \quad (22)$$

with  $\zeta = c / \sum_{j=1}^p \|\mathbf{b}^T \mathbf{u}_j\|^2 \geq 0$ . The detailed proof of this conclusion can be found in Appendix B.

Hence, the solution for (21) can be efficiently determined by using the Newton's method in the above interval. The Newton sequence is

$$\eta_{k+1} = \eta_k - \frac{f(\eta_k)}{f'(\eta_k)} \quad (23)$$

where  $k$  is the iteration index. The solution  $\eta^*$  is used in (20) to obtain the  $\varepsilon^p$  and then the square distance vector  $\hat{\mathbf{d}}^p$  can be obtained from (10).

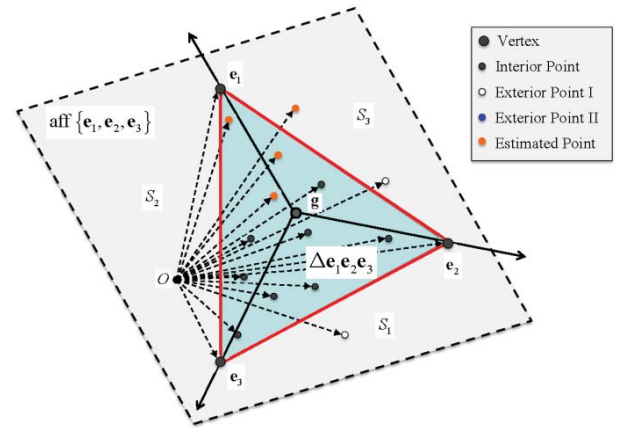


Fig. 3. Geometric illustration for affine hull partitioning.

It should be noted that an inappropriate initial value  $\eta_0$  for the Newton's method may make the sequence  $\{\eta_k\}$  behave erratically instead of approaching the actual root. Using the nonlocal convergence theorem for Newton's method [37, Th. 3.2], we can determine the initial value  $\eta_0 = \min(0.5\zeta, 1/\lambda_1 - \zeta)$ , where the  $\zeta$  is a predefined small real (the default is  $10^{-4}$ ).

To summarize, the geometric illustration for the distance geometry constraint-based localization estimation (DGCLE) algorithm is given in Fig. 2 and the detailed steps of the algorithm will be given in the Section III-D.

From the above analyses, we can see that the major computational burden of our approach comes from the eigendecomposition of the Hermitian matrix  $\mathbf{A}$  which requires  $O(p^3)$  floating operations (flops) and the iterative steps of Newton's method to find  $\eta$ , in which the initialization and Newton steps consume most of the work.

With the estimation algorithm above, we can get estimated location of observation pixel in the form of the square distance vector  $\hat{\mathbf{d}}^p$ , which is in the affine endmember-hull. Then, by means of (9), the barycentric coordinate of this estimated point can be obtained. If the barycentric coordinate satisfies the constraints in (2) as shown in Fig. 2(a), it will be considered as the abundances of observation pixel  $\mathbf{q}$ , or else the estimated point will fall into the class of the exterior point I for endmember-simplex just as shown in Fig. 2(b). To get the abundances of the exterior point I in this case, the next algorithm will



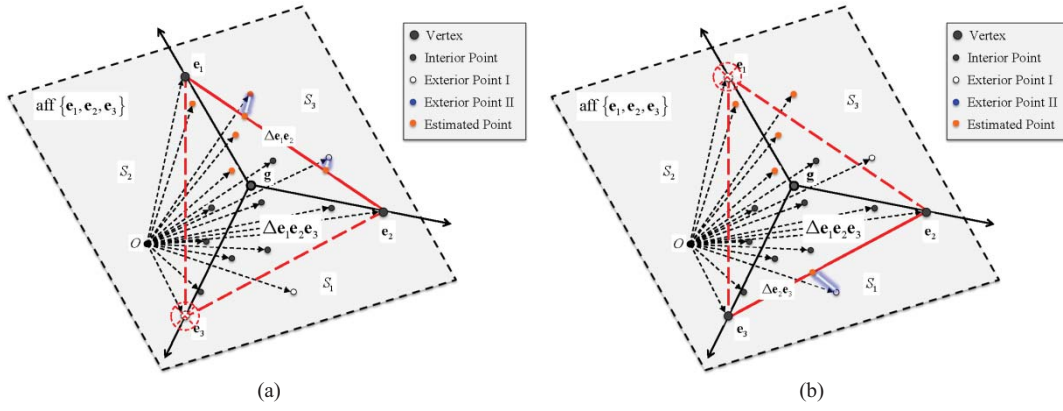


Fig. 4. Geometric illustration for new recursive process. (a) Case with exterior points I contained in  $S_3$ . (b) Case with exterior point I contained in  $S_1$ .

be introduced as subspace determination algorithm based on interior point.

We must point that in this subsection we described the algorithm for the single point. It is easy to apply the algorithm to the entire data set simultaneously, which decreases computational complexity significantly.

### C. Interior-Based Subspace Determination Algorithm

The optimal estimated points of exterior points II in the affine endmember-hull are obtained by DGCLE algorithm proposed in the last Subsection, and some of these estimated points are exterior points I for endmember-simplex. Estimating the fractional abundances of exterior points I (including the exterior points I for endmember-simplex and the exterior points I converted from the exterior points II by DGCLE algorithm) becomes the next obstacle.

We know that it is difficult to get the estimated points of exterior points I inside the endmember-simplex directly. However, with the fact that  $(p-1)$ -dimensional affine endmember-hull can be divided into  $p$  subspaces by a point inside the endmember-simplex, we can convert the problem to an optimal estimation problem for the endmember-subsimplex, whose vertices are contained inside the endmember-set.

**Definition 3 (Affine Hull Partitioning):** Let  $\mathbf{i}$  be the interior point of endmember-simplex  $\Delta \mathbf{e}_1 \mathbf{e}_2 \cdots \mathbf{e}_p$  and it is not inside the boundary of simplex, i.e.,  $\mathbf{i} \in \Delta \mathbf{e}_1 \mathbf{e}_2 \cdots \mathbf{e}_p$  and  $\mathbf{i} \notin \partial(\Delta \mathbf{e}_1 \mathbf{e}_2 \cdots \mathbf{e}_p)$ . Then the affine endmember-hull is partitioned into  $p$  isolated subspaces being split from  $\mathbf{i}$  toward the vertices of the simplex. The points in these subspaces can be defined as

$$\left\{ \mathbf{x} | \mathbf{x} \in S = \bigcup_i S_i \in \mathbf{R}^{p-1} \right\} \quad (24)$$

where

$$S_i = \left\{ \mathbf{x} | \mathbf{x} = \sum_{j=1, j \neq i}^p t_j \mathbf{e}_j + t_i \mathbf{i} \right\} \quad (25)$$

with  $0 \leq t_j \leq 1, j = 1, 2, \dots, i-1, i+1, \dots, p$  and  $\sum_{j=1}^p t_j = 1$ .

Equation (25) implies that when  $\mathbf{x} \in S_i$ ,  $p$ -dimensional column vector  $\mathbf{t} = (t_1, t_2, \dots, t_p)^T$  is the barycentric coordinate of  $\mathbf{x}$  with respect to the  $(p-1)$ -simplex  $\Delta \mathbf{e}_1 \mathbf{e}_2 \cdots \mathbf{e}_{i-1} \mathbf{e}_{i+1} \mathbf{e}_p$

with  $\mathbf{e}_1, \mathbf{e}_2, \dots, \mathbf{e}_{i-1}, \mathbf{i}, \mathbf{e}_{i+1}, \mathbf{e}_p$ , as base points. The geometric illustration for the partitioning is shown in Fig. 3.

**Assumption 1:** Let  $\Delta \mathbf{e}_1 \mathbf{e}_2 \cdots \mathbf{e}_p$  be the endmember-simplex and the exterior point  $\mathbf{q} \in S_i$  defined according to (25). Let  $\hat{\mathbf{q}}$  be the optimal estimated point of  $\mathbf{q}$  in the endmember-simplex. Then we assume that the barycentric coordinate  $s_i$  for the vertex  $\mathbf{e}_i$  at  $\hat{\mathbf{q}}$  holds

$$s_i = 0. \quad (26)$$

Assumption 1 means that when the exterior point  $\mathbf{q} \in S_i$ , the impact of  $\mathbf{e}_i$  on  $\mathbf{q}$  is neglectable, and the barycentric coordinate of the optimal estimated point  $\hat{\mathbf{q}}$  can be estimated from the endmember-subsimplex, which is obtained by removing the endmember  $\mathbf{e}_i$  from the endmember-set.

In this paper, we choose the barycenter  $\mathbf{g}$  of endmember-simplex, whose barycentric coordinate is  $(1, 1, \dots, 1)/p$ , as the interior to partition the affine endmember-hull. Hence, a fast and accurate method given in Theorem 3 is developed to judge whether the subspace contains  $\mathbf{q}$ .

**Theorem 3:** Let  $\mathbf{g}$  be the barycenter of endmember-simplex  $\Delta \mathbf{e}_1 \mathbf{e}_2 \cdots \mathbf{e}_p$ . Suppose that  $\mathbf{g}$  partitions the  $(p-1)$ -dimensional affine endmember-hull  $\text{aff}\{\mathbf{e}_1, \mathbf{e}_2, \dots, \mathbf{e}_p\}$  into  $p$  subspaces, i.e.,  $S_1, S_2, \dots, S_p$  defined as Definition 3, respectively. Then for the endmember-simplex, an exterior point  $\mathbf{q} \in S_i$  if and only if its barycentric coordinate  $\mathbf{s} = (s_1, s_2, \dots, s_i, \dots, s_p)^T$  satisfies

$$\forall_{i=1}^p : s_i \geq 0. \quad (27)$$

The detailed proof of Theorem 3 can be found in Appendix C.

The details of the approach will be given in the Section III-D. When the endmember-subsimplex is obtained, the exterior point  $\mathbf{q}$  for the endmember-simplex has become the exterior point II for the endmember-subsimplex as shown in Fig. 4. Then the DGCLE algorithm will be applied to the endmember-subsimplex, which starts a new recursive process.

From Assumption 1 and Theorem 3, it can be concluded that if the exterior point  $\mathbf{q} \in S_i$ , the coordinate  $s_i$  is minimal in the barycentric coordinate  $\mathbf{s} = (s_1, s_2, \dots, s_p)^T$  and its estimated abundance with respect to the endmember  $\mathbf{e}_i$  is zero, which is reasonable for the real application. In addition, the subspace determination algorithm shown in Theorem 3 only involves

---

**Algorithm 1** Pseudocode of DGAE Algorithm (Recursive Version)

---

**Input :** The  $L \times N$  observed data matrix  $\mathbf{X} = (\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N)$  with observation pixel  $\mathbf{x}_l$  ( $l = 1, 2, \dots, N$ )  
The  $L \times p$  endmember-matrix  $\mathbf{E} = (\mathbf{e}_1, \mathbf{e}_2, \dots, \mathbf{e}_p)$  with endmember-spectrum  $\mathbf{e}_l$  ( $l = 1, 2, \dots, p$ )  
The  $(p+1) \times N$  square distance matrix  $\mathbf{DI}$   
The inversion of Cayley–Menger matrix  $\mathbf{C}^{p+1}$  for the endmember-simplex, i.e.,  $(\mathbf{C}^{p+1})^{-1}$ .  
( $L$ ,  $N$ , and  $p$  are the number of bands, pixels, and endmembers, respectively).

**Output :** The  $p \times N$  abundances matrix  $\mathbf{S} = (\mathbf{s}_1, \mathbf{s}_2, \dots, \mathbf{s}_N)$ .

**Step 1:** If  $p=1$ , then  $\mathbf{S} = \mathbf{1}$  return.

If the number of inputs is 2

Calculate the Cayley–Menger matrix  $\mathbf{C}^{p+1}$  and its inversion  $(\mathbf{C}^{p+1})^{-1}$ .

Calculate the original distance matrix  $\mathbf{DI}$  with the  $l$ th ( $l = 1, 2, \dots, N$ ) column is

$$\begin{aligned} \left(1, (\mathbf{d}^p)^T\right)^T &= \left(1, \|\mathbf{x}_l - \mathbf{e}_1\|^2, \right. \\ &\quad \left. \|\mathbf{x}_l - \mathbf{e}_2\|^2, \dots, \|\mathbf{x}_l - \mathbf{e}_p\|^2\right)^T. \end{aligned}$$

end

**Step 2:** For every observation pixel  $\mathbf{x}_l$  ( $l = 1, 2, \dots, N$ ), carry out the DGAE algorithm and update the  $\mathbf{DI}$ .

2a) Compute the inverse of  $\mathbf{C}^{p+1}$  and calculate  $\mathbf{A}$ ,  $\mathbf{b}$ , and  $c$  using (14), respectively.

2b) Compute the eigendecomposition of  $\mathbf{A}$  [see (19)].

2c) If  $c = 0$ , then set  $\eta = 0$ , and go to the next observation pixel. If  $c > 0$ , then solve (21) for  $\eta$  by a Newton's method with the initial value  $\eta_0$ , using the knowledge that the solution is unique and it belongs to the interval in (22).

2d) Use the  $\eta$  obtained in Step 2c) to get  $\mathbf{e}^p$  by using (20). Then, obtain the square distance vector  $\mathbf{d}^p$ .

2e) Update the  $l$ th column of  $\mathbf{DI}$  with  $\left(1, (\mathbf{d}^p)^T\right)^T$ .

**Step 3:** Calculate the barycentric coordinates  $\mathbf{S} = (\mathbf{s}_1, \mathbf{s}_2, \dots, \mathbf{s}_N)$  for the endmember-simplex with (9).

**Step 4:** For every barycentric coordinate vector  $\mathbf{s}_l$  do if the minimal value of  $\mathbf{s}_l$ , i.e.,  $\min(\mathbf{s}_l) < 0$  Add  $\min(\mathbf{s}_l)$  and its index  $l$  to the column vector  $\mathbf{S}_{\min}$  and  $\mathbf{S}_{\text{index}}$ , respectively.

end

end

if  $\mathbf{S}_{\text{index}}$  is empty, return.

**Step 5:**  $\mathbf{S}((1, \dots, p), \mathbf{S}_{\text{index}}) = \mathbf{0}$ .

For  $i = 1, 2, \dots, p$  do

Add the index equal to  $i$  in the  $\mathbf{S}_{\text{index}}$  to the column vector  $\mathbf{S}_{\text{index\_i}}$ .

if  $\mathbf{S}_{\text{index\_i}}$  is nonempty, then

Calculate the input matrix  $\left((\mathbf{C}^{p+1})_{(i+1)}^{\vee}\right)^{-1}$  from the  $(\mathbf{C}^{p+1})^{-1}$ .

Remove the  $i$ th row of  $\mathbf{DI}$  to get  $\mathbf{DI}_{\vee_i}$ .

Execute the recursive process with

$$\mathbf{S}([1, \dots, i-1, i+1, \dots, p], \mathbf{S}_{\text{index\_i}}) = \text{DGAE}\left(\sim, \sim, \mathbf{DI}_{\vee_i}, \left((\mathbf{C}^{p+1})_{(i+1)}^{\vee}\right)^{-1}\right).$$

end

end

---

some scalar comparisons and index managements with low computational complexity.

#### D. Algorithm Summarization

Here, we summarize the algorithm below, named the distance geometry-based abundance estimation (DGAE) algorithm, and show the detailed pseudocode step-by-step in Algorithm I.

In the above pseudocode, the symbol  $(\mathbf{C}^{p+1})_{(i+1)}^{\vee}$  is the remaining submatrix after the  $(i+1)$ th column and the  $(i+1)$ th row of square matrix  $\mathbf{C}^{p+1}$  are removed. The symbol  $\mathbf{DI}_{\vee_i}$  is the remaining submatrix with the  $i$ th row of matrix  $\mathbf{DI}$  removed. The symbol  $\sim$  means that the input can be neglected.

The pseudocode is a recursive version for the DGAE algorithm. Among the four inputs,  $\mathbf{X}$  and  $\mathbf{E}$  are necessary for the first recursion, in which the  $\mathbf{DI}$  and  $(\mathbf{C}^{p+1})^{-1}$  are computed, whereas in the following recursions, the first two input can be neglected and the  $\mathbf{DI}$  and  $(\mathbf{C}^{p+1})^{-1}$  are updated in the last recursion.

To decrease the computational complexity, we use the extended version of Hermitian Matrix Inversion Theorem to get the  $\left((\mathbf{C}^{p+1})_{(i+1)}^{\vee}\right)^{-1}$  in Step 5 of Algorithm I. Hence, the calculation of  $\left((\mathbf{C}^{p+1})_{(i+1)}^{\vee}\right)^{-1}$  can be replaced by the multiplication operation. The detailed proof will be given in Appendix D.

#### E. Computational Complexity

Here, we give the computational complexity of the proposed algorithm, which is measured by the number of flops. The total complexity of the algorithm depends strongly on the implementation details and optimizations used and it is hard to determine explicitly. So we only analyze the computational complexity of each step in the first single recursion of the algorithm.

The computational operations involved in the algorithm can be categorized into one-time operations and recursive operations. The one-time operations are only executed once in the whole algorithm and their computational complexity can be neglected when the number of recursive calls is large. The major computational demand of the proposed algorithm comes from the recursive operations.



The computational costs of different steps within the first single recursion are listed as follows.

- 1) The initialization process is achieved in Step 1 of Algorithm I, which only involves the one-time operations. In this step, a  $(p+1)$ -dimensional Cayley–Menger matrix  $\mathbf{C}^{p+1}$  and its inversion  $(\mathbf{C}^{p+1})^{-1}$  will be calculated. Meantime, a  $(p+1) \times N$  matrix  $\mathbf{DI}$  is obtained, involving  $NpL + pN + pL$  multiplication operations.
- 2) The DGCLE algorithm is executed in Step 2 of Algorithm I. As described in Subsection III-B, the major computational demand of the algorithm comes from two sources. One is the eigendecomposition of  $p$ -dimensional symmetric and semipositive definite matrix, which requires  $O(p^3)$  flops. The other is that the algorithm needs to search for  $\eta$  via a Newton's method, which requires  $O(p^2N)$  flops.
- 3) The barycentric coordinates are calculated in Step 3 of Algorithm I. A matrix multiplication of a  $p \times p$  matrix with a  $p \times N$  matrix is involved in this step. It should be noted that from the second recursion process on, the inversion  $\left((\mathbf{C}^{p+1})_{(i+1)}^{\vee}\right)^{-1}$  in (9) is calculated beforehand from the last recursion process, which requires  $p^2$  multiplication as shown in Step 5 of Algorithm I.
- 4) All other steps only involve index management or scalar comparison, which have a negligible computational complexity.

We can know that the computational complexity involved the number of band  $L$  comes from the Step 1, in which a  $(p+1) \times N$  distance matrix  $\mathbf{DI}$  is calculated. Therefore, there is only a slight increase of the computational complexity when the number of bands increases. As for the following recursive operations, the computational complexity is dependent on  $p^2N$ . With the recursive process going on, both  $p$  and  $N$  decrease, and so does the computational complexity.

In the end, we have a comparison of computational complexity between SPU and DGAE. For the SPU algorithm, the main recursive computational cost comes from simplex plane projection, determining the location of the incenter and the vector  $\mathbf{b}^i$  for a given cone  $Z_i$ . For ease of representation we still use the symbols in [30]. A  $p$ -dimensional matrix inversion and multiplication of a  $p \times p$  matrix and matrix multiplication of a  $pL$  matrix with a  $LN$  matrix are involved in the simplex plane projection. To obtain the incenter, the  $p$  determinants of  $(p-1)$ -dimensional matrix are calculated. Simultaneously, the computation operations including the determining the vector  $\mathbf{b}^i$ ,  $(p-1)$ -dimensional matrix inversion, and matrix multiplication of a  $(p-1)$ -dimensional matrix with a  $(p-1) \times N$  matrix have to be executed  $p$  times.

Based on the above analysis, we conclude that the computational complexity of SPU depends on the  $L$  and  $p^3N$ , and therefore, our algorithm always has a lower computational complexity than the SPU algorithm.

#### IV. EXPERIMENTAL RESULTS

Here, we utilize the synthetic and real hyperspectral data to test the performance of the proposed DGAE algorithm and

compare it with the other three state-of-the-art algorithms, i.e., FCLS [12], C-SUnSAL [24], and SPU [30]. The FCLS is a famous iterative algorithm for LSU, which considers the unmixing task as a least-squares regression with both ASC and ANC. The C-SUnSAL algorithm employs the ADMM algorithm to solve the hyperspectral unmixing problem. The SPU algorithm is based upon the equivalence between the FCLS algorithm and the simplex-projection algorithm, and is also a recursive algorithm. The MATLAB codes of the FCLS and SPU algorithms are obtained from the authors, whereas the MATLAB implementation of C-SUnSAL is available online.<sup>1</sup>

The C-SUnSAL algorithm can be used to solve the constrained basis pursuit denoising as follows:

$$\min_{\mathbf{s}} \|\mathbf{s}\|_1 \quad \text{s.t.} \quad \|\mathbf{E}\mathbf{s} - \mathbf{x}\|_2 \leq \delta, \quad \mathbf{s} \geq 0 \quad (28)$$

where  $\|\mathbf{s}\|_1$  is the  $L_1$  norm of  $\mathbf{s}$ ,  $\delta$  is the tolerated reconstruction error. As in [24], the ANC and ASC can be enforced or deactivated trivially. In this paper, the parameters of C-SUnSAL are set as:  $\delta = 0$ , the maximum number of iterations AL\_ITEERS is 400, the ASC and the ANC are enforced, and the tolerance for the primal and dual residuals TOL is  $10^{-4}$  [24]. The algorithm will run until the stopping criteria is met:  $\text{TOL} < 1e-4$  or  $\text{AL\_ITEERS} > 400$ . With the parameters above, the C-SUnSAL algorithm is employed to compute a faster FCLS solution rather than the primary sparse one as in [24].

If the standard abundances are known, as in the case with the synthetic images, we can evaluate algorithms by root mean square error (RMSE) between the estimated fractional abundances and true fractional abundances, which is widely used as the criterion for abundance estimation [38].

Considering the true abundance matrix  $\mathbf{S} = (\mathbf{s}_1, \mathbf{s}_2, \dots, \mathbf{s}_N) \in \mathbf{R}^{p \times N}$  and its estimated abundance matrix  $\hat{\mathbf{S}} = (\hat{\mathbf{s}}_1, \hat{\mathbf{s}}_2, \dots, \hat{\mathbf{s}}_N) \in \mathbf{R}^{p \times N}$ , the RMSE is defined as [38]

$$\text{RMSE} = \frac{1}{p} \sum_{k=1}^p \left[ \sqrt{\frac{1}{N} \sum_{j=1}^N (\hat{\mathbf{s}}_{kj} - \mathbf{s}_{kj})^2} \right]. \quad (29)$$

A small RMSE as possible is wished for a good unmixing accuracy.

In real hyperspectral images, as we do not know the true abundance matrix, the RMSE is impractical. So another criterion, named reconstruction error (RE), is classically used to evaluate the quality of the unmixing methods. It is defined as [39]

$$\text{RE} = \frac{1}{L} \sum_{k=1}^L \left[ \sqrt{\frac{1}{N} \sum_{j=1}^N (\hat{\mathbf{x}}_{kj} - \mathbf{x}_{kj})^2} \right] \quad (30)$$

where  $\mathbf{X} = (\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N) \in \mathbf{R}^{L \times N}$  and  $\hat{\mathbf{X}} = (\hat{\mathbf{x}}_1, \hat{\mathbf{x}}_2, \dots, \hat{\mathbf{x}}_N) \in \mathbf{R}^{L \times N}$  are original data matrix and data matrix reconstructed from the estimated abundance matrix, respectively. Of course, a small RE is the expected result. However, in some cases the least RE does not mean the best unmixing accuracy especially when the overfitting happens [22].

<sup>1</sup>Available at [http://www.lx.it.pt/~bioucas/code/sunsal\\_demo.zip](http://www.lx.it.pt/~bioucas/code/sunsal_demo.zip).

TABLE I  
RESULTS OF ALGORITHMS WITH DIFFERENT NOISE LEVELS

SNR	RMSE				Runtime (s)				Recursion Numbers of DGAE
	FCLS	C-SUnSAL	SPU	DGAE	FCLS	C-SUnSAL	SPU	DGAE	
$\infty$	0.0000	0.0000	0.0006	<b>0.0000</b>	12.7784	0.3982	1.3487	<b>0.1875</b>	38
50	0.0033	0.0033	0.0036	<b>0.0028</b>	15.0537	1.1690	2.0331	<b>0.2740</b>	66
45	0.0061	0.0061	0.0062	<b>0.0047</b>	15.4696	1.4225	2.1633	<b>0.2786</b>	67
40	0.0116	0.0116	0.0118	<b>0.0078</b>	16.0150	1.6600	2.3117	<b>0.2840</b>	71
35	0.0225	0.0225	0.0225	<b>0.0124</b>	16.5521	2.1280	2.5226	<b>0.3016</b>	79
30	0.0416	0.0417	0.0417	<b>0.0192</b>	16.8237	2.5384	2.6341	<b>0.3285</b>	87
25	0.0710	0.0710	0.0710	<b>0.0325</b>	16.8218	2.9429	2.6908	<b>0.5393</b>	95
20	0.1074	0.1074	0.1074	<b>0.0645</b>	16.1388	3.1043	2.5573	<b>0.5855</b>	105
15	0.1463	0.1464	0.1463	<b>0.1218</b>	15.1765	3.0908	2.2758	<b>0.6812</b>	107

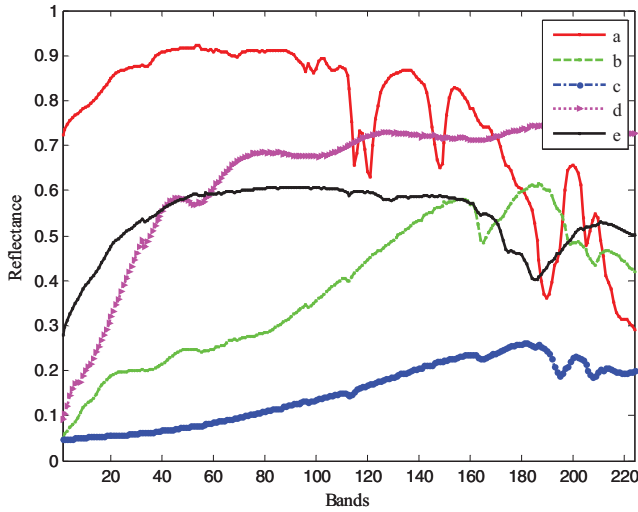


Fig. 5. Endmember-spectra selected from USGS library. (a) Alunite GDS83 Na63. (b) Nontronite GDS41. (c) Desert Varnish GDS78A Rhy. (d) Pyrope WS474. (e) Buddingtonite NHB2301.

The running time of algorithm can be used as a criterion to evaluate the computational complexity of algorithm quantitatively. In the following experiments we measure the runtimes of the algorithms implemented in MATLAB 7.13. These tests are done on a regular desktop machine with an Intel(R) Xeon(R) CPU E5504 @ 2.00 GHz (dual core) and 24 GB of RAM. It should be noted that computational complexity alone does not necessarily determine the shortest time to a solution and so the runtime is just a quantitative criterion [40].

#### A. Results on Synthetic Hyperspectral Images

The synthetic data sets used for the algorithm comparison are created as linear mixtures of a set of spectra with synthesized abundance maps. The endmember-spectra are selected from the mineral spectra in United States Geological Survey (USGS) digital spectral library,<sup>2</sup> which consists of 420 spectra with 224 bands. Fig. 5 shows the spectra of five endmembers.

<sup>2</sup>Available at <http://speclab.cr.usgs.gov/spectral-lib.html>.

In addition, to investigate behaviors of the unmixing algorithms for the different spectral libraries, we consider other two spectral libraries: the ASTER Library<sup>3</sup> and the library made of independent and identically distributed (i.i.d.) components.

The corresponding abundance maps are generated following a Dirichlet distribution in the same way of [38] with  $\mu = (\mu_1, \mu_1, \dots, \mu_p)^T$  and  $\mu_1 = \mu_1 = \dots = \mu_p = 1/p$ , whose probability density function is given by

$$P(\mathbf{s}, \mu) = \frac{\Gamma\left(\sum_{i=1}^P \mu_i\right)}{\prod_{i=1}^P \Gamma(\mu_i)} \prod_{i=1}^P s_i^{\mu_i-1} \quad (31)$$

where  $\mathbf{s} = (s_1, s_2, \dots, s_p)^T$ ,  $s_i \geq 0$ ,  $\sum_{i=1}^P s_i = 1$ , and  $\Gamma(\bullet)$  are the Gamma function. The expected value of the  $i$ th fraction  $s_i$  is  $E(s_i) = \mu_i / \sum_{i=1}^P \mu_i$ . The Dirichlet distribution of the abundances automatically satisfies the two constraints in (2). All the fractional abundances can be arranged as matrix  $\mathbf{S}$ , and each row of  $\mathbf{S}$  corresponds to one abundance map. Then the synthetic data  $\mathbf{X}$  is produced by multiplying the endmember-matrix and abundances, i.e.,  $\mathbf{X} = \mathbf{E}\mathbf{S}$ .

To simulate the possible errors and sensor noise, zero mean Gaussian random noise measured by signal-to-noise ratio SNR is added to the synthetic scenes, and the SNR is defined as

$$\text{SNR} = 10 \log_{10} \frac{E(\mathbf{x}^T \mathbf{x})}{E(\mathbf{n}^T \mathbf{n})} \quad (32)$$

where  $\mathbf{x}$  and  $\mathbf{n}$  are defined as in (1) and  $E(\bullet)$  is the expectation operator.

Six experiments are designed to evaluate the four algorithms. In the first experiment, we study the robustness to noise corruptions of all four algorithms. In the second experiment, the size of synthetic data is changed to illustrate the impact of data quantity on the algorithm performance. The performance on the dimensionality of the data will be investigated in the third experiment. In the fourth experiment, we study the impact of the number of endmembers on the algorithm performance. The other two different libraries are applied to

<sup>3</sup>Available at <http://speclib.jpl.nasa.gov>.

TABLE II  
RESULTS OF ALGORITHMS WITH DIFFERENT NUMBER OF PIXELS

Size of Images	RMSE				Runtime (s)				Recursion Numbers of DGAE
	FCLS	C-SUnSAL	SPU	DGAE	FCLS	C-SUnSAL	SPU	DGAE	
100 × 100	0.0104	0.0104	0.0112	<b>0.0067</b>	2.2679	0.1910	0.4565	<b>0.0798</b>	60
200 × 200	0.0114	0.0114	0.0116	<b>0.0076</b>	9.0313	1.0598	1.4985	<b>0.2049</b>	69
300 × 300	0.0118	0.0118	0.0119	<b>0.0079</b>	20.0688	2.3658	3.2285	<b>0.3662</b>	73
400 × 400	0.0121	0.0122	0.0122	<b>0.0082</b>	35.8515	4.2592	5.6600	<b>0.6073</b>	76
500 × 500	0.0125	0.0125	0.0125	<b>0.0086</b>	56.0783	6.6485	8.7244	<b>0.9196</b>	77
600 × 600	0.0127	0.0127	0.0127	<b>0.0088</b>	80.9179	9.3646	12.4027	<b>1.3064</b>	78
700 × 700	0.0127	0.0128	0.0127	<b>0.0088</b>	110.8109	12.6770	16.9371	<b>1.7628</b>	79
800 × 800	0.0129	0.0129	0.0129	<b>0.0090</b>	145.0150	16.3516	22.0353	<b>2.2823</b>	82
900 × 900	0.0131	0.0131	0.0132	<b>0.0092</b>	182.8889	20.6045	28.0213	<b>2.8545</b>	82
1000 × 1000	0.0132	0.0132	0.0132	<b>0.0092</b>	225.0245	25.1307	34.0036	<b>3.5321</b>	84

TABLE III  
RESULTS OF ALGORITHMS WITH DIFFERENT NUMBER OF BANDS

Number of Bands	RMSE				Runtime (s)				Recursion Numbers of DGAE
	FCLS	C-SUnSAL	SPU	DGAE	FCLS	C-SUnSAL	SPU	DGAE	
20	0.0294	0.0294	0.0295	<b>0.0269</b>	15.0286	1.2639	0.5074	<b>0.1998</b>	77
40	0.0218	0.0218	0.0218	<b>0.0188</b>	14.8066	1.2866	0.7189	<b>0.2196</b>	73
60	0.0189	0.0190	0.0190	<b>0.0157</b>	14.7881	1.4413	0.9154	<b>0.2231</b>	74
80	0.0166	0.0166	0.0167	<b>0.0133</b>	14.7491	1.4668	1.0829	<b>0.2147</b>	72
100	0.0154	0.0154	0.0155	<b>0.0119</b>	14.7586	1.5389	1.2708	<b>0.2448</b>	74
120	0.0146	0.0146	0.0147	<b>0.0110</b>	14.9025	1.5343	1.4591	<b>0.2550</b>	72
140	0.0136	0.0136	0.0137	<b>0.0099</b>	14.6698	1.5377	1.6284	<b>0.2513</b>	70
160	0.0130	0.0130	0.0131	<b>0.0093</b>	14.7977	1.6304	1.8314	<b>0.2530</b>	72
180	0.0126	0.0126	0.0127	<b>0.0088</b>	14.8085	1.7593	2.1296	<b>0.2685</b>	71
200	0.0120	0.0120	0.0121	<b>0.0082</b>	15.0326	1.7115	2.1458	<b>0.2877</b>	72
220	0.0117	0.0117	0.0118	<b>0.0079</b>	14.8482	1.7532	2.4039	<b>0.3070</b>	72

the four algorithms in the fifth experiment to compare their performances. In the final experiment, we will consider the robustness of the algorithms to the under- or over-estimation of the number of endmembers.

In all experiments, we use two criteria, RMSE and runtime, to benchmark the performances. Moreover, the number of recursions in the DGAE algorithm will be given to analyze the results. To avoid the error from single run, these two criteria are approximated by sample means based on 40 Monte Carlo runs. Moreover, except the case in the sixth experiment, we assume the number of endmembers to be known as *a priori* for all algorithms and the endmember signatures are extracted by the famous EEA N-FINDR proposed by Winter in [43]. As a simplex-based algorithm, N-FINDR aims to search the maximum volume simplex spanned by the observation pixels and extracts the vertices as the endmembers.

1) *Experiment 1 (Simulations for Various SNRs)*: In this experiment, each synthetic image has  $256 \times 256$  pixels and five endmember signatures (shown in Fig. 5), with the number of bands 224. To test anti-noise capability of these algorithms, we change the SNR from  $\infty$  (noiseless), 50 to 15 dB.

Table I shows the RMSEs and runtimes as a function of the SNR. It can be seen that when  $\text{SNR} = \infty$ , all the four algorithms have perfect unmixing results. With the decrease of SNR, they perform worse, i.e., the RMSEs increase. Compared with the FCLS and SPU algorithms, the DGAE algorithm gives the best unmixing accuracy measured by RMSE, whereas the results obtained from the other two algorithms are comparable in quality. The runtimes have slight fluctuations for the four algorithms. As can be expected, the runtimes of the DGAE are less than those of the other algorithms for all values of SNR and have a slight increase along with the increase of number of recursions in the algorithm as shown in the rightmost column of Table I. Nonetheless, one can find that the runtime of the DGAE algorithm does not scale linearly with the number of recursions strictly. This is because the recursions depend on both the inherent property (e.g., the SNR, the number of pixels, endmembers, bands, and so on) and geometric structure of the data set. Moreover, the runtime of each recursion varies because of the different implementation details. The C-SUnSAL algorithm, which has a FCLS solution with the parameters adopted, always runs faster than the FCLS algorithm.

TABLE IV  
RES OF ALGORITHMS WITH DIFFERENT NUMBERS OF ENDMEMBERS

Number of Endmembers	RMSE				Runtime (s)				Recursion Numbers of DGAE
	FCLS	C-SUnSAL	SPU	DGAE	FCLS	C-SUnSAL	SPU	DGAE	
4	0.0100	0.0100	0.0101	<b>0.0071</b>	11.2628	1.5331	1.2627	<b>0.2266</b>	19
6	0.0146	0.0147	0.0148	<b>0.0091</b>	16.0169	2.3729	3.5299	<b>0.3947</b>	258
8	0.0142	0.0142	0.0144	<b>0.0086</b>	19.9013	3.0460	8.4513	<b>1.4069</b>	2735
10	0.0175	0.0175	0.0178	<b>0.0099</b>	24.5125	<b>4.3303</b>	21.7121	7.1153	19243
12	0.0187	0.0187	0.0190	<b>0.0104</b>	28.6626	<b>5.2336</b>	57.2817	25.4843	72558
14	0.0194	0.0194	0.0197	<b>0.0110</b>	33.0304	<b>6.7019</b>	141.7823	60.9539	167517
16	0.0211	0.0211	0.0214	<b>0.0122</b>	37.6896	<b>8.2372</b>	298.6524	106.6562	277956
18	0.0209	0.0209	0.0213	<b>0.0128</b>	42.4887	<b>9.3719</b>	538.2889	159.5010	393544
20	0.0218	0.0218	0.0222	<b>0.0135</b>	47.3635	<b>10.3424</b>	850.3764	219.4502	510708

2) *Experiment 2 (Simulations for Various Numbers of Pixels)*: In this experiment, we compare the performances of different algorithms for various numbers of pixels. The image size of the synthetic data varies from  $100 \times 100$ ,  $200 \times 200$  to  $1000 \times 1000$ . The SNR, the number of endmembers, and bands are set as 40 dB, 5, and 224, respectively.

The RMSEs and time consumptions of all the algorithms along with the increase of image sizes are given in Table II. The performances of all four algorithms including RMSEs and runtimes increase with the increasing number of pixels as expected. Similarly, the FCLS, C-SUnSAL and SPU algorithms have comparable unmixing accuracy and the DGAE algorithm performs significantly better than the FCLS solutions. The runtimes of the DGAE algorithm are, again, much lower than those of other three algorithms and the growth of runtime is the slowest with the increase of the number of pixels. The numbers of recursions in the DGAE algorithm are given in the rightmost column of Table II, from which we can get the similar conclusions with that of the experiment 1.

3) *Experiment 3 (Simulations for Various Numbers of Bands)*: The performances of algorithms on the dimensionality  $L$  are investigated in this experiment. We set the endmember-number, the size of images and the SNR as 5,  $256 \times 256$  and 40 dB, respectively. The number of bands  $L$  changes from 20 to 220 with the step of 20. We get the endmember signatures with different number of bands by resampling randomly from the original 224-band spectra. For example, if we want to get an  $L$ -band spectrum we resample randomly  $L$  values from the original spectrum to reconstruct it.

Table III shows the changes of RMSEs and runtimes of all the algorithms along with the increase of  $L$ . It can be seen that the unmixing accuracy is improved with the increase of dimensionality expressed in the decrease of RMSEs. And the DGAE algorithm has the better RMSEs than other methods. As expected, the runtimes of DGAE algorithm have a lower slope to the dimensionality, whereas the analysis in Section III-E, the computational time of SPU algorithm increases remarkably with the increase of the number of bands  $L$ . Over all, for the synthetic data sets with different dimensionalities, the DGAE algorithm has better unmixing accuracy and less time consumption and its runtimes only have a slight increase with the increase of the dimensionalities.

TABLE V  
RES OF ALGORITHMS WITH DIFFERENT NUMBERS OF ENDMEMBERS

Number of Endmembers	RE ( $\times 10^{-1}$ )			
	FCLS	C-SUnSAL	SPU	DGAE
4	0.0705	0.0705	0.0719	0.0706
6	0.0696	0.0696	0.0761	0.0698
8	0.0711	0.0711	0.0843	0.0714
10	0.0698	0.0699	0.0896	0.0703
12	0.0711	0.0712	0.0976	0.0717
14	0.0700	0.0703	0.1019	0.0708
16	0.0694	0.0697	0.1081	0.0704
18	0.0698	0.0703	0.1133	0.0709
20	0.0693	0.0698	0.1202	0.0705

4) *Experiment 4 (Simulations for Various Numbers of Endmembers)*: In this experiment, we change the endmember-number  $p$  from 4 to 20 with the step of 2 and the rest parameters of the synthetic images are set as follows: SNR = 40 dB, the image size is  $256 \times 256$ , and the number of bands is 224, respectively. We select the spectra in the following way: We first create a spectral library consisted of 200 spectra selected from the USGS spectral library. Then we choose  $p$  spectra randomly from this library to generate the synthetic data. The benefit of the way to create the data is that the impact of artificial selection for spectra can be avoided.

Table IV shows the RMSEs and time consumptions of all the algorithms as a function of  $p$ . We can notice that the RMSEs of the four algorithms increase (denoting that the results become worse) with the increase of the number of endmembers, as the higher the number of the endmembers is, the more complicated the structure of simplex formed by the endmembers is. Because of explicitly taking into account the geometric structure of data set by use of the pairwise distances between the observation pixels and the endmembers, the DGAE algorithm has the best unmixing accuracy, whereas the other three that compute the FCLS solutions, have almost the same unmixing accuracy.

The required processing times increase exponentially as a function of  $p$  for all algorithms. When the  $p$  is small, e.g.,

TABLE VI  
RES OF ALGORITHMS WITH DIFFERENT NUMBERS OF ENDMEMBERS

Number of Endmembers	RMSE (ASTER Library)				RMSE (Uniform Library)( $\times 10^{-1}$ )			
	FCLS	C-SUnSAL	SPU	DGAE	FCLS	C-SUnSAL	SPU	DGAE
4	0.0063	0.0063	0.0064	<b>0.0049</b>	0.0169	0.0169	0.0202	<b>0.0160</b>
6	0.0104	0.0103	0.0106	<b>0.0067</b>	0.0157	0.0157	0.0226	<b>0.0150</b>
8	0.0132	0.0132	0.0137	<b>0.0089</b>	0.0143	0.0143	0.0253	<b>0.0137</b>
10	0.0137	0.0136	0.0144	<b>0.0094</b>	0.0136	0.0136	0.0282	<b>0.0131</b>
12	0.0169	0.0167	0.0175	<b>0.0111</b>	0.0129	0.0129	0.0296	<b>0.0125</b>
14	0.0189	0.0186	0.0196	<b>0.0129</b>	0.0125	0.0125	0.0318	<b>0.0121</b>
16	0.0185	0.0183	0.0192	<b>0.0134</b>	0.0120	0.0120	0.0336	<b>0.0116</b>
18	0.0199	0.0192	0.0206	<b>0.0153</b>	0.0115	0.0116	0.0345	<b>0.0111</b>
20	0.0219	0.0210	0.0226	<b>0.0162</b>	0.0112	0.0113	0.0355	<b>0.0109</b>

TABLE VII  
RESULTS MEASURED BY RMSE AS FUNCTIONS OF NUMBER OF ENDMEMBERS UNDER-OR OVER-ESTIMATED BY ONE

Number of Endmembers		RMSE			
		FCLS	C-SUnSAL	SPU	DGAE
Under-estimated by one	5 for 6	0.0874	0.0874	0.0874	<b>0.0844</b>
	6 for 7	0.0792	0.0792	0.0792	<b>0.0681</b>
	7 for 8	0.0671	0.0671	0.0671	<b>0.0569</b>
	8 for 9	0.0666	0.0666	0.0667	<b>0.0529</b>
	9 for 10	0.0615	0.0616	0.0616	<b>0.0472</b>
Over-estimated by one	7 for 6	0.0566	0.0567	0.0567	<b>0.0492</b>
	8 for 7	0.0647	0.0647	0.0648	<b>0.0514</b>
	9 for 8	0.0592	0.0593	0.0593	<b>0.0454</b>
	10 for 9	0.0597	0.0598	0.0598	<b>0.0452</b>
	11 for 10	0.0582	0.0582	0.0583	<b>0.0409</b>

$p < 10$ , the DGAE algorithm is the fastest among the four algorithms. However, with the increase of  $p$ , the runtime of DGAE algorithm rises rapidly, as indicated by the rapid rise of the number of recursions. As said above, the computational complexity of the recursive operations in the DGAE algorithm is dependent on  $p^2N$ . So at this moment, the impact of  $p$  on the runtime has exceeded all the other factors. Even when  $p \geq 10$ , its computational time is more than the C-SUnSAL algorithm but is still below the SPU algorithm. We can notice that for very high values of  $p$ , the C-SUnSAL algorithm has a lower computational complexity than the other three algorithms. The runtime of SPU algorithm increases most dramatically among the four algorithms.

For comparing the results between the synthetic and real hyperspectral data (that will be discussed in Section IV-B) for various numbers of endmembers, we give the REs of all the algorithms as a function of  $p$  in Table V.

The C-SUnSAL algorithm has almost the same REs with the FCLS algorithm, whereas the SPU algorithm has an opposite result that the RE increases along with the increase of  $p$ . The increase of RE for the SPU algorithm might result from the slightly flawed method used in the algorithm to estimate endmembers with zero abundance, which sometimes leads to

the incorrect unmixing results [41]. It is clear that the RE of the DGAE algorithm decreases with the increase of  $p$  and is very close to that of the FCLS algorithm. Although the RE of DGAE is slightly higher compared with the FCLS algorithm, the RMSE of DGAE is better because the overfitting is weakened.

5) *Experiment 5 (Simulations for Different Spectral Libraries)*: Here, the other two libraries are used. One is ASTER library Version 2.0 released on December 3rd, 2008, a compilation of over 2400 spectra of natural and man-made materials [42]. A selection of 200 spectra will be used in the simulations. The other is generated from the i.i.d. components uniformly distributed in the interval  $[0, 1]$ . Also, it contains 200 spectra.

We apply these two libraries to the synthetic datasets with the number of endmembers changing from 4 to 20 just as the case in Experiment 4. Table VI shows the results of the experiment, denoting the similar results compared with the USGS library.

6) *Experiment 6 (Simulations for the Robustness to the Under- or Over-Estimation of  $p$ )*: Here, we consider the situation that the number of endmembers in the images is under- or over-estimated, i.e., the number of endmembers is not known or correctly evaluated, which is often a crucial problem in hyperspectral unmixing [44].

When the  $p$  is underestimated, we can identify the true abundance maps and the RMSEs can be computed. However, when the  $p$  is overestimated, because of the lack of the true abundance maps corresponding to the derived abundance maps, we only compute the RMSEs for the right abundance maps.

As before, the results are given in Table VII, in which  $m$  for  $n$  means that the true  $p$  is  $n$  whereas the  $p$  used in unmixing is  $m$ . From the Table VII, we can see that the results of the DGAE algorithm are more accurate than those of the other three algorithms.

Based on the above simulated experiments and the theoretical analysis in Section III, it is worthwhile to highlight several notable characters of the proposed algorithm here.

- 1) A new calculation formula for barycentric coordinates. Although a number of formulas are proposed to cal-

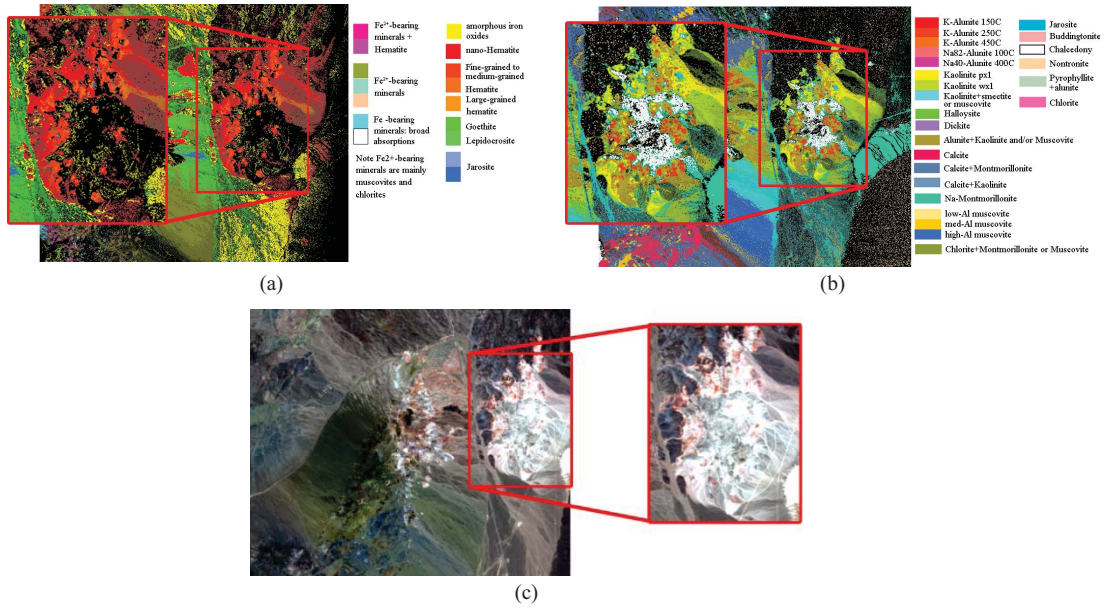


Fig. 6. Two USGS ground-truth maps of Cuprite data set and true-color image (RGB channels are bands 40, 20, 10). (a) USGS Tetracorder map for electronic absorption features of minerals and zoomed-in region of interest. (b) USGS Tetracorder map for vibrational absorption features of minerals and zoomed-in region of interest. (c) True-color image and zoomed-in region of interest.

TABLE VIII  
NUMBER OF ENDMEMBERS DETERMINED BY VD AND  
HYSIME FOR CUPRITE DATA SET

PF	VD			HySime
	HFC	NWHFC	NSP	
1.00E-01	28	19	20	18
1.00E-02	23	16	20	
1.00E-03	22	16	20	
1.00E-04	18	15	20	
1.00E-05	18	15	20	
1.00E-06	15	14	20	
1.00E-07	12	12	20	

culate the barycentric coordinate with respect to the simplex [31]. Few of them are fit for the data set in high-dimensional space, e.g., in the  $L$ -dimensional band space when the dimension of simplex  $p$  is much less than  $L$ . Generally, several feature extraction techniques are performed prior to bring the information in the high-dimensional space to the right subspace. However, the loss of some useful information follows inevitably. The CMBCC algorithm that we proposed in Section III-A overcomes this disadvantage by utilizing the distances between the observation pixel and endmembers in the form of Cayley–Menger matrix. In addition, we should point out that it is easy to apply this formula to other fields, e.g., computer vision.

- 2) A novel estimation algorithm based on the distance geometry constraint. The distance geometry theory is applied in several fields, such as localization algorithm for wireless sensor network, computer vision and macro-molecule structure, and so on, to deal with the geometric

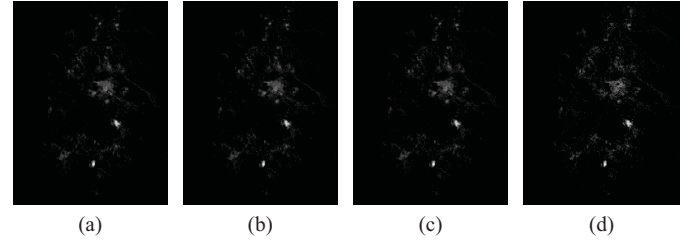


Fig. 7. Estimated abundance maps of Alunite for Cuprite data set. (a) FCLS. (b) C-SUNSAL. (c) SPU. (d) DGAE.

structure of the object. In this paper, we consider the geometric structure by the distance geometry constraint that describes the geometric relationship in the form of Euclidean distances. From the least squares problem in (15), in the whole unmixing procedure, the geometric structure of hyperspectral data set can be preserved, which is indicated by the least deformation on the Euclidean distance matrix. As a result, the proposed algorithm considers not only the relationship between observation pixels and hyperplane like SPU but also the geometric structure between the observation pixels and endmembers.

- 3) The improvement of accuracy as well as speed. From the results of the synthetic experiments, we can see that the proposed algorithm has obviously better unmixing accuracy than the FCLS solutions. And it always has a lower computational complexity than the SPU algorithm, whereas the time consumptions of the FCLS and C-SUNSAL algorithms are less than that of DGAE only when the number of endmembers is larger.
- 4) Unmixing large-scale data with high performance. From the Experiment 3, we can see that the computational complexity of the DGAE algorithm has a lower slope to



TABLE IX  
RESULTS OF ALGORITHMS WITH DIFFERENT NUMBER OF ENDMEMBERS

Number of Endmembers	RE				Runtime (s)				Recursion Numbers of DGAE
	FCLS	C-SUnSAL	SPU	DGAE	FCLS	C-SUnSAL	SPU	DGAE	
8	0.0137	0.0136	0.0146	0.0137	15.9538	3.4053	7.1747	1.7431	2288
10	0.0121	0.0121	0.0132	0.0122	18.9277	4.3548	16.4306	6.5013	15 226
12	0.0114	0.0114	0.0129	0.0115	22.4859	5.1923	40.2860	19.8302	51 413
14	0.0096	0.0095	0.0115	0.0098	26.2735	5.9543	91.0356	43.5952	11 4689
16	0.0091	0.0088	0.0113	0.0093	30.5140	6.8514	184.8907	81.4321	20 7626
18	0.0075	0.0076	0.0103	0.0079	35.3125	7.5901	317.9228	120.4878	29 5381
20	0.0077	0.0076	0.0111	0.0080	40.0619	8.4520	537.2005	174.5244	40 3072
22	0.0073	0.0075	0.0114	0.0078	46.3113	9.3624	827.7057	237.2765	51 7395

TABLE X  
RESULTS OF FOUR ALGORITHMS FOR INTERESTED REGION

$p$	RE				Runtime (s)			
	FCLS	C-SUnSAL	SPU	DGAE	FCLS	C-SUnSAL	SPU	DGAE
3	0.0083	0.0083	0.0083	0.0083	0.4635	0.1122	0.0513	0.0154
4	0.0086	0.0086	0.0091	0.0087	0.6499	0.1313	0.1467	0.0235
5	0.0093	0.0093	0.0099	0.0093	1.0199	0.1471	0.2114	0.0438
6	0.0174	0.0174	0.0190	0.0175	0.8876	0.2012	0.4102	0.0739



Fig. 8. (a) Moffett Field. (b) Region of interest. Represented in synthetic color.

the dimensionality of data. So, it is unnecessary for the DGAE algorithm to execute the dimensionality reduction or band selection before unmixing, which might lose useful information and discard some important features. Meantime, the runtime of the DGAE algorithm increases slowly with the increase of the number of pixels in the images. Therefore, we consider that the DGAE algorithm should be very useful for the large-scale data, which is often encountered in real world applications.

### B. Results on Real Hyperspectral Images

Here, we apply the proposed DGAE algorithm to real hyperspectral data sets to illustrate its performance and compare it with the FCLS, C-SUnSAL, and SPU algorithms. The real hyperspectral data sets are collected by airborne visible/infrared imaging spectrometer (AVIRIS) sensor, which generates 224 bands across the whole spectral region from 0.41 to 2.45  $\mu\text{m}$  with a 10-nm bandwidth [45].

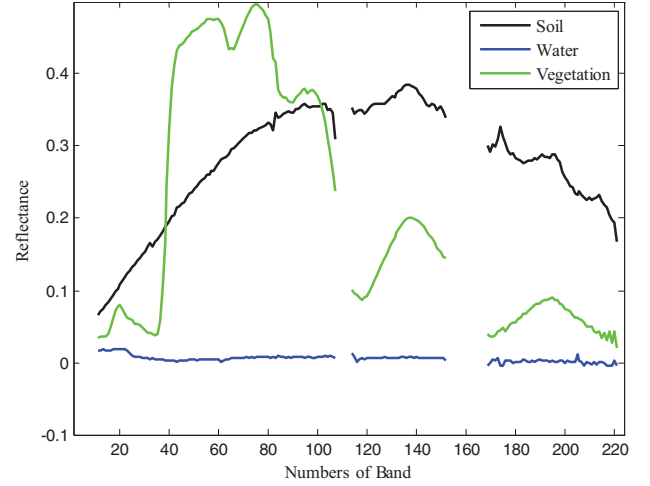


Fig. 9. Endmember-spectra obtained by N-FINDR for interested region of Moffett field.

In interpreting the following results of real data set experiments, several caveats should be kept in mind. Because the real images are more complex and uncertain than the synthetic images, the slight difference may exist between the results for the real and synthetic images. Because of the lack of true abundance maps, we only use the RE to evaluate the unmixing results of four algorithms qualitatively but not quantitatively. Also, the runtime will be considered as a criterion to evaluate the performances of the four algorithms.

1) *Experiment 1 (Experiment on the Cuprite Data Set):*  
The first real image scene we use is over the Cuprite mining site, Nevada, acquired in 1997.<sup>4</sup> This scene is widely used to validate the performances of unmixing algorithms, and the geologic characteristics of this area, dominated by kinds of

<sup>4</sup>Available on the website <http://aviris.jpl.nasa.gov/html/aviris.freedata.html>.

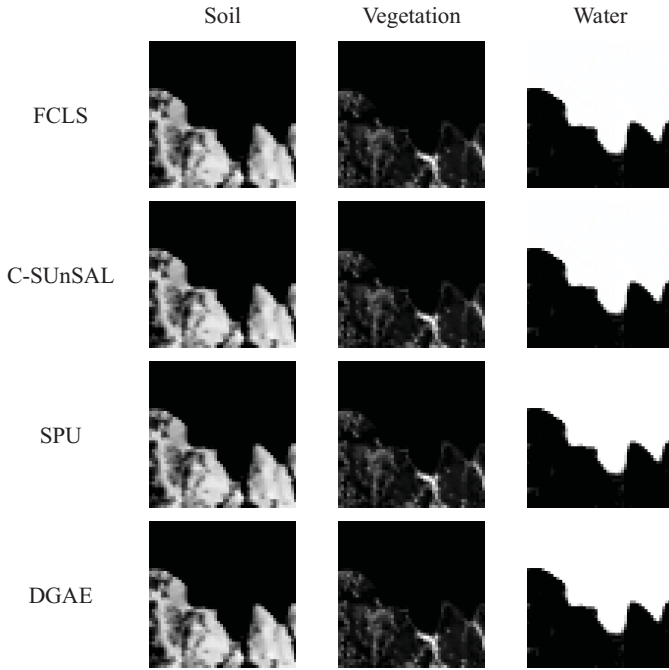


Fig. 10. Abundance maps estimated by four algorithms. Black: absence of material. White: presence of material.

minerals, are investigated in [45]. The image used in the experiment is a 224-band subscene with the size of  $250 \times 190$  gotten from the rfl file named f970619t01p02\_r02\_sc04.a.rfl. For illustrative purposes, Fig. 6(a) and (b) shows the ground-truth classification maps for this region alongside the interested subscene, which cover lots of minerals and are suitable for testing the performances for the various number of endmembers. Fig. 6(c) shows the synthetic color image of entire image and the subscene of interest.

Before unmixing, a preprocessing to the original image is necessary. Only 189 bands, i.e., bands 4–104, 116–149, and 171–224, are remained for use by discarding water absorption and low SNR bands [46].

Unlike the synthetic data set, we assume that the number of endmembers in this area is unknown. Two different methods are used in this experiment to estimate the number of endmembers  $p$  in the original hyperspectral image: the virtual dimensionality (VD) method [47] and the HySime method [48]. The former is a Neyman–Pearson detector based on prescribed probabilities of false alarm (PFs), and the latter does not have any input parameters. In spite of different criticisms to these methods, they are the two most widely used approaches for estimating the number of endmembers available in the literature [2]. The results of three VD algorithms with different PFs and HySime algorithm are shown in Table VIII. The number of endmembers in the subarea may be overestimated by the HySime and VD algorithms [49]. Not to miss some endmembers and according to the ground truth presented in [50] and [51] and showed in Fig. 5(a) and (b), we set the  $p$  in the range [8, 22].

As in the synthetic data experiments, we use the N-FINDR algorithm to locate the endmember signatures for every value

of  $p$  in the range [8, 22]. The found endmembers correspond to different minerals in this data set. As the typical unmixing results, we give the abundance maps of the Alunite endmember in Fig. 7. As can be seen, the four algorithms generate qualitatively approximate abundance maps.

For various numbers of endmembers, the REs, and runtimes are shown in Table IX. We can conclude that the FCLS and C-SUNSAL algorithms have the lower RE than the other two algorithms. The reason is that both the aims of FCLS and C-SUNSAL are just to minimize the RE, which gives a best fit for the LSMM. The DGAE algorithm has a slightly higher result compared with the FCLS algorithm, whereas SPU has the highest RE among the four algorithms. Different from the result in Experiment 4 shown in Table VI, the RE of SPU has a downward trend. The reason is that the number of endmembers in the synthetic data sets is known beforehand whereas it is not the case with the real hyperspectral data.

Among the four algorithms, the C-SUNSAL algorithm always has the lowest runtime in the whole range when  $p \geq 10$ . As expected, the DGAE algorithm has a lower runtime for all the  $p$  in the range [8, 22] than the SPU algorithm, whereas it has less runtime than the FCLS and C-SUNSAL algorithm on this data set only when  $p < 10$ .

Just as in the synthetic data experiment, we give the number of recursions in the DGAE algorithm for various number of endmembers in the rightmost column of Table IX. It is clear that the dramatic increase of number of recursions is the main reason for rapid increase of computational complexity.

*2) Experiment 2 (Experiment on the Moffett Field Data Set):* Another real data is acquired over Moffett Field (CA, USA) in 1997 by the JPL spectro-imager AVIRIS<sup>4</sup>. Many previous works have used this image to illustrate and compare algorithm performance with hyperspectral images [52]. We apply the four algorithms on the region of interest with the size of  $50 \times 50$  shown in Fig. 8. The bands 1–10, 108–113, 153–168, and 222–224 are removed because of water absorption and low SNR.

The ground-truth in this area, which contains a lake (top) and a coastal zone (bottom), is relatively simple and mainly composed of three components: water, soil, and vegetation as shown in Fig. 8. Fig. 9 shows the endmembers' spectra corresponding to water, soil, and vegetation extracted by the N-FINDR algorithm. Here as typical unmixing results, we give the abundance maps for three endmembers, i.e., water, soil, and vegetation, in Fig. 10.

From the recent investigation, we know that, because of complex geographical features and the existence of nonlinearity, more than three endmembers are actually present in this area [53], [54]. Therefore, we set the number of endmembers  $p$  in the range [3, 6]. Table X shows the results of four algorithms including REs and runtimes.

The results indicate that among the four algorithms, the FCLS algorithm has the lowest RE, meaning the best fit for the LSMM, whereas the DGAE algorithm has a close RE to the FCLS algorithm. Considering the runtime, the DGAE is fastest for this subscene and the runtime increases with the increase of  $p$ .

## V. CONCLUSION

In this paper, using high-dimensional geometry theory we proposed a fast and effective algorithm called DGAE, which provided the fully constrained result of LSU problem using distances between the endmembers and the observation pixels. According to this approach, the observation pixels in the hyperspectral images can be divided into three types: interior points, exterior points I, and exterior points II, respectively. Then, three subalgorithms were designed to deal with the three types. The final barycentric coordinates, corresponding to the abundances of observation pixels, were obtained using a new calculation formula for barycentric coordinates in the form of Cayley–Menger matrix. The entire method had a recursive implementation with a low computational complexity.

The experimental results on the synthetic and real hyperspectral images showed that the proposed DGAE algorithm had an obviously better unmixing accuracy than the other three state-of-the-art algorithms, namely FCLS, C-SUnSAL, and SPU algorithms, in which C-SUnSAL was used to compute the FCLS solutions. And the computational complexity of DGAE was always much lower than that of SPU, which was alike a recursive algorithm. However, when the endmember-number  $p$  was larger, the computational complexity of DGAE will exceed the FCLS and C-SUnSAL algorithms because of the rapid increase of recursions. The C-SUnSAL algorithm was the fastest as expected.

In future research efforts, there were several aspects worthy to be investigated. Initially, the computational complexity needs to be decreased especially when the number of endmembers was large. A nonrecursive or GPU-based version was worthwhile to be considered. Next, some other versions of distance, such as curvilinear distance, geodesic distance, and so on, could be introduced to solve the linear or nonlinear spectral unmixing problems.

## APPENDIX A

### PROOF OF THEOREM 1

The square distance between endmember  $\mathbf{e}_i$  and  $\mathbf{e}_j$  is

$$\begin{aligned} d_{i,j}^2 &= (\mathbf{e}_i - \mathbf{e}_j)^T (\mathbf{e}_i - \mathbf{e}_j) \\ &= (\mathbf{e}_i - \mathbf{q} + \mathbf{q} - \mathbf{e}_j)^T (\mathbf{e}_i - \mathbf{q} + \mathbf{q} - \mathbf{e}_j) \\ &= d_{i,q}^2 + d_{j,q}^2 + 2(\mathbf{e}_i - \mathbf{q})^T (\mathbf{q} - \mathbf{e}_j). \end{aligned} \quad (33)$$

Multiplying both sides of (33) by  $s_i$  with  $i$  from 1 to  $p$ , we can get

$$\begin{aligned} \sum_{i=1}^p s_i d_{i,j}^2 &= \sum_{i=1}^p s_i d_{i,l}^2 + \sum_{i=1}^p s_i d_{j,l}^2 \\ &\quad + 2 \left( \sum_{i=1}^p s_i \mathbf{e}_i - \mathbf{l} \right)^T (\mathbf{l} - \mathbf{e}_j) \\ &= \sum_{i=1}^p s_i d_{i,l}^2 + d_{j,l}^2 \end{aligned} \quad (34)$$

which means that

$$d_{j,l}^2 = \sum_{i=1}^p s_i d_{i,j}^2 - \mathbf{s}^T \mathbf{d} \quad (35)$$

Combining (6) and (35) and  $\mathbf{s}^T \mathbf{l} = 1$ , we can obtain

$$\mathbf{C}^{p+1} \begin{pmatrix} \varsigma \\ \mathbf{s} \end{pmatrix} = \begin{pmatrix} 1 \\ \mathbf{d} \end{pmatrix} \quad (36)$$

with  $\varsigma = -\mathbf{s}^T \mathbf{d}$ .

With the inverse of nonsingular matrix  $\mathbf{C}^{p+1}$ , the barycentric coordinate  $\mathbf{s}$  of observation pixel  $\mathbf{q}$  can be given using

$$\begin{pmatrix} \varsigma \\ \mathbf{s} \end{pmatrix} = (\mathbf{C}^{p+1})^{-1} \begin{pmatrix} 1 \\ \mathbf{d} \end{pmatrix}. \quad (37)$$

The proof of Theorem 1 is completed.

## APPENDIX B

### PROOF OF UPPER AND LOWER BOUND IN (22)

Given the function with respect to  $x$

$$g(x) = \frac{2\eta - \eta^2 x}{(1 - \eta x)^2}, \quad x \in [\lambda_p, \lambda_1], \quad \eta \in \left(0, \frac{1}{\lambda_1}\right). \quad (38)$$

Noting that  $x \leq \lambda_1 < 1/\eta$ , we can know that the derivative of  $g(x)$  with respect to  $x$  is positive. So the codomain of  $g(x)$  is  $[g(\lambda_p), g(\lambda_1)]$ , which means that

$$\frac{2\eta - \eta^2 \lambda_p}{(1 - \eta \lambda_p)^2} \leq \frac{2\eta - \eta^2 \lambda_j}{(1 - \eta \lambda_j)^2} \leq \frac{2\eta - \eta^2 \lambda_1}{(1 - \eta \lambda_1)^2}. \quad (39)$$

Then combining (21) and (39) and  $\lambda_p = 0$ , we can get a upper and lower bound on the solution to (21)

$$\frac{1}{\lambda_1} \left( 1 - \frac{1}{\sqrt{1 + \lambda_1 \zeta}} \right) \leq \eta \leq \frac{1}{2} \zeta \quad (40)$$

with  $\zeta = c / \sum_{j=1}^p \|\mathbf{b}^T \mathbf{u}_j\|^2 \geq 0$ .

The upper and lower bound in (40) is usually tighter than the bound  $(0, 1/\lambda_1)$  but not always. Summing up all these facts shows that the solution to (21) is unique and belongs to the following interval:

$$\frac{1}{\lambda_1} \left( 1 - \frac{1}{\sqrt{1 + \lambda_1 \zeta}} \right) \leq \eta \leq \min \left( \frac{1}{2} \zeta, \frac{1}{\lambda_1} \right) \text{ and } \eta \neq \frac{1}{\lambda_1}. \quad (41)$$

So the proof is completed.

## APPENDIX C

### PROOF OF THEOREM 3

$$\mathbf{q} = \sum_{j=1}^p s_j \mathbf{e}_j, \quad \mathbf{g} = \frac{1}{p} \sum_{j=1}^p \mathbf{e}_j \quad \text{so}$$

$$\begin{aligned} \mathbf{q} - p s_i \mathbf{g} &= \sum_{j=1}^p s_j \mathbf{e}_j - s_i \sum_{j=1}^p \mathbf{e}_j \\ &= \sum_{j=1, j \neq i}^p (s_j - s_i) \mathbf{e}_j. \end{aligned} \quad (42)$$

From (42), we can get

$$\mathbf{q} = \sum_{j=1, j \neq i}^p (s_j - s_i) \mathbf{e}_j + p s_i \mathbf{g}. \quad (43)$$

We know that if  $\mathbf{q} \in S_i$ , there exists a vector  $\mathbf{t} = (t_1, t_2, \dots, t_p)^T$  with  $0 \leq t_j \leq 1$ ,  $j = 1, 2, \dots, i-1, i+1, \dots, p$ , and  $\sum_{j=1}^p t_j = 1$ , which satisfies the definition of  $S_i$ . So we can get  $s_j > s_i$  ( $j = 1, 2, \dots, i-1, i+1, \dots, p$ ). This is the proof of the necessity of Theorem 3.

The proof of sufficiency is as follow. Combining  $s_j > s_i$  ( $j = 1, 2, \dots, i-1, i+1, \dots, p$ ) with  $\sum_{j=1, j \neq i}^p (s_j - s_i) + ps_i = 1$ , we know that there exists a vector

$$\mathbf{t} = (s_1 - s_i, s_2 - s_i, \dots, s_{i-1} - s_i, ps_i, s_{i+1} - s_i, \dots, s_p - s_i)^T \quad (44)$$

which satisfies the definition of  $S_i$ .

So the proof of Theorem 3 is completed.

#### APPENDIX D

##### PROOF OF THEOREM 4

**Theorem 4:** Let  $\mathbf{C}^{p+1}$  be the  $(p+1)$ -dimensional Cayley–Menger matrix and the  $(\mathbf{C}^{p+1})_{(i+1)}^\vee$  and  $((\mathbf{C}^{p+1})^{-1})_{(i+1)}^\vee$  are the remaining submatrix after the  $(i+1)$ th column and the  $(i+1)$ th row of matrix  $\mathbf{C}^{p+1}$  and  $(\mathbf{C}^{p+1})^{-1}$  are removed, respectively. Then we can get  $((\mathbf{C}^{p+1})_{(i+1)}^\vee)^{-1}$  using the following formula:

$$((\mathbf{C}^{p+1})_{(i+1)}^\vee)^{-1} = ((\mathbf{C}^{p+1})^{-1})_{(i+1)}^\vee - \mathbf{v}\mathbf{v}^T \quad (45)$$

where the vector  $\mathbf{v}$  is the remaining vector of the  $(i+1)$ th column of  $(\mathbf{C}^{p+1})^{-1}$  after the  $(i+1)$ th element is removed and  $v$  is the  $(i+1)$ th element.

**Proof:** From the  $\mathbf{C}^{p+1}$  defined in (6), we can get

$$\mathbf{C}^{p+1}\mathbf{T} = \begin{pmatrix} (\mathbf{C}^{p+1})_{(i+1)}^\vee & \mathbf{r} \\ \mathbf{r}^T & 0 \end{pmatrix} \quad (46)$$

where  $\mathbf{T}$  is elementary matrix of Type II and  $\mathbf{r} = (1, d_{i,1}^2, \dots, d_{i,i-1}^2, d_{i,i+1}^2, \dots, d_{i,p}^2)^T$ .

Using the Hermitian matrix inversion theorem [55], we can obtain

$$\begin{pmatrix} (\mathbf{C}^{p+1})_{(i+1)}^\vee & \mathbf{r} \\ \mathbf{r}^T & 0 \end{pmatrix}^{-1} = \begin{pmatrix} ((\mathbf{C}^{p+1})_{(i+1)}^\vee)^{-1} & \mathbf{0} \\ \mathbf{0}^T & 0 \end{pmatrix} + \frac{1}{\beta} \begin{pmatrix} \mathbf{b}\mathbf{b}^T & \mathbf{b} \\ \mathbf{b}^T & 1 \end{pmatrix} \quad (47)$$

where  $\beta = -\mathbf{r}^T \mathbf{b}$ ,  $\mathbf{b} = -((\mathbf{C}^{p+1})_{(i+1)}^\vee)^{-1} \mathbf{r}$ .

We note that

$$\begin{aligned} (\mathbf{C}^{p+1}\mathbf{T})^{-1} &= \mathbf{T}^{-1}(\mathbf{C}^{p+1})^{-1} = (\mathbf{C}^{p+1})^{-1} \\ \mathbf{T} &= \begin{pmatrix} ((\mathbf{C}^{p+1})^{-1})_{(i+1)}^\vee & \mathbf{v} \\ \mathbf{v}^T & v \end{pmatrix}. \end{aligned} \quad (48)$$

It then follows from (47) and (48) that

$$\begin{aligned} ((\mathbf{C}^{p+1})^{-1})_{(i+1)}^\vee &= ((\mathbf{C}^{p+1})_{(i+1)}^\vee)^{-1} \\ &+ \frac{1}{\beta} \mathbf{b}\mathbf{b}^T, \mathbf{v} = \frac{1}{\beta} \mathbf{b}, \quad v = \frac{1}{\beta}. \end{aligned} \quad (49)$$

So (45) is obtained.

#### ACKNOWLEDGMENT

The authors would like to thank Prof. C.-I. Chang, Prof. R. Heylen, and Prof. J. M. B. Dias for providing the fully constrained least squares code, the simplex-projection unmixing code, and the C-SUNSAL code, respectively, Dr. M.-D. Iordache and Prof. A. Plaza for sharing the ASTER library. The authors would also like to thank the Associate Editor and the anonymous reviewers for their insightful comments and constructive suggestions that greatly helped to improve the technical quality and presentation of this paper.

#### REFERENCES

- [1] C. I. Chang, *Hyperspectral Imaging: Techniques for Spectral Detection and Classification*. New York, USA: Plenum, 2003.
- [2] J. Bioucas-Dias, A. Plaza, N. Dobigeon, M. Parente, Q. Du, P. Gader, and J. Chanussot, "Hyperspectral unmixing overview: Geometrical, statistical, and sparse regression-based approaches," *IEEE J. Sel. Topics Appl. Earth Observat. Remote Sens.*, vol. 5, no. 2, pp. 354–379, Apr. 2012.
- [3] M. Berman, P. Conner, L. Whitbourn, D. Coward, B. Osborne, and M. Southan, "Classification of sound and stained wheat grains using visible and near infrared hyperspectral image analysis," *J. Near Infr. Spectroscopy*, vol. 15, no. 6, pp. 351–358, 2007.
- [4] A. Gowen, C. O'Donnell, P. Cullen, G. Downey, and J. Frias, "Hyperspectral imaging—an emerging process analytical tool for food quality and safety control," *Trends Food Sci. Technol.*, vol. 18, no. 12, pp. 590–598, Dec. 2007.
- [5] N. Keshava and J. F. Mustard, "Spectral unmixing," *IEEE Signal Process. Mag.*, vol. 19, no. 1, pp. 44–57, Jan. 2002.
- [6] A. Plaza, J. A. Benediktsson, J. W. Boardman, J. Brazile, L. Bruzzone, G. Camps-Valls, J. Chanussot, M. Fauvel, P. Gamba, A. Gualtieri, M. Marconcini, J. C. Tilton, and G. Trianni, "Recent advances in techniques for hyperspectral image processing," *Remote Sens. Environ.*, vol. 113, no. 1, pp. S110–S122, Sep. 2009.
- [7] D. S. Lu, M. Batistella, E. Moran, and P. Mausel, "Application of spectral mixture analysis to amazonian land-use and land-cover classification," *Int. J. Remote Sens.*, vol. 25, no. 23, pp. 5345–5358, Dec. 2004.
- [8] D. Landgrebe, "Hyperspectral image data analysis," *IEEE Signal Process. Mag.*, vol. 19, no. 1, pp. 17–28, Jan. 2002.
- [9] D. Manolakis and G. Shaw, "Detection algorithms for hyperspectral imaging applications," *IEEE Signal Process. Mag.*, vol. 19, no. 1, pp. 29–43, Jan. 2002.
- [10] J. B. Adams, M. O. Smith, and P. E. Johnson, "Spectral mixture modeling: A new analysis of rock and soil types at the viking lander 1 suite," *J. Geophys. Res.*, vol. 91, no. B8, pp. 8098–8112, Jul. 1986.
- [11] L. Miao, H. Qi, and H. Szu, "A maximum entropy approach to unsupervised mixed-pixel decomposition spectral unmixing," *IEEE Trans. Image Process.*, vol. 16, no. 4, pp. 1008–1021, Apr. 2007.
- [12] D. C. Heinz and C. I. Chang, "Fully constrained least square linear spectral unmixing analysis method for material quantification in hyperspectral imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 39, no. 3, pp. 529–545, Mar. 2001.
- [13] M. D. Iordache, J. Bioucas-Dias, and A. Plaza, "Sparse unmixing of hyperspectral data," *IEEE Trans. Geosci. Remote Sens.*, vol. 49, no. 6, pp. 2014–2039, Jun. 2011.
- [14] A. Ambikapathi, T. H. Chan, W. K. Ma, and C. Y. Chi, "Chance-constrained robust minimum-volume enclosing simplex algorithm for hyperspectral unmixing," *IEEE Trans. Geosci. Remote Sens.*, vol. 49, no. 11, pp. 4194–4209, Nov. 2011.
- [15] H. Li, Y. Wang, and X. Wang, "Pixel-unmixing moderate-resolution remote sensing imagery using pairwise coupling support vector machines: A case study," *IEEE Trans. Geosci. Remote Sens.*, vol. 49, no. 11, pp. 4298–4307, Nov. 2011.
- [16] 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.
- [17] J. J. Settle and N. A. Drake, "Linear mixing and estimation of ground cover proportions," *Int. J. Remote Sens.*, vol. 14, no. 6, pp. 1159–1177, 1993.

- [18] R. Bro and S. D. Jong, "A fast non-negativity constrained least squares algorithm," *J. Chemometrics*, vol. 11, no. 5, pp. 393–401, Sep.–Oct. 1997.
- [19] Y. H. Hu, H. B. Lee, and F. L. Scarpace, "Optimal linear spectral unmixing," *IEEE Trans. Geosci. Remote Sens.*, vol. 37, no. 1, pp. 639–644, Jan. 1999.
- [20] P. Gill, W. Murray, and M. Wright, *Numerical Linear Algebra and Optimization*. Reading, MA, USA: Addison-Wesley, 1991.
- [21] J. L. Silván-Cárdenas and L. Wang, "Fully constrained linear spectral unmixing: Analytic solution using fuzzy sets," *IEEE Trans. Geosci. Remote Sens.*, vol. 48, no. 11, pp. 3992–4002, Nov. 2011.
- [22] X. H. Chen, J. Chen, B. Somer, J. Wu, and P. Coppin, "A quantitative analysis of virtual endmembers' increased impact on the collinearity effect in spectral unmixing," *IEEE Trans. Geosci. Remote Sens.*, vol. 49, no. 8, pp. 2945–2956, Aug. 2011.
- [23] J. Eckstein and D. Bertsekas, "On the Douglas–Rachford splitting method and the proximal point algorithm for maximal monotone operators," *J. Math. Progr.*, vol. 55, no. 3, pp. 293–318, Jun. 1992.
- [24] J. Bioucas-Dias and M. Figueiredo, "Alternating direction algorithms for constrained sparse regression: Application to hyperspectral unmixing," in *Proc. IEEE GRSS Workshop Hyperspectral Image Signal Process. Evolution Remote Sens.*, vol. 1, Jun. 2010, pp. 1–4.
- [25] J. Boardman, "Automating spectral unmixing of AVIRIS data using convex geometry concepts," in *Proc. Ann. JPL Airborne Geosci. Workshop*, vol. 1, 1993, pp. 11–14.
- [26] P. Honeine, and C. Richard, "Geometric unmixing of large hyperspectral images: A barycentric coordinate approach," *IEEE Trans. Geosci. Remote Sens.*, vol. 50, no. 6, pp. 2185–2195, Jun. 2011.
- [27] P. Bajorski, "Simplex projection methods for selection of endmembers in hyperspectral imagery," in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, vol. 5, Sep. 2004, pp. 3207–3210.
- [28] C. Michelot, "A finite algorithm for finding the projection of a point onto the canonical simplex of," *J. Optim. Theory Appl.*, vol. 50, no. 1, pp. 195–200, Jul. 1986.
- [29] H. J. H. Tuenter, "The minimum L2-distance projection onto the canonical simplex: A simple algorithm," *Algo Res. Quart.*, vol. 4, pp. 53–55, Dec. 2001.
- [30] R. Heylen, D. Burazerovic, and P. Scheunders, "Fully constrained least squares spectral unmixing by simplex projection," *IEEE Trans. Geosci. Remote Sens.*, vol. 49, no. 11, pp. 4112–4122, Nov. 2011.
- [31] A. A. Ungar, *Barycentric Calculus in Euclidean and Hyperbolic Geometry: A Comparative Introduction*. Singapore, World Scientific, 2010.
- [32] L. M. Blumenthal, *Theory and Applications of Distance Geometry*. New York, USA: Chelsea, 1970.
- [33] S. Roman, *Advanced Linear Algebra (Graduate Texts in Mathematics)*, 2nd ed. New York, USA: Springer-Verlag, 2000.
- [34] J. M. Lee, *Introduction to Topological Manifold (Graduate Texts in Mathematics)*. New York, USA: Springer-Verlag, 2000.
- [35] I. H. Lin, *Geometric Linear Algebra*, vol. 2. Singapore: World Scientific, 2008.
- [36] M. Cao, D. O. Brian, A. Anderson, and S. Morse, "Sensor network localization with imprecise distance," *Syst. Control Lett.*, vol. 55, no. 11, pp. 887–893, Nov. 2006.
- [37] S. D. Conte, and C. de Boor, *Elementary Numerical Analysis*. New York, USA: McGraw-Hill, 1980.
- [38] J. M. P. Nascimento and J. M. B. 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] J. Plaza, E. M. T. Hendrix, I. Garcia, G. Martin, and A. Plaza, "On endmember identification in hyperspectral images without pure pixels: A comparison of algorithms," *J. Math. Imag. Vis.*, vol. 42, nos. 2–3, pp. 163–175, Feb. 2012.
- [40] M. T. Heath, *Scientific Computing: An Introductory Survey*. New York, USA: McGraw-Hill, 2002.
- [41] R. Heylen and P. Scheunders, "A fast geometric algorithm for solving the inversion problem in spectral unmixing," in *Proc. IEEE GRSS Workshop Hyperspectral Image Signal Process. Evol. Remote Sens.*, Jun. 2012, pp. 1–15.
- [42] A. M. Baldridge, S. J. Hook, C. I. Grove, and G. Rivera, "The ASTER spectral library version 2.0," *Remote Sens. Environ.*, vol. 113, no. 4, pp. 711–715, Apr. 2009.
- [43] M. E. Winter, "N-FINDR: An algorithm for fast autonomous spectral endmember determination in hyperspectral data," *Proc. SPIE Image Spectrometry V*, vol. 3753, pp. 266–277, 1999.
- [44] Y. Qian, S. Jia, J. Zhou, and A. Robles-Kelly, "Hyperspectral unmixing via  $L_{1/2}$  sparsity-constrained nonnegative matrix factorization," *IEEE Trans. Geosci. Remote Sens.*, vol. 49, no. 11, pp. 4282–4297, Nov. 2011.
- [45] R. N. Clark and G. A. Swayze, "Evolution in imaging spectroscopy analysis and sensor signal-to-noise: An examination of how far we have come," in *Proc. 6th Annu. JPL Airborne Earth Sci. Workshop*, 1996, pp. 49–53.
- [46] S. Cai, Q. Du, and R. J. Moorhead, "Hyperspectral imagery visualization using double layers," *IEEE Trans. Geosci. Remote Sens.*, vol. 45, no. 10, pp. 3028–3036, Oct. 2007.
- [47] 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.
- [48] J. M. Bioucas-Dias and J. M. P. Nascimento, "Hyperspectral subspace identification," *IEEE Trans. Geosci. Remote Sens.*, vol. 46, no. 8, pp. 2435–2445, Aug. 2008.
- [49] J. Liu and J. Zhang, "A new maximum simplex volume method based on householder transformation for endmember extraction," *IEEE Trans. Geosci. Remote Sens.*, vol. 50, no. 1, pp. 104–118, Jan. 2012.
- [50] R. N. Clark, G. A. Swayze, and A. Gallagher, "Mapping minerals with imaging spectroscopy, U.S. Geological Survey," *Office Mineral Resour. Bull.*, vol. 2039, pp. 141–150, 1993.
- [51] G. Swayze, R. N. Clark, F. Kruse, S. Sutley, and A. Gallagher, "Ground-truthing AVIRIS mineral mapping at Cuprite, Nevada," in *Proc. JPL, Summaries 3rd Ann. JPL Airborne Geosci. Workshop*, Jun. 1992, pp. 47–49.
- [52] N. Dobigeon, J. Y. Tourneret, and C. I. Chang, "Semi-supervised linear spectral unmixing using a hierarchical Bayesian model for hyperspectral imagery," *IEEE Trans. Signal Process.*, vol. 56, no. 7, pp. 2684–2695, Jul. 2008.
- [53] O. Eches, N. Dobigeon, and J. Y. Tourneret, "Estimating the number of endmembers in hyperspectral images using the normal compositional model and a hierarchical Bayesian algorithm," *IEEE J. Sel. Topics Signal Process.*, vol. 4, no. 3, pp. 582–591, Jun. 2010.
- [54] A. Halimi, Y. Altmann, N. Dobigeon, and J. Y. Tourneret, "Nonlinear unmixing of hyperspectral images using a generalized bilinear model," *IEEE Trans. Geosci. Remote Sens.*, vol. 49, no. 11, pp. 4153–4162, Nov. 2011.
- [55] B. Noble and J. W. Danniel, *Applied Linear Algebra*, 3rd ed. Englewood Cliffs, NJ, USA: Prentice-Hall, 1988.



**Hanye Pu** received the B.S. degree from Fudan University, Shanghai, China, in 2009, where he is currently pursuing the Ph.D. degree with the Department of Electronic Engineering.

His current research interests include hyperspectral image analysis, manifold learning, image processing, and pattern recognition.



**Wei Xia** (S'11) received the B.E. degree from Shandong University, Jinan, China, in 2008. He is currently pursuing the Ph.D. degree with the Department of Electronic Engineering, Fudan University, Shanghai, China.

His current research interests include hyperspectral image analysis, image processing, and pattern recognition.





**Bin Wang** (M'03–SM'11) received the B.S. and M.S. degrees in communication and electronic systems from Xidian University, Xi'an, China, in 1985 and 1988, respectively, and the Ph.D. degree in system science from Kobe University, Kobe, Japan, in 1999.

He was with Xidian University as a Teacher. From 1999 to 2000, he was with the Communications Research Laboratory, Ministry of Posts and Telecommunications, Japan, as a Research Fellow, working on magnetoencephalography signal processing and its application for brain science. He was with the Department of Etching, Tokyo Electron AT Ltd., Tokyo, Japan, as a Senior Supervisor, from 2000 to 2002, dealing with the development of advanced process control systems for etching semiconductor equipment. Since 2002, he has been with the Department of Electronic Engineering, Fudan University, Shanghai, China, where he is currently a Professor and the Leader of the Image and Intelligence Laboratory. He has published more than 80 scientific papers in important domestic and international periodicals. He holds several patents. His current research interests include multispectral/hyperspectral image analysis, automatic target detection and recognition, pattern recognition, signal detection and estimation, and machine learning.



**Geng-Ming Jiang** received the B.S. degree in photogrammetric engineering and remote sensing from the Wuhan Technical University of Surveying and Mapping, Wuhan, China, the M.S. degree in cartography and geographical information system from the Institute of Remote Sensing Applications, Chinese Academy of Sciences, Beijing, China, and the Ph.D. degree in remote sensing from Université Louis Pasteur de Strasbourg I, Strasbourg, France in 1997, 2003, and 2007, respectively.

He is currently an Associate Professor with the Key Laboratory of Wave Scattering and Remote Sensing Information, Fudan University, Shanghai, China. His current research interests include quantitative remote sensing and digital image processing.