We are IntechOpen, the world's leading publisher of Open Access books Built by scientists, for scientists

6,900
Open access books available

185,000
International authors and editors

200M
Downloads

Our authors are among the

154
Countries delivered to

TOP 1%
most cited scientists

12.2%
Contributors from top 500 universities



WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.
For more information visit www.intechopen.com



Chapter 9

Graph Construction for Hyperspectral Data Unmixing

Zeng Li, Jie Chen and Susanto Rahardja

Additional information is available at the end of the chapter

http://dx.doi.org/10.5772/intechopen.73158

Abstract

This chapter presents graph construction for hyperspectral data and associated unmixing methods based on graph regularization. Graph is a ubiquitous mathematical tool for modeling relations between objects under study. In the context of hyperspectral image analysis, constructing graphs can be useful to relate pixels in order to perform corporative analysis instead of analyzing each pixel individually. In this chapter, we review fundamental elements of graphs and present different ways to construct graphs in both spatial and spectral senses for hyperspectral images. By incorporating a graph regularization, we then formulate a general hyperspectral unmixing problem that can be important for applications such as remote sensing and environment monitoring. Alternating direction method of multipliers (ADMM) is also presented as a generic tool for solving the formulated unmixing problems. Experiments validate the proposed scheme with both synthetic data and real remote sensing data.

Keywords: hyperspectral imaging, graph construction, spectral unmixing, graph regularization, spectral-spatial correlation

1. Introduction

Hyperspectral imaging analysis has found a wide range of applications including agricultural monitoring, environment detection, meteorological information forecast, medical examination, and camouflage tests [1]. In a hyperspectral image, pixels are typically mixtures of several pure material components due to the limitation of spatial resolution and intimate interactions among materials. Spectral unmixing is thus one of the most important tasks in hyperspectral data analysis, aiming to separate the observed pixel spectra into a collection of constituent spectra, or spectral signatures, called endmembers and to estimate fractions associated with each component called abundances. Spectral unmixing provides a comprehensive and quantitative mapping of the elementary materials that are present in the acquired data, and it is widely used for many

IntechOpen

© 2018 The Author(s). Licensee IntechOpen. This chapter is distributed under the terms of the Creative Commons Attribution License (http://creativecommons.org/licenses/by/3.0), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited. CC BY

applications, such as determining the constitutions of geological mixtures and performing a classification of crops and vegetation.

Most spectral unmixing approaches are designed based on pre-assumed mixture models that describe in an analytical way how the endmembers are combined to mixed spectra measured by the sensor [2]. The linear mixing model (LMM) is the most widely used one, and it assumes that the mixing occurs at a macroscopic scale [3]. A measured spectrum is the linear combinations of the endmembers, weighted by the fractional abundances. To be physically interpretable, LMM is usually performed with two physical constraints, abundance nonnegative constraint (ANC) and abundance sum-to-one constraint (ASC). Multiple scattering effects and intimate interactions in real environment require using nonlinear mixture models. Such models include intimate mixture model [4], bilinear model [5], linear-quadratic mixing model (LQM) [6], polynomial post-nonlinear mixing model (PPNM) [7], to cite a few. However, due to the simplicity and interpretability of the analysis results, LMM-based unmixing strategies are mostly used in practice [2]. A number of unmixing algorithms are proposed, including long-standing geometrical and statistical approaches and the recently introduced sparse regression-based unmixing algorithms [8–11].

Considering inherent spatial-spectral duality exists in hyperspectral data, regularized unmixing algorithms have been proposed in recent years to make use of spectral information and spatial contextual information to enhance the unmixing performance. For instance, in [8], authors introduce a total variation (TV) regularizer to promote spatial consistency of estimated abundances. In [12], the quadratic Laplacian regularization is introduced based on graph representation. In [13], authors present a spatial spectral coherence regularization that relates abundance estimation of a pixel to that of its neighboring pixels with spectral similarities. In [14], authors perform the unmixing with low-rank spatial regularization within fixed-size square windows.

However, it is necessary to establish a frame for these various ways of regularization via a proper mathematical tool. A graph is a ubiquitous structure that describes the connection relationship of a set of vertices. Graph theory is actively used in fields such as biochemistry (genomics), electrical engineering (communication networks and coding theory), computer science (algorithms and computation), and operations research (scheduling) [15]. In addition, several works apply graph theoretical techniques to hyperspectral images, including methods for dimensionality reduction [16], anomaly detection [17], and classification [18]. In the context of hyperspectral data unmixing, a graph can be used to model relations of spatial and spectral information of hyperspectral pixels. In this chapter, we will present a variety of ways to construct a graph for hyperspectral unmixing and formulate the associated unmixing problem with solutions given by the alternating direction method of multipliers (ADMM) strategy.

The remainder of the chapter is organized as follows: Section 2 introduces graph theory and graph construction methods in the context of hyperspectral unmixing. Section 3 formulates the sparse linear unmixing problem based on graph regularization. In Section 4, the solution to the formulated problem is derived via the ADMM algorithm. Section 5 reports the experiment's results. Finally, Section 6 concludes this chapter.

http://dx.doi.org/10.5772/intechopen.73158

2. Graph construction

2.1. Introduction to graphs

We firstly review some fundamental elements of a graph. A graph is a general data structure described by $G = (V, E)$, where a finite set of vertices, also called nodes, is denoted by V and a finite set of pairs of the form $(v_i v_j)$ is referred to as edges. Edges indicate the relation between vertices, and they can be directed or undirected. Directed edges utilize ordered pairs of points that indicate the source and sink of each connection, that is, $(v_i v_j)$ represents an edge from v_i to v_j. Undirected edges only indicate the relationship between vertices and do not consider the ordering, that is, $(v_i v_j)$ is the same as $(v_j v_i)$. We may associate each edge with a weight to describe the importance or the cost of this connection (**Figure 1**).

In a simple setting, if two vertices are connected by an edge, the weight is set to 1, otherwise the weight is 0. The following part introduces some other ways to measure the similarity among vertices, in other words, to define the weights. We can use either adjacency matrices or incidence matrices to describe graphs depending on the type of operations to be performed. Elements of the matrix **A** indicate whether pairs of vertices are connected or not in a graph. Element A_{ij} is 1 when there is an edge from vertex i to vertex j and zero when there is no edge. If the graph is undirected, the adjacency matrix is symmetric. Incidence matrices show the relationship between vertices and edges. An undirected graph can have two kinds of incidence matrices: unoriented and oriented matrices. An oriented incidence matrix in the undirected graph can be denoted by $\mathbf{B} \in \mathbb{R}^{n \times m}$, where n is the number of vertices and m is the number of edges. That is, in the column of edge e_k, the positive undirected graph can be denoted by $\mathbf{B} \in \mathbb{R}^{n \times m}$, where n is the number of vertices and m is the number of edges. That is, in the column of edge e_k, there is positive weight A_{ij} in the row corresponding to one vertex v_i of e_k and negative weight -A_{ij} in the row corresponding to the other vertex v_j of e_k, and all other rows are set to 0.

In addition, a degree matrix for a graph is a diagonal matrix $\mathbf{D} = \mathrm{diag}(d_1, \cdots d_i, \cdots d_n)$, where n is the number of vertices and d_i is the degree of the vertex v_i in G. The degree matrix is indicating every vertex's degree which is the number of edges connecting to one vertex. It is

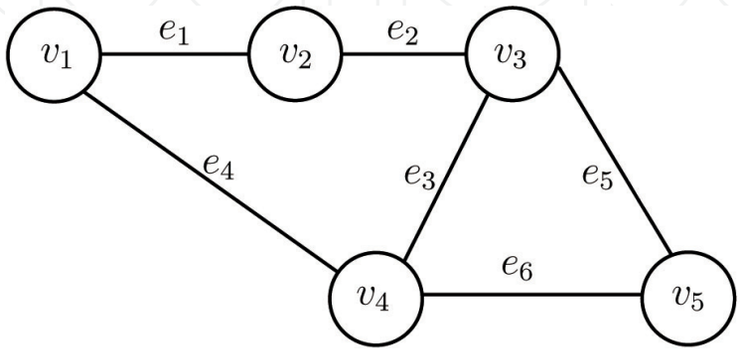


Figure 1. Example of a graph.

normally used together with the adjacency matrix to construct the Laplacian matrix **L** of a graph, which is $\mathbf{L} = \mathbf{D} - \mathbf{A}$.

2.2. Graph construction for hyperspectral images

In this part, we elaborate the ways to construct graphs in the context of hyperspectral image analysis. The performance of spectral unmixing is closely tied to the graph construction of images, and in the hyperspectral remote sensing literature, there are a number of techniques. In [19], authors summarize a survey of spectral graph construction techniques and discuss advantages and disadvantages of these techniques. Generally, each pixel can be viewed as a vertex (or node), and each vertex is associated with a continuous spectrum. A set of edges can be set and assigned with weights in different senses as presented here below.

2.2.1. Four-neighbor graph

A common and straightforward construction is to consider the four-neighbor graph, where every vertex (i.e., every pixel) is connected to four nearest spatially adjacent neighboring pixels, as illustrated in **Figure 2**.

2.2.2. Threshold-compared graph

Another alternative to construct a graph is to calculate all pairwise distances and an edge is placed if the distance between two vertices is less than a user-predefined threshold. The distance in the hyperspectral image can be measured using the spectral distance or spatial distance. For instance, $\boldsymbol{v}_i$ and $\boldsymbol{v}_j$ are two vertices that are associated with spectral vectors with L bands, then their Euclidean distance is $\left\|\boldsymbol{v}_i - \boldsymbol{v}_j\right\| = \sqrt{\sum_{k=1}^{L}\left(v_{ik} - v_{jk}\right)^2}$.

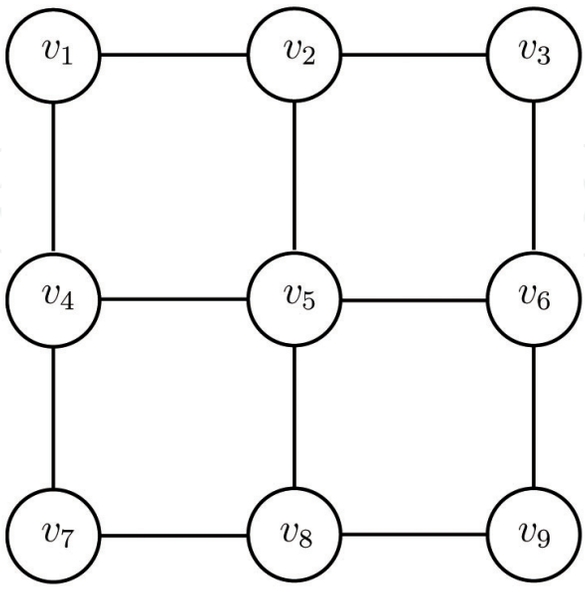


Figure 2. Spatial four-neighbor method.

http://dx.doi.org/10.5772/intechopen.73158

2.2.3. K-nearest neighbor graph

Constructing a graph with *k*-nearest neighbors (k-NN) is a popular method. In this case, an edge is set between two vertices if vertex v_j is in *k*-NN of vertex v_i. Each vertex has its own *k*-nearest neighbors. Consequently, the graph is a directed graph. It is worth noting that constructing such a graph requires calculating all pairwise distances and ordering these values on each vertex, and these operations lead to high computational costs.

2.2.4. Spatial-spectral graph

As pixels in a hyperspectral image possess spatial locations and spectral signatures, it can be beneficial to construct a graph by incorporating both spatial and spectral information. For instance, a graph can be constructed with local four neighborhood pixels and by considering spectral similarity among pixels, as described in **Figure 3**.

2.2.5. Weighted graph

Above methods construct unweighted graphs with only connection indications among pixels. Several other methods further impose weights on each edge. For instance, spectral similarity measured by the Gaussian kernel can be used to define weights:

$$A_{ij} = \exp\left(-\frac{\|v_i - v_j\|^2}{2\sigma^2}\right) \tag{1}$$

where σ is the kernel bandwidth and defined by users. As a generalization, a radial basis function (also called a diffusion kernel) in spectral distance with two parameters σ_i and σ_j is introduced in [20], given by:

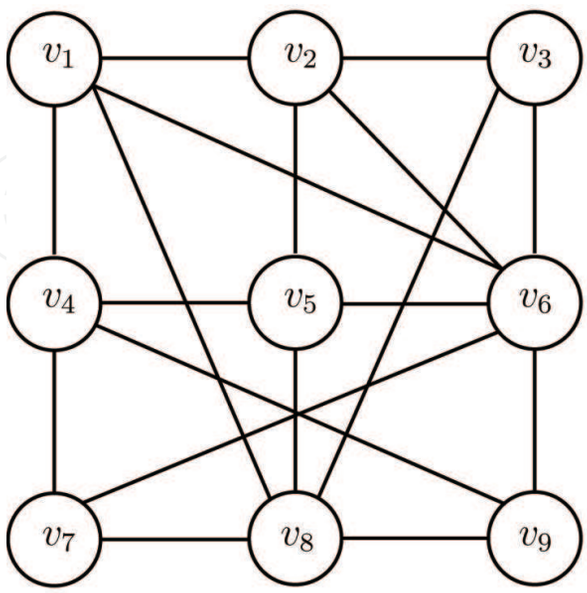


Figure 3. An example of four spatial neighbors and *k*-NN spectral neighbors.

$$A_{ij} = \exp\left(-\frac{\left\|\boldsymbol{v_i} - \boldsymbol{v_j}\right\|^2}{\sigma_i \sigma_j}\right). \tag{2}$$

Weights can also be calculated by considering both spatial and spectral information. For instance, [21] proposes to define weights by:

$$A_{ij} = \exp\left(-\frac{\left\|\boldsymbol{v_i} - \boldsymbol{v_j}\right\|^2}{c_{ij}\sigma_i \sigma_j}\right) \cdot \exp\left(-\frac{\left\|\boldsymbol{x_i} - \boldsymbol{x_j}\right\|^2}{\sigma_d^2}\right) \tag{3}$$

where $\boldsymbol{x_i}$ is the spatial coordinates of pixel $\boldsymbol{v_i}$, c_{ij} is an integer indicating the number of common neighbors between $\boldsymbol{v_i}$ and $\boldsymbol{v_j}$, σ_i and σ_j are defined in [20], and the parameter σ_d is defined by users which limits the size of regions spatially. In [22], authors consider the similarity of the spectral angle instead of the spectral Euclidean distance.

$$A_{ij} = \exp\left(-\theta_{ij}\right) \cdot \exp\left(-\frac{\left\|\boldsymbol{x_i} - \boldsymbol{x_j}\right\|^2}{\sigma}\right) \tag{4}$$

where θ_{ij} denotes the spectral angle between $\boldsymbol{v_i}$ and $\boldsymbol{v_j}$, $\boldsymbol{x_i}$ is the spatial coordinates of pixel $\boldsymbol{v_i}$ and σ is the parameter defined by users. Note that some schemes of calculating weights can make edges to be severed so as to change the structure of the graph [19].

There are also some other methods to construct graphs adapted to hyperspectral images, such as adaptive nearest neighbor graphs, density-weighted *k*-NN graphs, and shared nearest-neighbor graphs [19].

3. Graph-based regularization in unmixing

With a constructed graph at hand to model the relation of pixels, in this section, we present the way to perform a sparse unmixing with the graph regularization.

3.1. Sparse unmixing

Consider the linear mixing model: $\mathbf{y} = \mathbf{S}\mathbf{x} + \mathbf{n}$, where $\mathbf{y} \in \mathbb{R}^L$ is one observed pixel with L spectral bands, $\mathbf{S} = [\mathbf{s}_1, \mathbf{s}_2, \cdots, \mathbf{s}_R] \in \mathbb{R}^{L \times R}$ is the library of spectral signatures including R pure spectral signatures, and $\mathbf{x} \in \mathbb{R}^R$ is an abundance vector, $\mathbf{n}$ is the additive white noise vector. Since it is often that an observed pixel is only composed of a small number of materials in the library, the majority of entries of the abundance vector $\mathbf{x}$ are zero-valued, namely, $\mathbf{x}$ is sparse. Assuming the library is available beforehand and the spare unmixing problem can be defined as [23]:

$$\begin{aligned} &\min_{\mathbf{x}} \ \frac{1}{2}\|\mathbf{y}-\mathbf{S}\mathbf{x}\|_2^2+\lambda\|\mathbf{x}\|_1 \\ &\text{subject to}: \mathbf{x}\geq 0 \end{aligned} \tag{5}$$

where λ is the regularized parameter.

In this chapter, we formulate the problem without ASC constraint because of using the l_1 norm regularization. Moreover, the validity of ASC is often questioned in the literature for practical scenarios. In what follows, we introduce the graph regularization to the above formulated problem.

3.2. Graph regularization for sparse unmixing

Since a graph relates the pixels of image via spatial and spectral relations, we can regularize the unmixing problem with pixel relations defined by the graph. Let $\mathbf{Y}=\left[\mathbf{y}_1,\mathbf{y}_2,\cdots\mathbf{y}_n\right]\in\mathbb{R}^{L\times n}$ be a spectral matrix, where each column is one observed pixel including L spectral bands and n is the number of pixels, and in a graph, n is also the number of vertices. Let $\mathbf{X}=[\mathbf{x}_1,\mathbf{x}_2,\cdots\mathbf{x}_n]\in\mathbb{R}^{R\times n}$ be an abundance matrix in which each column is an abundance vector associated with one observed pixel. With the graph representation of hyperspectral data, we achieve the sparse unmixing by solving the following optimization problem:

$$\begin{aligned} &\min_{\mathbf{x}} \ \frac{1}{2}\|\mathbf{Y}-\mathbf{S}\mathbf{X}\|_F^2+\mu\|\mathbf{X}\|_{1,1}+\lambda_g g_1(\mathbf{X}) \\ &\text{subject to}: \ \mathbf{X}\geq 0 \end{aligned} \tag{6}$$

where

$$g_1(\mathbf{X})=\sum_{i=1}^{n}\sum_{j=1}^{n}A_{ij}\left\|\mathbf{x}_i-\mathbf{x}_j\right\|_1 \tag{7}$$

This graph regularization term Eq. (7) is based on the assumption that if two vertices are connected by an edge, then the abundances of the two vertices are similar. This term measures the differences between all pairs of abundances weighted by their degrees of similarity in the graph. The graph regularization then promotes piecewise constant transitions of estimates among the related pixels. Parameter λ_g controls the regularization strength. Note that we can rewrite Eq. (7) with the incidence matrix $\mathbf{B}$ as:

$$\sum_{i=1}^{n}\sum_{j=1}^{n}A_{ij}\left\|\mathbf{x}_i-\mathbf{x}_j\right\|_1=\|\mathbf{X}\mathbf{B}\|_{1,1} \tag{8}$$

Problem Eq. (6) is equivalently expressed as:

$$\begin{aligned} &\min_{\mathbf{x}} \ \frac{1}{2}\|\mathbf{Y}-\mathbf{S}\mathbf{X}\|_F^2+\mu\|\mathbf{X}\|_{1,1}+\lambda_g\|\mathbf{X}\mathbf{B}\|_{1,1} \\ &\text{subject to}: \ \mathbf{X}\geq 0 \end{aligned} \tag{9}$$

If we use a spatial four-neighborhood graph in this unmixing problem with the weights being simply set to 1 and 0, it can generally be identical with the SUnSAL-TV algorithm [8]. The right term can promote piecewise constant transitions in the fractional abundance among neighborhood pixels and achieve spatial consistency of estimated abundances.

Instead of considering promoting the similarities among estimated abundances, an alternative way is to promote the similarities of reconstructed spectra among the connected pixels. In [24], authors propose a nonlocal TV regularization, with the regularization term given as:

$$g_2(\mathbf{X}) = \sum_{i=1}^{n}\sum_{j=1}^{n} A_{ij}\left\|\mathbf{S}\mathbf{x}_i - \mathbf{S}\mathbf{x}_j\right\|_1 \tag{10}$$

This can also be written with incidence matrix $\mathbf{B}$ as:

$$g_3(\mathbf{X}) = \|(\mathbf{S}\mathbf{X})\mathbf{B}\|_1 \tag{11}$$

4. Solution to the formulated problem

We propose to solve the formulated unmixing problem Eq. (9) via the ADMM algorithm. In this section, we first briefly review the ADMM algorithm and then apply it to our unmixing problem.

4.1. Introduction of ADMM

ADMM is an algorithm that is intended to blend the decomposability of dual ascent with the superior convergence properties of the method of multipliers. The algorithm solves problems in the form [25]:

$$\begin{aligned} &\min f(x) + g(z) \\ &\text{s.t. } Ax + Bz = c \end{aligned} \tag{12}$$

with variables $x \in \mathbb{R}^n$ and $z \in \mathbb{R}^m$, where $A \in \mathbb{R}^{p\times n}$, $B \in \mathbb{R}^{p\times m}$, and $c \in \mathbb{R}^p$.

The first step is to write the augmented Lagrangian of problem Eq. (12):

$$L_\rho(x,z,y) = f(x) + g(z) + y^T(Ax + Bz - c) + \frac{\rho}{2}\|Ax + Bz - c\|_2^2 \tag{13}$$

ADMM suggests achieving the optimum via the following iterations:

$$x^{k+1} = \underset{x}{\operatorname{argmin}}\, L_\rho\left(x, z^k, y^k\right) \tag{14}$$

$$z^{k+1} = \underset{z}{\operatorname{argmin}}\, L_\rho\left(x^{k+1}, z, y^k\right) \tag{15}$$

$$y^{k+1} = y^k + \rho\left(Ax^{k+1} + Bz^{k+1} - c\right) \tag{16}$$

where $\rho > 0$. The algorithm is very similar to dual ascent and the method of multipliers: it consists of an x-minimization step Eq. (14), a z-minimization step Eq. (15), and a dual variable update Eq. (16). As in the method of multipliers, the dual variable update uses a step size equal to the augmented Lagrangian parameter ρ.

4.2. Solutions via ADMM

In order to apply the canonical ADMM algorithm to the problem (9), we introduce the auxiliary variables and transform the problem as follows:

$$\begin{aligned} \underset{\mathbf{X},\,\mathbf{V}_1,\,\mathbf{V}_2,\,\mathbf{Z}}{\text{minimize}}\ & \frac{1}{2}\|\mathbf{Y}-\mathbf{S}\mathbf{X}\|_F^2+\mu\|\mathbf{V}_2\|_{1,1}+\lambda_g\|\mathbf{V}_4\|_{1,1}+l_{\mathbb{R}_+^{R\times n}}(\mathbf{V}_2) \\ \text{subject to}\ & \mathbf{V}_1=\mathbf{S}\mathbf{X} \\ & \mathbf{V}_2=\mathbf{X} \\ & \mathbf{V}_3=\mathbf{X} \\ & \mathbf{V}_4=\mathbf{V}_3\mathbf{B} \end{aligned} \tag{17}$$

where l_S is the indicator function of the set S , such as $l_S(\mathbf{x})=0$ if $\mathbf{x}\in S$ and $l_S(\mathbf{x})=+\infty$ if $\mathbf{x}\notin S$. Thus the augmented Lagrangian for Eq. (17) is as follows:

$$\begin{aligned} L(\mathbf{X},\mathbf{V}_{1\to4},\mathbf{D}_{1\to4}) = & \frac{1}{2}\|\mathbf{Y}-\mathbf{V}_1\|_F^2+\mu\|\mathbf{V}_2\|_{1,1}+\lambda_g\|\mathbf{V}_4\|_{1,1}+l_{\mathbb{R}_+^{R\times n}}(\mathbf{V}_2) \\ & +\frac{\rho}{2}\|\mathbf{S}\mathbf{X}-\mathbf{V}_1-\mathbf{D}_1\|_F^2+\frac{\rho}{2}\|\mathbf{X}-\mathbf{V}_2-\mathbf{D}_2\|_F^2 \\ & +\frac{\rho}{2}\|\mathbf{X}-\mathbf{V}_3-\mathbf{D}_3\|_F^2+\frac{\rho}{2}\|\mathbf{V}_3\mathbf{B}-\mathbf{V}_4-\mathbf{D}_4\|_F^2 \end{aligned} \tag{18}$$

where $\mathbf{D}_1$, $\mathbf{D}_2$, $\mathbf{D}_3$, $\mathbf{D}_4$ are Lagrange multipliers and ρ is the penalty parameter.

The algorithm steps are as follows:

Step 1. Input the observed pixels $\mathbf{Y}$ matrix, the library $\mathbf{S}$, and the regularization parameters μ, λ_g;

Step 2. Initialization: $\mathbf{X}^{(0)}$, $\mathbf{V}_1^{(0)}$, ⋯, $\mathbf{V}_4^{(0)}$, $\mathbf{D}_1^{(0)}$, ⋯, $\mathbf{D}_4^{(0)}$, ρ, set $k=0$

Step 3. Repeat:

Step 4. $\mathbf{X}^{(k+1)} \leftarrow \text{argmin}_{\mathbf{X}} L_\rho\left(\mathbf{X},\mathbf{V}_1^{(k)},\mathbf{V}_2^{(k)},\mathbf{V}_3^{(k)},\mathbf{V}_4^{(k)},\mathbf{D}_1^{(k)},\mathbf{D}_2^{(k)},\mathbf{D}_3^{(k)},\mathbf{D}_4^{(k)}\right)$

Step 5. $\mathbf{V}_1^{(k+1)} \leftarrow \text{argmin}_{\mathbf{V}_1} L_\rho\left(\mathbf{X}^{(k+1)},\mathbf{V}_1,\mathbf{V}_2^{(k)},\mathbf{V}_3^{(k)},\mathbf{V}_4^{(k)},\mathbf{D}_1^{(k)},\mathbf{D}_2^{(k)},\mathbf{D}_3^{(k)},\mathbf{D}_4^{(k)}\right)$

Step 6. $\mathbf{V}_2^{(k+1)} \leftarrow \text{argmin}_{\mathbf{V}_2} L_\rho\left(\mathbf{X}^{(k+1)},\mathbf{V}_1^{(k+1)},\mathbf{V}_2,\mathbf{V}_3^{(k)},\mathbf{V}_4^{(k)},\mathbf{D}_1^{(k)},\mathbf{D}_2^{(k)},\mathbf{D}_3^{(k)},\mathbf{D}_4^{(k)}\right)$

Step 7. $\mathbf{V}_3^{(k+1)} \leftarrow \text{argmin}_{\mathbf{V}_3} L_\rho\left(\mathbf{X}^{(k+1)},\mathbf{V}_1^{(k+1)},\mathbf{V}_2^{(k+1)},\mathbf{V}_3,\mathbf{V}_4^{(k)},\mathbf{D}_1^{(k)},\mathbf{D}_2^{(k)},\mathbf{D}_3^{(k)},\mathbf{D}_4^{(k)}\right)$

Step 8. $\mathbf{V}_4^{(k+1)} \leftarrow \text{argmin}_{\mathbf{V}_4} L_\rho\left(\mathbf{X}^{(k+1)},\mathbf{V}_1^{(k+1)},\mathbf{V}_2^{(k+1)},\mathbf{V}_3^{(k+1)},\mathbf{V}_4,\mathbf{D}_1^{(k)},\mathbf{D}_2^{(k)},\mathbf{D}_3^{(k)},\mathbf{D}_4^{(k)}\right)$

Step 9. Update the Lagrangian multipliers:

$$\begin{aligned}
\mathbf{D}_1^{(k+1)} &\leftarrow \mathbf{D}_1^{(k)} - \left(\mathbf{S}\mathbf{X}^{(\mathrm{k}+1)} - \mathbf{V}_1^{(\mathrm{k}+1)}\right) \\
\mathbf{D}_2^{(k+1)} &\leftarrow \mathbf{D}_2^{(k)} - \left(\mathbf{X}^{(\mathrm{k}+1)} - \mathbf{V}_2^{(\mathrm{k}+1)}\right) \\
\mathbf{D}_3^{(k+1)} &\leftarrow \mathbf{D}_3^{(k)} - \left(\mathbf{X}^{(\mathrm{k}+1)} - \mathbf{V}_3^{(\mathrm{k}+1)}\right) \\
\mathbf{D}_4^{(k+1)} &\leftarrow \mathbf{D}_4^{(k)} - \left(\mathbf{V}_3^{(\mathrm{k}+1)}\mathbf{B} - \mathbf{V}_4^{(\mathrm{k}+1)}\right)
\end{aligned}$$

Step 10. Until stopping criterion is satisfied.

In step 4 of minimizing the augmented Lagrangian with respect to $\mathbf{X}$, the solution is:

$$\mathbf{X} \leftarrow \left(\mathbf{S}^T\mathbf{S} + 2\mathbf{I}\right)^{-1}\left(\mathbf{S}^T(\mathbf{V}_1 + \mathbf{D}_1) + \mathbf{V}_2 + \mathbf{D}_2 + \mathbf{V}_3 + \mathbf{D}_3\right) \tag{19}$$

Similarly, the solution of $\mathbf{V}_1$ minimization step 5 is:

$$\mathbf{V}_1 \leftarrow \frac{1}{1+\rho}\left(\mathbf{Y} + \rho(\mathbf{S}\mathbf{X} - \mathbf{D}_1)\right) \tag{20}$$

To compute $\mathbf{V}_2$ in step 6, the solution is the well-known soft threshold [17]:

$$\mathbf{v}_2 \leftarrow \max\left(soft\left(\mathbf{x} - \mathbf{d}_2, \frac{\mu}{\rho}\right), 0\right) \tag{21}$$

where $\mathbf{v}_2$, $\mathbf{x}$, $\mathbf{d}_2$ is the row of $\mathbf{V}_2$, $\mathbf{X}$, $\mathbf{D}_2$, respectively.

The solution of $\mathbf{V}_3$ minimization step 7 is:

$$\mathbf{V}_3 \leftarrow \left(\mathbf{X} - \mathbf{D}_3 + (\mathbf{V}_4 + \mathbf{D}_4)\mathbf{B}^T\right)\left(\mathbf{I} + \mathbf{B}\mathbf{B}^T\right)^{-1} \tag{22}$$

The solution of $\mathbf{V}_4$ minimization step 8 is:

$$\mathbf{v}_4 \leftarrow soft\left(\mathbf{f} - \mathbf{d}_4, \frac{\lambda}{\rho}\right) \tag{23}$$

where $\mathbf{v}_4$, $\mathbf{f}$, $\mathbf{d}_4$ is the row of $\mathbf{V}_4$, $\mathbf{F} = \mathbf{V}_3 \times \mathbf{B}$, $\mathbf{D}_4$, respectively.

5. Experiments

In this section, we illustrate the experimental results via a synthetic hyperspectral data set (denoted by Data 1) and a real hyperspectral data set (denoted by Data 2) with various ways of graph construction.

5.1. Experiments with simulated data sets

In this part, the synthetic data consists of 75×75 pixels and is generated by 9 endmembers. The endmembers are randomly selected from the spectral library advanced spaceborne thermal

http://dx.doi.org/10.5772/intechopen.73158

emission and reflection radiometer (ASTER). Each signature of endmembers has reflectance values measured over 420 spectral bands. The pure regions and mixed regions involved between two and five endmembers, spatially distributed in the form of square regions. The background is a mixture of the five endmembers with the abundance values $[0, 0, 0, 0, 0.1149, 0.0741, 0.2003, 0.2055, 0.4051]$.

The quality of unmixing results for the simulated data can be measured by comparing the estimated and actual abundances using the root mean square error (RMSE),

	15 dB	**20 dB**	**30 dB**
Four-neighbor graph	0.0246 $\mu = 0.005, \lambda = 0.05$	0.0184 $\mu = 0.005, \lambda = 0.05$	0.0051 $\mu = 5 \times 10\text{^}(-4), \lambda = 0.01$
Spectral-spatial combined graph	0.0085 k = 25; $\mu = 5 \times 10\text{^}(-4), \lambda = 0.01$	0.0052 k = 25 $\mu = 0.005, \lambda = 0.01$	0.0021 k = 25 $\mu = 5 \times 10\text{^}(-4), \lambda = 0.005$
Threshold-compared graph	0.0025 threshold = 9 $\mu = 5 \times 10\text{^}(-4), \lambda = 0.1$	0.0015 threshold = 3 $\mu = 5 \times 10\text{^}(-4), \lambda = 0.5$	0.0009 threshold = 0.25 $\mu = 5 \times 10\text{^}(-4), \lambda = 0.1$

Table 1. RMSE evaluating performances with different values of SNR, with three constructed graphs and optimal regularized parameters. The values of threshold and *k* are also shown.

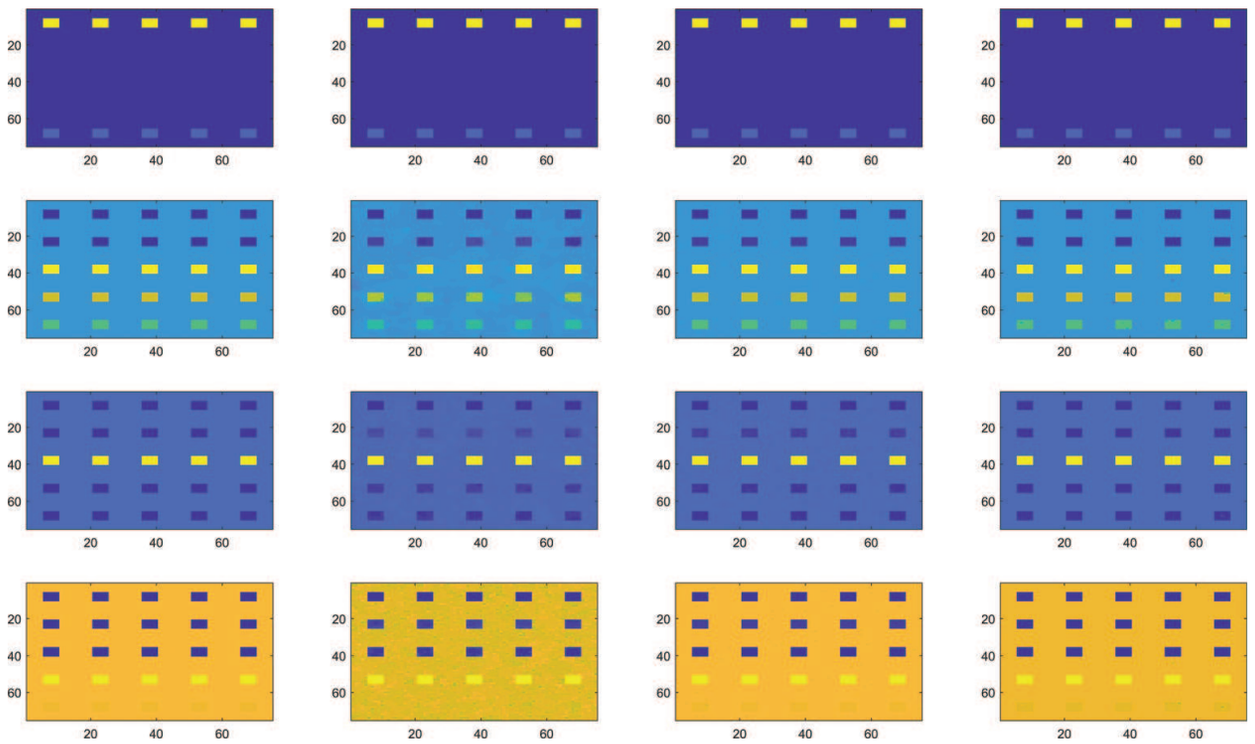


Figure 4. From top to bottom: The abundance maps of first, fifth, sixth, and eighth . From left to right: Real abundance maps, estimated abundance maps with four-neighbor graph, spectral-spatial graph and threshold-compared graph, respectively.

$$RMSE = \sqrt{\frac{1}{nR}\sum_{i=1}^{n}\|\mathbf{x}_i - \widehat{\mathbf{x}}_i\|^2} \tag{24}$$

where $\mathbf{x}_i$ and $\widehat{\mathbf{x}}_i$ are the actual and estimated abundance vectors of the ith pixel, n is the number of pixels, and R is the number of endmembers.

We define the graph based on the simulated data using three methods: the four-neighbor graph, the threshold-compared graph and the spectral-spatial graph respectively.

In the experiment, the threshold-compared undirected graph is constructed as follows:

$$\mathbf{A}_{ij} = \begin{cases} 1 & \text{if} \left\|\mathbf{y}_i - \mathbf{y}_j\right\|_2^2 < threshold \\ 0 & \text{otherwise} \end{cases} \tag{25}$$

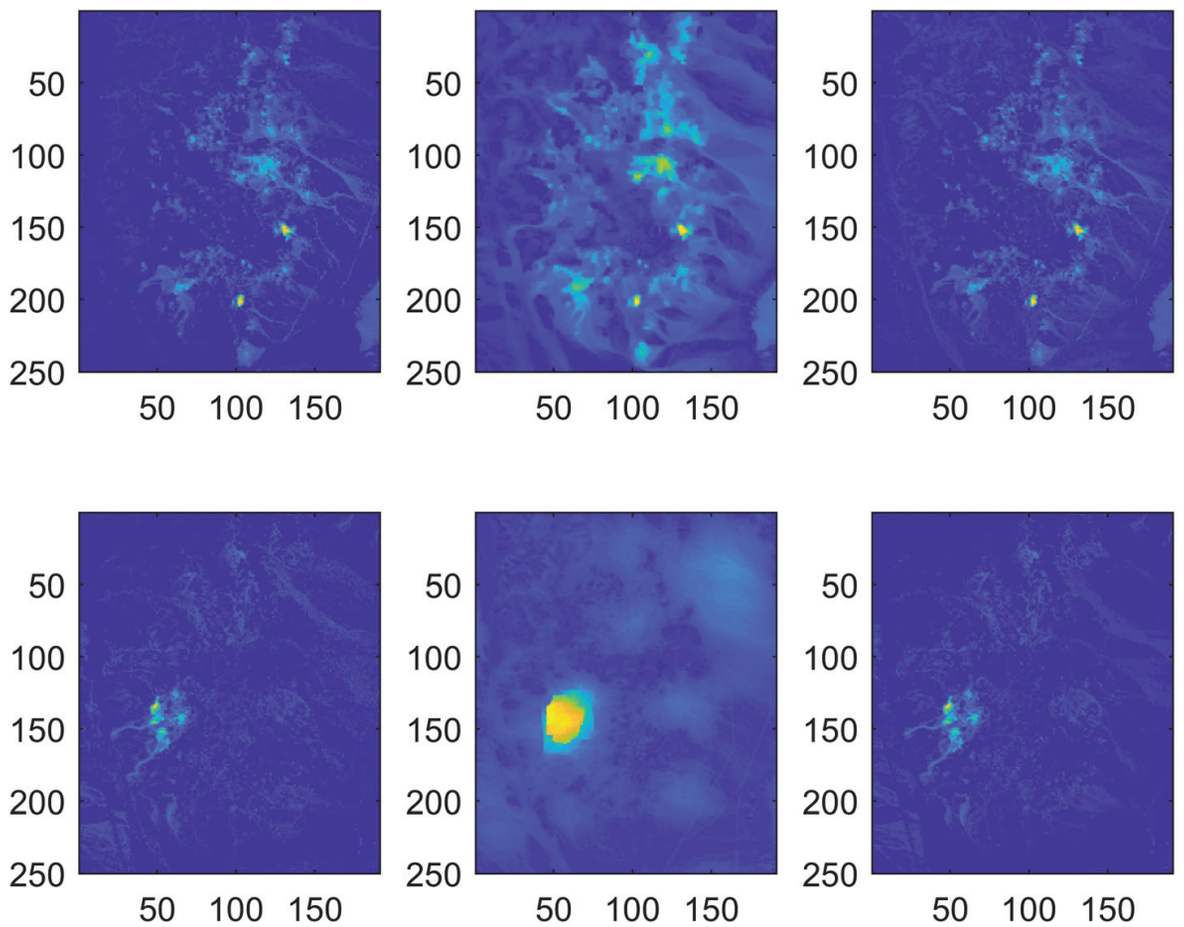


Figure 5. From top to bottom: The abundance maps of first and fifth. From left to right: FCLS, SUnSAL-TV, and the proposed algorithm with the threshold-compared graph.

where all pairs of spectral distance are compared with a user-defined threshold. Meanwhile, the spectral-spatial graph is constructed by considering four neighbors of spatial location and k-nearest neighbors of spectral distance.

From this table, we can see that the performance of the proposed algorithm with a threshold-compared graph is better than the others. Although the second graph in **Table 1** combines the spectral and spatial information, using spatial relation is not always a good way to connect pixels because it is possible that two adjacent pixels may have significantly different spectral features. **Figure 4** shows the true abundance maps and the abundances estimated by the proposed algorithm associated with the three constructed graphs. We observe that the second row of the square regions is better conserved with the proposed algorithm using the threshold-compared graph.

5.2. Experiments with AVIRIS data

We also tested algorithms with a real hyperspectral image. The image is captured on the Cuprite mining district by AVIRIS. A sub-image of 250×191 pixels was chosen, and it contains 188 spectral bands. The number of endmembers was estimated and set to 12 [26]. VCA algorithm was then used to extract the endmembers. Here, we compare the FCLS [9], SUnSAL-TV, and the proposed algorithm with a threshold-compared graph. **Figure 5** shows the first and fifth abundance maps of three algorithms respectively. We can see that the proposed algorithm highlights localized targets without oversmoothing the image like in SUnSAL-TV and with less impurity than in FCLS.

6. Conclusion

In this chapter, we propose to use graph as a mathematical tool to relate pixels in hyperspectral data. We the present a variety of methods of constructing a graph according to spatial information and spectral information embedded in an image. A sparse unmixing problem is then formulated with the graph regularization to enhance the estimation performance. An ADMM-based algorithm is then presented to solve the formulated problem. In the experiments, we compare the unmixing performance of the presented unmixing algorithm with different graphs, using a synthetic hyperspectral data and a real data. Future works include evaluating the unmixing performance with weighted graphs.

Acknowledgements

This work was supported in part by National Natural Science Foundation of China under grant 61671382 and in part by Natural Science Foundation of Shenzhen under grant JCYJ2017030155315873.

Author details

Zeng Li[1,2], Jie Chen[1,2]* and Susanto Rahardja[1]

*Address all correspondence to: dr.jie.chen@ieee.org

1 Centre of Intelligent Acoustics and Immersive Communications, School of Marine Science and Technology, Northwestern Polytechnical University, China

2 Research and Development Institute of Northwestern Polytechnical University in Shenzhen, China

References

[1] Keshava N, Mustard JF. Spectral unmixing. IEEE Signal Processing Maganize. 2002;**19**(1): 44-57

[2] Dobigeon N, Tourneret J-Y, Richard C, Bermudez JCM, McLaughlin S, Hero AO. Nonlinear unmixing of hyperspectral images: Models and algorithms. IEEE Signal Processing Magazine. 2014;**31**(1):82-94

[3] Singer RB, McCord TB. Mars-large scale mixing of bright and dark surface materials and implications for analysis of spectral reflectance. In: Lunar and Planetary Science Conference Proceedings; 1979. p. 1835-1848

[4] Hapke B. Bidirectional reflectance spectroscopy: 1. Theory. Journal of Geophysical Research: Solid Earth. 1981;**86**:3039-3054

[5] Halimi A, Altmann Y, Dobigeon N, Tourneret J-Y. Nonlinear unmixing of hyperspectral images using a generalized bilinear model. IEEE Transactions on Geoscience and Remote Sensing. 2011;**49**(11):4153-4162

[6] Meganem I, Deliot P, Briottet X, Deville Y, Hosseini S. Linear–quadratic mixing model for reflectances in urban environments. IEEE Transactions on Geoscience and Remote Sensing. 2014;**52**(1):544-558

[7] Altmann Y, Dobigeon N, Tourneret J-Y. Nonlinearity detection in hyperspectral images using a polynomial post-nonlinear mixing model. IEEE Transactions on Image Processing. 2013;**22**(4):1267-1276

[8] Iordache M-D, Bioucas-Dias JM, Plaza A. Total variation spatial regularization for sparse hyperspectral unmixing. IEEE Transactions on Geoscience and Remote Sensing. 2012; **50**(11):4484-4502

[9] Heinz DC, Chang C-I. Fully constrained least squares linear spectral mixture analysis method for material quantification in hyperspectral imagery. IEEE transactions on geoscience and remote sensing. 2001;**39**(3):529-545

[10] Dobigeon N, Tourneret J-Y, Chang C-I. Semi-supervised linear spectral unmixing using a hierarchical Bayesian model for hyperspectral imagery. IEEE Transactions on Signal Processing. 2008;**56**(7):2684-2695

[11] Chang C-I, Wu C-C, Liu W, Ouyang Y-C. A new growing method for simplex-based endmember extraction algorithm. IEEE Transactions on Geoscience and Remote Sensing. 2006;**44**(10):2804-2819

[12] Ammanouil R, Ferrari A, Richard C. A graph laplacian regularization for hyperspectral data unmixing. In: Acoustics, Speech and Signal Processing (ICASSP), 2015 IEEE International Conference; 2015. p. 1637-1641

[13] Castrodad A, Xing Z, Greer JB, Bosch E, Carin L, Sapiro G. Learning discriminative sparse representations for modeling, source separation, and mapping of hyperspectral imagery. IEEE Transactions on Geoscience and Remote Sensing. 2011;**49**(11):4263-4281

[14] Qu Q, Nasrabadi NM, Tran TD. Abundance estimation for bilinear mixture models via joint sparse and low-rank representation. IEEE Transactions on Geoscience and Remote Sensing. 2014;**52**(7):4404-4423

[15] Pirzada S. Applications of graph theory. Journal of the Korean Society for Industrial and. Applied Mathematics. 2007;**7**(1)

[16] Bachmann CM, Ainsworth TL, Fusina RA. Exploiting manifold geometry in hyperspectral imagery. IEEE transactions on Geoscience and Remote Sensing. 2005;**43**(3):441-454

[17] Basener B, Ientilucci EJ, Messinger DW. Anomaly detection using topology. Proceedings of SPIE. 2007

[18] Camps-Valls G, Bandos TV, Zhou D. Semi-supervised graph-based hyperspectral image classification. IEEE Transactions on Geoscience and Remote Sensing. 2007;**45**(10):3044-3054

[19] Stevens JR, Resmini RG, Messinger DW. Spectral-density-based graph construction techniques for Hyperspectral image analysis. IEEE Transactions on Geoscience and Remote Sensing. 2017;**55**(10):5966-5983

[20] Zelnik-Manor L, Perona P. Self-tuning spectral clustering. In: Advances in Neural Information Processing Systems. 2005. pp. 1601-1608

[21] Shi J, Malik J. Normalized cuts and image segmentation. IEEE Transactions on pattern analysis and machine intelligence. 2000;**22**(8):888-905

[22] Gillis DB, Bowles JH. Hyperspectral image segmentation using spatial-spectral graphs. Proc. SPIE. 2012

[23] Bioucas-Dias JM. Figueiredo MAT. Alternating direction algorithms for constrained sparse regression: Application to hyperspectral unmixing. In: Hyperspectral Image and Signal Processing: Evolution in Remote Sensing (WHISPERS), 2010 2nd Workshop; 2010. p. 1-4

[24] Ammanouil R, Ferrari A, Richard C. Hyperspectral data unmixing with graph-based regularization. Proc. IEEE GRSS Workshop Hyperspectral Image Signal. 2015:1-4

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

[26] Chen J, Richard C, Honeine P. Nonlinear unmixing of hyperspectral data based on a linear-mixture/nonlinear-fluctuation model. IEEE Transactions on Signal Processing. 2013;**61**(2):480-492

IntechOpen

IntechOpen