Title
Graph-Based Active Learning for Nearly Blind Hyperspectral Unmixing

Authors
Chen, Bohan
Lou, Yifei
Bertozzi, Andrea L
et al.

Publication Date
2023

DOI
10.1109/tgrs.2023.3313933

Peer reviewed

Graph-based Active Learning for Nearly Blind Hyperspectral Unmixing

Bohan Chen *Student Member, IEEE* Yifei Lou Andrea L. Bertozzi *Member, IEEE* Jocelyn Chanussot *Fellow, IEEE*

***Abstract*—Hyperspectral unmixing is an effective tool to ascertain the material composition of each pixel in a hyperspectral image with typically hundreds of spectral channels. In this paper, we propose two graph-based semi-supervised unmixing methods. The first one directly applies graph learning to the unmixing problem, while the second one solves an optimization problem that combines the linear unmixing model and a graph-based regularization term. Following a semi-supervised framework, our methods require a very small number of training pixels that can be selected by a graph-based active learning method. We assume to obtain the ground truth information at these selected pixels, which can be either the exact abundance value or the one-hot pseudo label. In practice, the latter is much easier to obtain, which can be achieved by minimally involving a human in the loop. Compared to other popular blind unmixing methods, our methods significantly improve performance with minimal supervision. Specifically, the experiments demonstrate that the proposed methods improve the state-of-the-art blind unmixing approaches by 50% or more using only 0.4% of training pixels.**

***Index Terms*—Hyperspectral Unmixing, Graph Learning, Active Learning, Semi-supervised Learning**

I. Introduction

Data obtained by hyperspectral sensors provide both spatial and spectral representations of a scene. Compared to regular color images, which only have three color channels (Red, Green, and Blue), hyperspectral images often contain hundreds to thousands of spectral channels. However, hyperspectral imaging is limited by its low spatial resolution, and hence hyperspectral unmixing (HSU) is an effective tool to identify the pure materials and estimate the proportions of constituent endmembers at each pixel, also known as the *abundance map*. The spectral signature of a pure material is called *endmember*, which can often be measured under a laboratory setting. Unfortunately, the ground-truth endmember is often unavailable due to its large variability in any real scenario. The blind unmixing process involves the estimation of all the endmembers and the abundance map simultaneously.

BC and ALB are with the Department of Mathematics, University of California, Los Angeles (UCLA), CA, USA. YL is with the University of North Carolina, Chapel Hill, NC, USA. JC is with Univ. Grenoble Alpes, CNRS, Grenoble INP, GIPSA-Lab, 38000 Grenoble, France.

BC is supported by the UC-National Lab In-Residence Graduate Fellowship Grant L21GF3606. YL is partially supported by NSF CAREER Award No. 1846690. ALB is funded by an academic grant from the National Geospatial-Intelligence Agency (Award No. HM0476-21-1-0003) project title Active Learning Methods on Graphs for Image, Video and Multispectral Datasets. Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Geospatial-Intelligence Agency. This work has been partially supported by MIAI@Grenoble Alpes, (ANR-19-P3IA-0003).

A. Literature Review of HSU

In our study, we employ a *linear* mixing model for unmixing, wherein each pixel's spectral measurement is represented as a linear combination of constituent endmembers. Given the physical interpretation of hyperspectral mixing, we impose nonnegativity constraints on both endmembers and the abundance map. Additionally, we apply a sum-to-one constraint, a common practice in HSU, signifying that each pixel's abundance vector resides within the probability simplex. There are some extended linear mixing models that consider endmember-wise scaling factors [1] and the spectral variability [2]. Note that these nonlinear mixing models [3], [4] rely on more complicated assumptions about how light rays interact with endmembers. It is also plausible to remove the sum-to-one constraint when illumination conditions or the topography of the scene change locally [1].

Specifically for blind HSU, it is natural to apply the nonnegative matrix factorization (NMF) [5], [6] that decomposes the data matrix into a product of two matrices with nonnegative entries (one encodes the endmember matrix and the other is the abundance map) [7]–[9]. However, even with the nonnegativity and sum-to-one constraints, blind HSU is a highly ill-posed inverse problem, and hence a variety of regularizations have been proposed to refine the solution space. One classic method is the ℓ_2-norm in fully constrained least squares unmixing (FCLSU) [10]. Furthermore, spatial sparsity of abundances is a reasonable assumption due to the fact that only a few endmembers could appear in a single pixel. Some popular sparsity-promoting regularizations used in HSU include the ℓ_0-norm [11], the ℓ_1-norm [12], the $\ell_{1/2}$-norm [13], and the mixed $\ell_{p,q}$-norm for group sparsity [14]. By treating the abundance map for each material as an image, total variation (TV) regularization [15] has been applied to HSU for spatial continuity and edge preservation. TV-related approaches include sparse unmixing via variable splitting augmented Lagrangian and total variation (SUnSAL-TV) [16], TV with sparse NMF [17], TV with nonnegative tensor factorization [18], and an improved collaborative NMF with TV (ICoNMF-TV) [19]. Recently, TV is reformulated as a quadratic regularization that promotes the minimum volume in the NMF framework, referred to as QMV [20].

Graph-based approaches [21] also play an important role in HSU. TV has been extended from vectors in Euclidean space to signals defined on a graph. For example, the graph TV (gTV) [22] is a special case of the p-Dirichlet form

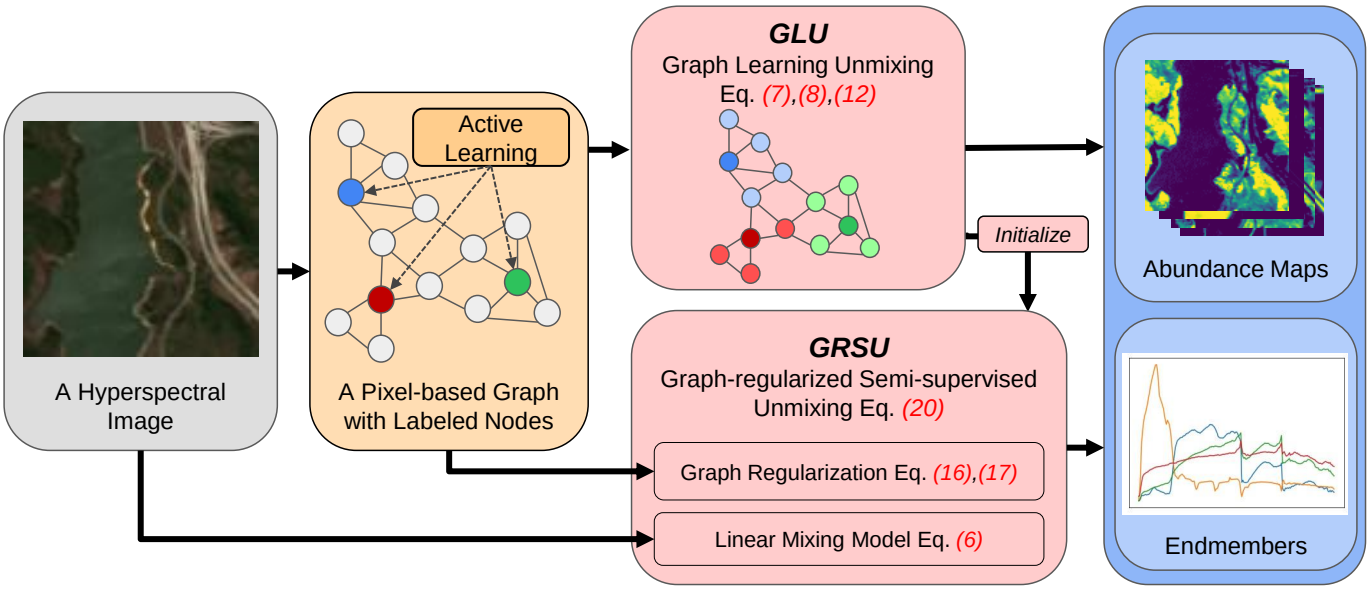


Fig. 1. The flowchart of our semi-supervised hyperspectral unmixing models. The gray box indicates an input hyperspectral image. The orange boxes are the graph construction and graph-based active learning to select labeled nodes (pixels) for the training process (Section II-A,III-A). Two red boxes are our proposed models, Graph Learning Unmixing, GLU, (Section III-B) and Graph-regularized Semi-Supervised Unmixing, GRSU, (Section III-C). GLU applies graph Laplace learning directly to the unmixing task while GRSU combines the graph-based regularization term with the linear unmixing model from hyperspectral imaging into a joint optimization to be solved. GLU also serves as the initialization of the GRSU optimization process. The blue boxes are the outputs of GLU and GRSU, i.e., estimated endmembers and abundance map.

[23], [24] in graph signal processing. Some graph regularization techniques for hyperspectral imaging include structured sparse regularized NMF (SS-NMF) [25] and graph-regularized $\ell_{1/2}$-NMF (GLNMF) [26]. Graph-based approaches, while powerful, can suffer from intensive computation, particularly when computing pairwise similarity between pixels. Strategies in speeding up the weight computation include the use of superpixels [27] rather than using the entire hyperspectral image and the Nyström method [28] to generate low-rank approximations of the graph Laplacian [29], [30]. Another efficient alternative is the use of sparse weight matrices, such as the K-Nearest Neighbors (KNN) weight matrix [31], [32].

In recent years, neural networks have been applied to the blind unmixing problem, such as two-staged self-supervised networks [33], [34], a minimal simplex convolutional neural network [35], a two-stream Siamese deep network [36], and attention networks [37]. Furthermore, some semi-supervised advanced deep learning methods [38], [39] have integrated the use of the graph Laplacian and exhibited remarkable potential in HSU. None of these methods address the issues discussed in the next paragraph.

B. Motivation and Our Contributions

There are limitations of existing blind HSU methods in many real-world scenarios. For example, unmixing often requires estimation of the number of endmembers [40], while the abundance maps require human experts to convey meaning to each endmember. Meanwhile, acquiring the ground-truth abundance maps or endmember spectra is a great challenge [41], [42], making it extremely difficult to train fully supervised models. Such drawback motivates us to consider a semi-supervised model with pseudo labels. These are representative pixels for each endmember, which can be easily obtained by expert's visual inspection of the data. Prior knowledge of the number of endmembers is naturally included in the pseudo labels. We further adopt an active learning [43]–[45] approach to reinforce semi-supervised machine learning methods by carefully automating the selection its training set. Active learning has been successfully applied to hyperspectral image classification and segmentation tasks [46]–[50]. The core of active learning is to sample data points according to an acquisition function [51], [52] that automates the introduction of new training data during the algorithm. We believe active learning and pseudo labels would be an ideal combination to improve the performance of HSU.

The key problem we solve in this paper is to maximize the improvement of our estimated abundance maps and endmember spectra using minimal supervision. With the training pixels selected by active learning, we propose two semi-supervised hyperspectral unmixing models. We refer to these methods as *nearly blind hyperspectral unmixing*, since both of them require a very small number of training labels, and accept either pseudo one-hot labels or ground-truth abundance maps. Our first model, called graph learning unmixing (GLU), takes the output of a graph learning method [21] directly as the abundance maps for the hyperspectral unmixing, followed by the estimation of the endmember matrix. Our second model, called graph-regularized semi-supervised unmixing (GRSU), combines the linear unmixing model, a graph-based regularization term, and a loss function applied to the label set (obtained by active learning) by solving a joint optimization problem. The flowchart of our proposed models (GLU and GRSU) is

illustrated in Figure 1. We summarize the novelties of this work as follows,

- We present an effective pipeline (Section III-A) to select labeled pixels by graph-based active learning for the hyperspectral unmixing problem.
- We apply the idea of graph Laplace learning to the hyperspectral unmixing problem by taking its output (class probability) as the estimated abundance map. Based on this idea, we develop the GLU model (Section III-B).
- We develop a novel semi-supervised hyperspectral unmixing model, GRSU (Section III-C), by combining graph-based regularization terms with the linear mixing model and a small number of labeled pixels.
- Our proposed methods, GLU and GRSU, bear significant practical implications. By utilizing only a small number of easily obtainable pseudo labels, our methods markedly improve the HSU performance.

The rest of the paper is organized as follows. Section II reviews knowledge about graph learning and active learning. Section III introduces our regularized graph Laplace learning together with semi-supervised unmixing algorithms. Section IV shows the experimental results that compare our proposed algorithms and other blind hyperspectral unmixing methods. Finally, Section V concludes the paper.

II. Background and Notation for Graph Active Learning

This section reviews the related works on graph construction, graph learning, and graph-based active learning approaches. These techniques are key ingredients in our proposed semi-supervised unmixing models (detailed in Section III).

A. Graph Construction using KNN Similarity Weight

Given a set of N nodes or vertices $\mathcal{X} = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N\}$ with $\mathbf{x}_i \in \mathbb{R}^{p\times 1}$ for $i = 1, \cdots, N$, we construct a graph $G = (\mathcal{X}, W)$, in which the similarity weight matrix W is computed by K-Nearest Neighbors (KNN). Specifically, we define the angular distance function $d(\mathbf{x}_i, \mathbf{x}_j)$ as the angle between $\mathbf{x}_i$ and $\mathbf{x}_j$, i.e.,

$$d(\mathbf{x}_i, \mathbf{x}_j) = \arccos\left(\frac{\mathbf{x}_i^\top \mathbf{x}_j}{\|\mathbf{x}_i\|_2 \|\mathbf{x}_j\|_2}\right). \tag{1}$$

We only consider edges between $\mathbf{x}_i$ and its K nearest neighbors according to the angular distance $d(\mathbf{x}_i, \cdot)$, which can be implemented by an approximate nearest neighbor search algorithm [31]. For each node $\mathbf{x}_i$, we denote $\{\mathbf{x}_{i_k}\},\ k = 1, 2, \dots, K$ as the K-nearest neighbors of $\mathbf{x}_i$ (including $\mathbf{x}_i$ itself). We define a weight value W_{ij}^{KNN} between nodes $\mathbf{x}_i$ and $\mathbf{x}_j$ as

$$W_{ij}^{\text{KNN}} = \begin{cases} \exp\left(-\frac{d(\mathbf{x}_i,\mathbf{x}_j)^2}{\sigma_i \sigma_j}\right) & j = i_1, i_2, \dots, i_K, \\ 0 & \text{Otherwise}, \end{cases} \tag{2}$$

where $\sigma_i = \sqrt{d(\mathbf{x}_i, \mathbf{x}_{i_K})}$. We define a new weight by $W_{ij} := (W_{ij}^{\text{KNN}} + W_{ji}^{\text{KNN}})/2$ so that the weight matrix W is symmetric. Note that W is sparse and non-negative (i.e. $W_{ij} \geq 0$).

Given the weight matrix W, the degree matrix D is a diagonal matrix, whose diagonal entry is $d_i = \sum_{j=1}^N W_{ij}$ for $i = 1, \cdots, N$. We further define the graph Laplacian matrix [53], [54] by $L = D - W$.

B. Graph Laplace Learning

With the graph $G = (\mathcal{X}, W)$ constructed in Section II-A, we review a graph-based approach for semi-supervised learning, referred to as graph Laplace learning [55]. We suppose ground-truth labels of q classes are available on a subset of vertices $\widehat{\mathcal{X}} \subset \mathcal{X}$. For every $\widehat{\mathbf{x}}_j \in \widehat{\mathcal{X}}$, its label is denoted by $y_j^\dagger \in \{1, 2, \cdots, q\}$ and its corresponding one-hot vector is defined as $\mathbf{y}_j^\dagger = \mathbf{e}_{y_j^\dagger}$, where $\mathbf{e}_k$ is the k^{th} standard basis vector with all zeros except a 1 at the i^{th} entry. The goal for graph learning is to predict the labels of the *unlabeled* set $\mathcal{X} \setminus \widehat{\mathcal{X}}$.

The classification of nodes is based on a node function $\mathbf{u} : \mathcal{X} \to \mathbb{R}^q$. Then the predicted label of $\mathbf{x}_i \in \mathcal{X}$ is $y_i = \arg\max_k \{\mathbf{u}_k(\mathbf{x}_i), k = 1, 2, \dots, q\}$, where $\mathbf{u}_k(\cdot)$ is the k^{th} entry of $\mathbf{u}(\mathbf{x}_i)$. The graph Laplace learning model [55] obtains the optimal node function $\mathbf{u}^*$ by minimizing the following objective function,

$$J_\ell(\mathbf{u}; \mathbf{y}^\dagger) = \frac{1}{4} \sum_{\mathbf{x}_i, \mathbf{x}_j \in \mathcal{X}} W_{ij} \|\mathbf{u}(\mathbf{x}_i) - \mathbf{u}(\mathbf{x}_j)\|_2^2 + \sum_{\widehat{\mathbf{x}}_j \in \widehat{\mathcal{X}}} \ell(\mathbf{u}(\widehat{\mathbf{x}}_j), \mathbf{y}_j^\dagger). \tag{3}$$

The first term of $J_\ell(\mathbf{u}; \mathbf{y}^\dagger)$ in (3) quantifies the smoothness on the node function in the sense that $\mathbf{u}(\mathbf{x}_i)$ is closer to $\mathbf{u}(\mathbf{x}_j)$ for a larger weight W_{ij} between the two nodes. The second term of $J_\ell(\mathbf{u}; \mathbf{y}^\dagger)$ uses the loss function ℓ to measure the difference between the prediction $\mathbf{u}(\widehat{\mathbf{x}}_j)$ and the ground-truth $\mathbf{y}_j^\dagger$ for $\widehat{\mathbf{x}}_j \in \widehat{\mathcal{X}}$. We follow [55] to choose the hard-constraint penalty as the loss function, i.e.,

$$\ell(\mathbf{x}, \mathbf{y}) = \begin{cases} +\infty, & \text{if } \mathbf{x} \neq \mathbf{y}, \\ 0, & \text{if } \mathbf{x} = \mathbf{y}. \end{cases} \tag{4}$$

The penalty function ℓ_h forces the minimizer to take $\mathbf{u}^*(\widehat{\mathbf{x}}_j) = \mathbf{y}_j^\dagger$ for $\widehat{\mathbf{x}}_j \in \widehat{\mathcal{X}}$. The node function $\mathbf{u}$ can be stored by an $N \times q$ matrix U whose ith row represents the output of $\mathbf{u}$ at node i. We can rewrite the first term in (3) as

$$\frac{1}{4} \sum_{\mathbf{x}_i, \mathbf{x}_j \in X} W_{ij} \|\mathbf{u}(\mathbf{x}_i) - \mathbf{u}(\mathbf{x}_j)\|^2 = \frac{1}{2} \langle U, LU \rangle_F, \tag{5}$$

where $\langle \cdot, \cdot \rangle_F$ is the Frobenius inner product for matrices. The matrix form (5) makes it easier to find the minimizer of the objective function $J_\ell(\mathbf{u}; \mathbf{y}^\dagger)$.

C. Graph-based Active Learning

The performance of the graph learning method depends on the selection of the labeled node set $\widehat{\mathcal{X}}$. The classification accuracy varies over different selections under the same number of label nodes. a graph-based active learning approach is an iterative process developed to boost classification accuracy by strategically selecting a series of nodes on a graph $G = (\mathcal{X}, W)$ to label. Specifically, one initializes a label set $\mathcal{X}_L^0$, chosen randomly in practice. At the i^{th} iteration with the current label set $\widehat{\mathcal{X}}^{i-1}$, one classifies the remaining unlabeled nodes $\mathcal{X} \setminus \widehat{\mathcal{X}}^{i-1}$ via a certain graph learning scheme. Using the classification result on the set $X \setminus \widehat{\mathcal{X}}^{i-1}$, one calculates

an acquisition function $\mathcal{A} : \mathcal{X} \setminus \widehat{\mathcal{X}}^{i-1} \to \mathbb{R}$. Then a query set $\mathcal{Q} \subset \mathcal{X} \setminus \widehat{\mathcal{X}}^{i-1}$ is selected to update the current label set $\widehat{\mathcal{X}}^i = \widehat{\mathcal{X}}^{i-1} \cup \mathcal{Q}$.

The sequential active learning selects the query set $\mathcal{Q}$ to be a single node with the largest acquisition function value in $\mathcal{X} \setminus \widehat{\mathcal{X}}$, i.e., $\mathcal{Q} = \{\hat{\mathbf{x}}_{k^*}\}$, $\hat{\mathbf{x}}_{k^*} = \arg\max_{\mathbf{x} \in \mathcal{X} \setminus \widehat{\mathcal{X}}} \mathcal{A}(\mathbf{x})$. The batch active learning selects a query set of multiple unlabeled nodes in graph G. In this paper, we adopt the LocalMax [56] batch active learning approach as it performs similarly to sequential active learning but is more efficient, proportionally to the batch size. It is originally developed for the classification task of the synthetic aperture radar (SAR) images [56] and is further proved successful in the image segmentation task [50]. Specifically, in each iteration of the active learning process, the batch active learning selects the query set $\mathcal{Q}_{\text{LM}}$ with the size $B > 1$, containing the unlabeled nodes that satisfy the local-max condition on the graph G.

This paper uses graph Laplace learning [55] as the underlying graph-based classifier in the active learning process, together with two specific acquisition functions: the Variance Optimality (VOpt) acquisition function [51] and the Model-Change Variance Optimality (MCVOpt) acquisition function [45]. Details about the formulas of these two acquisition functions are provided in Appendix.

III. Semi-supervised Hyperspectral Unmixing

This section details our semi-supervised hyperspectral unmixing methods based on the linear mixing model. Specifically given a hyperspectral data cube I of the dimension $m \times n \times p$ with p spectral channels, we reshape I into a matrix $X = [\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N] \in \mathbb{R}^{p \times N}$, where $N = m \times n$ is the number of pixels. Denote a nonnegative constraint set $\Omega_{p \times q} = \{S \in \mathbb{R}^{p \times q} : S_{ij} \geq 0\}$ and a probability simplex constraint set $\Pi_{q \times N} = \{A \in \Omega_{q \times N} : \mathbf{1}_q^\top A = \mathbf{1}_N^\top\}$. We assume a linear mixing model that generates the data, i.e.,

$$X = SA + E, \tag{6}$$

where $S \in \Omega_{p \times q}$ is the endmember spectrum matrix, $A = [\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_N] \in \Pi_{q \times N}$ is the matrix of abundance maps of q materials, and the matrix $E \in \mathbb{R}^{p \times N}$ denotes a noise term.

In Section III-A, we describe the training data selection process by adapting the graph-based active (Section II-C) to the hyperspectral setting. Then in Section III-B we introduce the graph learning unmixing (GLU) model, which applies graph Laplace learning (Section II-B) directly to the HSU problem. Lastly in Section III-C, we propose our graph-regularized semi-supervised unmixing (GRSU) model that combines graph-based regularization terms with the linear mixing model (6).

A. Training Data Selection

We use the notations X and $\mathcal{X}$ to denote a data matrix and a set of nodes, respectively. Given $\mathcal{X}$, we can construct a graph $G = (\mathcal{X}, W)$ according to Section II-A. Then we apply the graph-based active learning process (Section II-C) to select a set of pixels to acquire labels for both of the proposed models, GLU (Section III-B) and GRSU (Section III-C).

Given two positive integers $m < M$, we begin the active learning process with an initial label set of m random pixels. In each iteration, we apply graph Laplace learning (Section II-B) with the current label set and calculate the acquisition function on the remaining pixels. Based on the values obtained by the acquisition function, we select a query set to be augmented to the label set and terminate this iterative process when the size of the current label set reaches M. Algorithm 1 summarizes the active learning process; and its outputs, the label dataset $\widehat{\mathcal{X}} = \{\widehat{\mathbf{x}}_1, \widehat{\mathbf{x}}_2, \ldots, \widehat{\mathbf{x}}_M\}$ and the set of corresponding labels $\widehat{\mathcal{A}} = \{\widehat{\mathbf{y}}_1, \widehat{\mathbf{y}}_2, \ldots, \widehat{\mathbf{y}}_M\}$, serve as the training data for our semi-supervised unmixing framework.

We want to clarify three aspects of the outputs of the active learning (Algorithm 1). First, one example of acquiring labels in Algorithm 1 is a human-in-the-loop process. Second, the ground-truth "label" $\widehat{\mathbf{y}}_i \in \mathbb{R}^{q \times 1}$ for the "labeled" pixel $\widehat{\mathbf{x}}_i \in \mathbb{R}^{p \times 1}$ can be either the ground-truth abundance vector or its one-hot pseudo label, the latter of which can be determined by the experts for identifying the most significant endmember. According to the experimental results in Section IV, requiring the ground-truth abundance for active learning is not necessary. Third, we adopt the matrix forms of $\widehat{X} = [\widehat{\mathbf{x}}_1, \widehat{\mathbf{x}}_2, \ldots, \widehat{\mathbf{x}}_M] \in \mathbb{R}^{p \times M}$, $\widehat{A} = [\widehat{\mathbf{y}}_1, \widehat{\mathbf{y}}_2, \ldots, \widehat{\mathbf{y}}_M] \in \mathbb{R}^{q \times M}$ of the output sets $\widehat{\mathcal{X}}$ and $\widehat{\mathcal{A}}$, respectively.

Given the training matrix $\widehat{X}$, we estimate the abundance map A for the entire data matrix X. It is true that there is an overlap between $\widehat{X}$ and X, but we cannot completely trust the training labels, especially those obtained by using the one-hot pseudo labels. As a result, we update the abundance map for all the pixels even though a subset of them are selected to acquire some sort of ground-truth information. Another rationale to have two separate matrices X and $\widehat{X}$ is the option to select the training pixels from one image and perform semi-supervised unmixing on the other image, which falls out of the scope of this paper.

B. Graph Learning Unmixing (GLU)

Following the training data selection process, we obtain a training data matrix $\widehat{X} \in \mathbb{R}^{p \times M}$ consisting of M labeled pixels. We concatenate $\widehat{X}$ with the original data matrix X, and construct a graph $\widetilde{G}$ based on the combined data matrix $\widetilde{X} = [\widehat{X}, X]$ with the corresponding graph Laplacian $\widetilde{L}$ according to Section II-A. The graph Laplace learning produces the class probability, which can be regarded as the abundance map. Specifically, we adopt (5) to estimate the abundance map by projecting the graph Laplace learning solution A_{GL} onto $\Pi_{q \times N}$, i.e.,

$$A_{\text{GL}} = \underset{A \in \mathbb{R}^{q \times N}}{\arg\min} \frac{1}{2} \left\langle [\widehat{A}, A]^\top, \widetilde{L}[\widehat{A}, A]^\top \right\rangle_F, \tag{7}$$

$$A_{\text{GLU}} = \mathcal{P}_{\Pi_{q \times N}}(A_{\text{GL}}). \tag{8}$$

Problem (7) is the standard graph Laplace learning. Consider the block representation of $\widetilde{L}$ as

$$\widetilde{L} = \begin{bmatrix} L_{ll} & L_{lu} \\ L_{ul} & L_{uu} \end{bmatrix} \in \mathbb{R}^{(M+N) \times (M+N)}, \tag{9}$$

where $L_{ll} \in \mathbb{R}^{M \times M}$ and $L_{uu} \in \mathbb{R}^{N \times N}$ are the parts of the labeled pixels $\widehat{X}$ and the unlabeled pixels X, respectively, and

Algorithm 1 Sample Labeled Pixels through Active Learning

INPUT: dataset $\mathcal{X}$; corresponding graph $G = (\mathcal{X}, W)$; initial sample number m; total sample number M.
OUTPUT: the label dataset $\widehat{\mathcal{X}} \subset \mathcal{X}$ and the set of corresponding labels $\widehat{\mathcal{A}}$.
INITIALIZE: randomly sample m pixels as the initial label set $\widehat{X}$; acquire the labels for $\widehat{X}$ as $\widehat{A}$.
WHILE $|\widehat{\mathcal{X}}| < M$ ($|\cdot|$ means cardinality) DO:
  Apply the graph Laplace learning on G based on labels of $\widehat{\mathcal{X}}$ to predict labels on $\mathcal{X} \setminus \widehat{\mathcal{X}}$.
  Calculate the acquisition function $\mathcal{A}$ on $\mathcal{X} \setminus \widehat{\mathcal{X}}$ based on the Laplace learning outputs.
  Select a query set $\mathcal{Q}$ based on the acquisition function values according to the sequential active learning or LocalMax.
  Update the current label set: $\widehat{\mathcal{X}} \to \widehat{\mathcal{X}} \cup \mathcal{Q}$; Acquire the labels of $\mathcal{Q}$ and update the label set $\widehat{A}$ accordingly.
END WHILE

$L_{ul} = L_{lu}^\top \in \mathbb{R}^{N\times M}$ represents the cross interaction of graph Laplacian between $\widehat{X}$ and X. Then the minimization problem (7) has a closed-form solution

$$A_{\text{GL}} = -\widehat{A} L_{lu} L_{uu}^{-1}, \tag{10}$$

which can be solved by the preconditioned conjugate gradient method thanks to the symmetric and semi-positive definite properties of the part L_{uu}. Equation (8) is to project the output $A_{\text{GL}} \in \mathbb{R}^{q\times N}$ of the graph Laplace learning onto the set $\Pi_{q\times N}$ by a projection operator $\mathcal{P}_{\Pi_{q\times N}}$, defined by,

$$\mathcal{P}_{\Pi_{q\times N}}(A) = \underset{V\in\Pi_{q\times N}}{\arg\min} \|V - A\|_F. \tag{11}$$

This projection operator can be implemented by a fast algorithm [57].

Given A_{GLU}, we can then find the optimal endmember matrix S_{GLU} that minimizes the combination of the least-squares errors of the linear mixing model (6) and the misfit of the training data, i.e.,

$$S_{\text{GLU}} = \underset{S\in\Omega_{p\times q}}{\arg\min} \frac{1}{2}\|X - SA_{\text{GLU}}\|_F^2 + \frac{\alpha^2}{2}\|\widehat{X} - S\widehat{A}\|_F^2, \tag{12}$$

with a weighting parameter $\alpha > 0$. Equation (12) has a closed-form solution

$$S_{\text{GLU}}^0 = \left(XA^\top + \alpha^2 \widehat{X}\widehat{A}^\top\right)\left(AA^\top + \alpha^2 \widehat{A}\widehat{A}^\top\right)^{-1}, \tag{13}$$

$$S_{\text{GLU}} = \max\left(0, S_{\text{GLU}}^0\right). \tag{14}$$

Equation (14) means to take the entry-wise maximum of S_{GLU}^0 and 0, i.e., replace each negative entry in S_{GLU}^0 by 0. Algorithm 2 presents the pseudo-code of our GLU method, which only involves three steps to find A_{GLU} and S_{GLU} (no iteration is needed).

Algorithm 2 Graph Learning Unmixing (GLU)

INPUT: data matrix X, training data $(\widehat{X}, \widehat{A})$, and $\alpha > 0$.
OUTPUT: matrices S_{GLU} and A_{GLU}.
INITIALIZE: build a graph on $\widetilde{X} = [\widehat{X}, X]$ with corresponding Laplacian matrix $\widetilde{L}$. Segment $\widetilde{L}$ into the block form (9).

1, Graph Learning Step:
  $A_{\text{GL}} = -\widehat{A} L_{lu} L_{uu}^{-1}$.
2, Projection:
  $A_{\text{GLU}} = \mathcal{P}_{\Pi_{q\times N}}(A_{\text{GL}})$.
3, Estimate the endmember spectrum matrix:
  $S_{\text{GLU}}^0 = \left(XA^\top + \alpha^2 \widehat{X}\widehat{A}^\top\right)\left(AA^\top + \alpha^2 \widehat{A}\widehat{A}^\top\right)^{-1}$,
  $S_{\text{GLU}} = \max\left(0, S_{\text{GLU}}^0\right)$.

C. Graph-regularized Semi-supervised Unmixing (GRSU)

By assuming the linear mixing model (6), it is standard to solve the blind unmixing problem in a regularized least square form,

$$\underset{\substack{S\in\Omega_{p\times q},\\ A\in\Pi_{q\times N}}}{\arg\min} \frac{1}{2}\|X - SA\|_F^2 + \lambda \mathcal{J}(A), \tag{15}$$

with a positive weighting parameter λ. The term $\frac{1}{2}\|X - SA\|_F^2$ is a least-squares misfit between the matrix product SA and the data measurement X, while $\mathcal{J}(A)$ is a regularization term of the abundance matrix A. We impose the nonnegative constraints of S and A as well as a sum-to-one constraint of A.

Following the graph Dirichlet energy [58], [59], we consider a graph Laplacian regularization, formulated by

$$\mathcal{J}_1(A) = \frac{1}{4}\sum_{i,j=1}^{N} \|\mathbf{a}_i - \mathbf{a}_j\|_2^2 w(\mathbf{x}_i, \mathbf{x}_j), \tag{16}$$

where the weight function $w : \mathbb{R}^p \times \mathbb{R}^p \to \mathbb{R}$ is defined by

$$w(\mathbf{x}_i, \mathbf{x}_j) = \exp\left(-\frac{d(\mathbf{x}_i, \mathbf{x}_j)^2}{\sigma_i \sigma_j}\right),$$

with the angular distance $d(\mathbf{x}_i, \mathbf{x}_j)$ between node $\mathbf{x}_i$ and $\mathbf{x}_j$ and σ_i, σ_j defined in (1) and (2), respectively.

In addition, we further develop a semi-supervised graph regularization term $\mathcal{J}_2(A, \widehat{A}, \widehat{X})$ that includes the label information $\widehat{X}, \widehat{A}$, that is,

$$\mathcal{J}_2(A; \widehat{A}, \widehat{X}) = \frac{1}{4}\sum_{i=1}^{N}\sum_{j=1}^{M} \|\mathbf{a}_i - \widehat{\mathbf{y}}_j\|_2^2 w(\mathbf{x}_i, \widehat{\mathbf{x}}_j). \tag{17}$$

We consider the sum of both terms $\mathcal{J}_1(A)$ and $\mathcal{J}_2(A; \widehat{A}, \widehat{X})$ as the regularization $\mathcal{J}(A)$. As $\mathcal{J}_1$ and $\mathcal{J}_2$ have the same form that leads to the same scale, we assign the equal weight of them when formulating $\mathcal{J}$, i.e., $\mathcal{J}(A) = \mathcal{J}_1(A) + \mathcal{J}_2(A; \widehat{A}, \widehat{X})$.

Putting (15)–(17) together with the label information $(\widehat{X}, \widehat{A})$ obtained by the active learning approach, we propose a graph-regularized semi-supervised unmixing (GRSU) model to si-

multaneously estimate the abundance map A_{GRSU} and the endmember matrix S_{GRSU}, that is,

$$S_{\text{GRSU}}, A_{\text{GRSU}} = \underset{\substack{S\in\Omega_{p\times q},\\ A\in\Pi_{q\times N}}}{\arg\min} \frac{1}{2}\|X - SA\|_F^2 + \frac{\alpha^2}{2}\|\widehat{X} - S\widehat{A}\|_F^2 + \lambda\mathcal{J}_1(A) + \lambda\mathcal{J}_2(A;\widehat{A},\widehat{X}), \quad (18)$$

with two positive parameters α and λ. We define the indicator function

$$\chi_\Delta(Z) = \begin{cases} 0, & Z \in \Delta, \\ \infty, & \text{otherwise}, \end{cases} \quad (19)$$

to rewrite the minimization problem (18) into an unconstrained formulation as follows,

$$\min_{A,S} \frac{1}{2}\|X - SA\|_F^2 + \frac{\alpha^2}{2}\|\widehat{X} - S\widehat{A}\|_F^2 + \lambda\mathcal{J}_1(A) + \lambda\mathcal{J}_2(A;\widehat{A},\widehat{X}) + \chi_{\Omega_{p\times q}}(S) + \chi_{\Pi_{q\times N}}(A). \quad (20)$$

We apply the alternating direction method of multipliers (ADMM) [60] to solve the unconstrained problem (20). In particular, we introduce two auxiliary variables $T \in \mathbb{R}^{p\times q}$ and $B \in \mathbb{R}^{q\times N}$ to express the problem (20) equivalently as

$$\begin{aligned} \min_{A,B,S,T} \ & \frac{1}{2}\|X - TA\|_F^2 + \frac{\alpha^2}{2}\|\widehat{X} - T\widehat{A}\|_F^2 + \lambda\mathcal{J}_1(B) \\ & + \lambda\mathcal{J}_2(B;\widehat{A},\widehat{X}) + \chi_{\Omega_{p\times q}}(S) + \chi_{\Pi_{q\times N}}(A) \\ \text{s.t.} \quad & A = B, S = T. \end{aligned} \quad (21)$$

The augmented Lagrangian of (21) is written as

$$\mathcal{L} = \frac{1}{2}\|X - TA\|_F^2 + \frac{\alpha^2}{2}\|\widehat{X} - T\widehat{A}\|_F^2 + \lambda\mathcal{J}_1(B) + \lambda\mathcal{J}_2(B;\widehat{A},\widehat{X}) + \chi_{\Omega_{p\times q}}(S) + \chi_{\Pi_{q\times N}}(A) + \frac{\rho}{2}\|A - B + \bar{B}\|_F^2 + \frac{\gamma}{2}\|S - T + \bar{T}\|_F^2,$$

with dual variables $\bar{B}, \bar{T}$ and two positive constants ρ, γ. One benefit of ADMM is that it turns the joint minimization problem (21) into four subproblems that are associated A, B, S, T separately. In each iteration, we iterate as follows,

$$\begin{aligned} T &\leftarrow \underset{T\in\mathbb{R}^{p\times q}}{\arg\min} \frac{1}{2}\|[X, \alpha\widehat{X}] - T[A, \alpha\widehat{A}]\|_F^2 + \frac{\gamma}{2}\|S - T + \bar{T}\|_F^2, \\ S &\leftarrow \underset{S\in\Omega_{p\times q}}{\arg\min} \frac{\gamma}{2}\|S - T + \bar{T}\|_F^2, \\ A &\leftarrow \underset{A\in\Pi_{q\times N}}{\arg\min} \frac{1}{2}\|X - TA\|_F^2 + \frac{\rho}{2}\|A - B + \bar{B}\|_F^2, \\ B &\leftarrow \underset{B\in\mathbb{R}^{q\times N}}{\arg\min} \lambda\mathcal{J}_1(B) + \lambda\mathcal{J}_2(B,\widehat{A},\widehat{X}) + \frac{\rho}{2}\|A - B + \bar{B}\|_F^2, \\ \bar{B} &\leftarrow \bar{B} - A + B, \\ \bar{T} &\leftarrow \bar{T} - S + T. \end{aligned} \quad (22)$$

The A, S, and T - subproblems are the same as in the blind unmixing paper [29], thus the details are omitted.

As for the B-subproblem, we write it explicitly by using $B = [\mathbf{b}_1, \mathbf{b}_2, \ldots, \mathbf{b}_N]$,

$$\begin{aligned} B &= \underset{B\in\mathbb{R}^{q\times N}}{\arg\min} \mathcal{J}_1(B) + \mathcal{J}_2(B,\widehat{A},\widehat{X}) + \frac{\rho}{2\lambda}\|A - B + \bar{B}\|_F^2 \\ &= \underset{B\in\mathbb{R}^{q\times N}}{\arg\min} \frac{1}{4}\sum_{i,j=1}^{N} \|\mathbf{b}_i - \mathbf{b}_j\|_2^2 w(\mathbf{x}_i, \mathbf{x}_j) \\ &\quad + \frac{1}{4}\sum_{i=1}^{N}\sum_{j=1}^{M} \|\mathbf{b}_i - \widehat{\mathbf{y}}_j\|_2^2 w(\mathbf{x}_i, \widehat{\mathbf{x}}_j) \\ &\quad + \frac{1}{4}\sum_{i,j=1}^{M} \|\widehat{\mathbf{y}}_i - \widehat{\mathbf{y}}_j\|_2^2 w(\widehat{\mathbf{x}}_i, \widehat{\mathbf{x}}_j) + \frac{\rho}{2\lambda}\|A - B + \bar{B}\|_F^2. \end{aligned} \quad (23)$$

Note that we add the term $\frac{1}{4}\sum_{i,j=1}^{M} \|\widehat{\mathbf{y}}_i - \widehat{\mathbf{y}}_j\|_2^2 w(\widehat{\mathbf{x}}_i, \widehat{\mathbf{x}}_j)$ in (23) that does not affect the minimization over B, but rather turns the B-subproblem (23) into a regularized graph Laplacian learning problem (see Section II-B). Specifically, we consider the graph $\widetilde{G}$ built on the combined data matrix $\widetilde{X} = [\widehat{X}, X]$ with the graph Laplacian matrix $\widetilde{L}$. Then (23) is equivalent to

$$B = \underset{B\in\mathbb{R}^{q\times N}}{\arg\min} \frac{1}{2}\left\langle [\widehat{A}, B]^\top, \widetilde{L}[\widehat{A}, B]^\top \right\rangle_F + \frac{\rho}{2\lambda}\|A - B + \widetilde{B}\|_F^2. \quad (24)$$

Using the block representation of $\widetilde{L}$ in (9), we have a closed-form solution to (24) as

$$B = \left(-\widehat{A}L_{lu} + \frac{\rho}{\lambda}(A + \widetilde{B})\right)\left(L_{uu} + \frac{\rho}{\lambda}I_N\right)^{-1}, \quad (25)$$

where I_N is an $N \times N$ identity matrix.

The semi-supervised unmixing (GRSU) method is summarized in Algorithm 3. Its initial values of A^0 and S^0 are obtained by GLU (Section III-B) that outputs $A_{\text{GLU}}, S_{\text{GLU}}$.

IV. Experiments and Results

In this section, we conduct extensive experiments to demonstrate the performance of the proposed unmixing models. All codes of our proposed methods and following experiments are available on our Github repository[1]. Specifically in Section IV-A, we compare our semi-supervised methods (GLU and GRSU) with the state-of-the-art (unsupervised) blind unmixing methods, followed by a discussion of our semi-supervised unmixing methods with respect to different numbers of training pixels in Section IV-B. In Section IV-C, we test the robustness of various methods by adding different amounts of Gaussian white noise to the HSI. We test on four standard hyperspectral image datasets, described as follows,

1) Jasper Ridge: The Jasper Ridge dataset [61] is a hyperspectral image of the size 100×100 with 198 channels. Originally it had 224 hyperspectral channels spanning from 380 to 2500 nm, and 26 channels are removed as a preprocessing step due to dense water vapor and atmospheric effects. Four endmembers are latent: Tree, Water, Dirt, and Road.
2) Samson: The Samson dataset [61] is of the size $95\times95\times$ with 156 channels that span from 401 to 889 nm. Three

[1]https://github.com/wispcarey/Nearly-Blind-Hyperspectral-Unmixing

Algorithm 3 Semi-supervised Unmixing

INPUT: data matrix X, label information $(\widehat{X}, \widehat{A})$, parameters $\alpha, \lambda, \gamma, \rho$, maximum iteration $I_{\max}$, and error tolerance ϵ.
CONSTRUCT: a graph $\widetilde{G} =$ on $\widetilde{X} = [\widehat{X}, X]$ (by Section II-A) with the graph Laplacian $\widetilde{L}$ and its block form of (9)
INITIALIZE: $S^0 = S_{\text{GLU}}$ and $A^0 = A_{\text{GLU}}$ (by GLU Algorithm 2); $B^0 = A^0$, $\bar{B}^0 = 0$, $\bar{T} = 0$, Err $= 1$, and $i = 0$.
WHILE $i < I_{\max}$ AND Err $> \epsilon$, DO:

$$T^{i+1} = \left(X(A^i)^\top + \alpha^2 \widehat{X}(\widehat{A})^\top + \gamma(S^i + \bar{T}^i)\right)\left(A^i(A^i)^\top + \alpha^2 \widehat{A}(\widehat{A})^\top + \gamma I_q\right)^{-1}.$$

$$S^{i+1} = \max\left(T^{i+1} - \bar{T}^i, 0\right).$$

$$A^{i+1} = \mathcal{P}_{\Pi_{q\times N}}\left(((S^{i+1})^\top S^{i+1} + \rho I_q)^{-1}((S^{i+1})^\top X + \rho(B^i - \bar{B}^i)\right).$$

$$B^{i+1} = \left(-\widehat{A}L_{ul} + \tfrac{\rho}{\lambda}(A^{i+1} + \bar{B}^i)\right)\left(L_{uu} + \tfrac{\rho}{\lambda} I_N\right)^{-1}$$

$$\bar{B}^{i+1} = \bar{B}^i + (A^{i+1} - B^{i+1}).$$

$$\bar{T}^{i+1} = \bar{T}^i + (S^{i+1} - T^{i+1}).$$

$i \leftarrow i + 1$ and Err $= \max\left((\|S^{i+1} - S^i\|_F)/(\|S^i\|_F), (\|A^{i+1} - A^i\|_F)/(\|A^i\|_F)\right)$
END WHILE
OUTPUT: $S_{\text{GRSU}} = S^i$, $A_{\text{GRSU}} = A^i$

TABLE I
TRAINING INFORMATION FOR EACH DATASET

	Parameters				***Training Data Info***			
Dataset	α	λ	γ	ρ	*Acq Fun*	*Training pixels*	*Training Percentage*	*Num Each Class*
Jasper	10	1	1	1	MCVOPT	44	0.44%	13, 7, 16, 8
Samson	20	50	0.1	0.1	VOPT	36	0.40%	13, 14, 9
Urban4	50	500	0.1	0.1	VOPT	364	0.38%	138, 105, 72, 49
Apex	10	50	1	1	VOPT	54	0.40%	7, 24, 13, 10

Parameter choices and training data information for our GLU and GRSU models. Parameters $\alpha, \lambda, \gamma, \rho$ are all associated with GRSU, while GLU only involves one parameter α (same as the one used in GRSU). "Acu Fun" means the acquisition function applied in the active learning process. "Training Pixels" means the number of labeled pixels used for the training process. "Training Percentage" means the percentage of labeled pixels to all pixels. "Num Each Class" means the number of labeled pixels of each endmember.

endmembers are latent: Soil, Tree, and Water. Note that a different ground truth is considered in [35], while we use the original ground-truth information.

3) Urban4: The Urban dataset [61] is of the size 307×307 with 162 channels. Each pixel of this image corresponds to a $2 \times 2m^2$ area. The original 221 channels span from 400 nm to 2500 nm. There are three versions of the ground truth, which contain 4, 5, and 6 endmembers. Here we use the version of four endmembers, labeled as Asphalt, Grass, Tree, and Roof.
4) Apex: The Apex dataset[2] [62] is a hyperspectral image of the size 111×122 with 285 bands spanning from 413 to 2420nm. Four endmembers are latent in this data: Road, Tree, Roof, and Water.

We apply two metrics, root mean square error (RMSE) and spectral angle distance (SAD), to evaluate the quality of the abundance matrix A and the endmember spectrum matrix S, respectively. RMSE and SAD are defined as follows,

$$\text{RMSE}(A, A^{\text{gt}}) = 100 \times \sqrt{\frac{1}{PN}\|A - A^{\text{gt}}\|_F^2} \tag{26}$$

$$\text{SAD}(S, S^{\text{gt}}) = \frac{180}{\pi} \times \frac{1}{p}\sum_{i=1}^{p} \arccos\left(\frac{\langle \mathbf{s}_i, \mathbf{s}_i^{\text{gt}}\rangle}{\|\mathbf{s}_i\|_2 \|\mathbf{s}_i^{\text{gt}}\|_2}\right), \tag{27}$$

where A, S are the fitted matrices, $A^{\text{gt}}, S^{\text{gt}}$ are the ground-truth, and $\mathbf{s}_i$ denotes the i^{th} column of the matrix S.

[2]https://github.com/BehnoodRasti/MiSiCNet

A. Method Comparison

We compare our semi-supervised methods with five state-of-the-art unsupervised unmixing methods, namely, FCLSU [10], GLNMF [26], QMV [20], GTVMBO [30], MSC [35] and EGU [36]. The first three methods (GLNMF, QMV, GTVMBO) initialize with the output of FCLSU [10]. MSC and EGU are neural network methods. Note that the EGU method supports the use of either the ground truth endmember spectrum matrix S or the estimated matrix S using vertex component analysis (VCA) [63]. We use VCA to estimate S, which serves an input for the pixel-wise EGU-net method. In the following experiments, GLNMF [26], QMV [20], and GTVMBO [30] are executed in MATLAB and conducted on an Intel i9-9900K CPU, while MSC [35] and EGU [36] are implemented in Python and conducted on an Nvidia 2080 Ti GPU.

Table II shows the computation times of various methods applied to each dataset. Our semi-supervised method GLU is much faster than the neural network methods approaches (MSC and EGU) and is comparable to traditional unsupervised methods (GLNMF, QMV, and GTVMBO) in terms of computation time. As the GRSU method requires solving the graph Laplace learning problem in each iteration, it does require more computation times compared to the regularization-based methods, while it is still faster than neural network methods.

For labels used in our semi-supervised framework, we consider the exact abundance map (EXT) and the one-hot pseudo label (OH). The latter (OH) can be obtained by

TABLE II
COMPUTATION TIMES FOR DIFFERENT METHODS

	Unsupervised Methods					*Our Semi-supervised Methods*			
Dataset	*GLNMF* [26]	*QMV* [20]	*GTVMBO* [30]	*MSC* [35]	EGU [36]	*GLU-OH*	*GLU-EXT*	*GRSU-OH*	*GRSU-EXT*
Jasper	$8.81s^*$	**2.51s***	$2.77s^*$	112.09s	51.36s	**2.38s**	2.45s	3.07s	2.92s
Samson	$3.89s^*$	$1.40s^*$	**0.51s***	95.32s	43.58s	1.52s	**1.47s**	10.42s	12.92s
Urban4	$67.31s^*$	$23.57s^*$	**14.60s***	974.05s	583.16s	**22.29s**	22.37s	201.89s	294.71s
Apex	$26.47s^*$	$2.12s^*$	**0.59s***	166.03s	87.84s	**3.81s**	3.97s	20.46s	24.04s

This table presents the computation times of various methods applied to each dataset. Specifically, the methods of GLNMF [26], QMV [20], and GTVMBO [30] are executed in MATLAB, with their times being indicated by an asterisk ($*$) while the other methods are implemented in Python. The computation time is measured in seconds. In each row, the best computation times from the unsupervised and semi-supervised methods are highlighted in bold. Regarding computational time, the proposed methods (GLU and GRSU) run much faster than the neural network methods approaches (MSC and EGU), and are comparable to traditional unsupervised methods (GLNMF, QMV, and GTVMBO).

TABLE III
RMSE BETWEEN THE ESTIMATED ABUNDANCE MATRIX AND THE GROUND TRUTH.

		Unsupervised Methods					*Our Semi-supervised Methods (0.4% training)*			
Dataset	***Class***	*GLNMF* [26]	*QMV* [20]	*GTVMBO* [30]	*MSC* [35]	*EGU* [36]	*GLU-OH*	*GLU-EXT*	*GRSU-OH*	*GRSU-EXT*
Jasper	Tree	**10.04**	10.59	18.70	12.34	11.25	7.09	10.91	**4.04**	6.48
	Water	14.06	**4.18**	7.38	7.30	6.22	7.33	11.40	**4.39**	4.27
	Dirt	14.32	**10.93**	17.10	15.35	14.59	8.64	13.04	**6.48**	6.85
	Road	51.82	**14.74**	15.64	52.09	32.84	14.78	15.75	6.99	**6.34**
	Overall	18.79	**9.51**	15.16	18.24	13.78	8.37	12.03	**5.10**	5.93
Samson	Soil	22.51	20.28	**8.01**	20.05	30.73	4.37	5.73	5.18	**4.10**
	Tree	31.29	24.92	**16.10**	28.90	29.56	7.51	6.02	6.35	**4.38**
	Water	16.26	16.63	**9.19**	16.45	15.58	11.06	4.70	11.32	**4.89**
	Overall	25.21	21.48	**12.19**	23.32	27.08	7.81	5.61	7.66	**4.43**
Urban4	Asphalt	33.89	21.58	19.19	**18.19**	21.4	14.48	**7.99**	14.55	8.32
	Grass	**10.15**	29.07	10.44	31.61	45.83	7.61	**6.86**	7.47	6.93
	Tree	15.82	25.65	**13.87**	31.37	14.32	**7.78**	9.51	7.88	9.67
	Roof	**17.53**	31.18	23.49	38.00	18.86	9.93	10.65	**9.78**	10.59
	Overall	22.13	26.17	**15.71**	28.58	30.77	10.48	**8.30**	10.49	8.46
Apex	Road	23.13	25.34	57.78	**22.77**	43.61	11.35	17.40	**10.16**	13.68
	Tree	**13.51**	19.99	32.29	14.08	25.27	11.33	12.22	11.07	**9.44**
	Roof	29.15	25.65	46.64	**19.91**	33.67	16.67	**12.69**	17.31	13.71
	Water	16.81	8.70	21.63	**5.50**	11.56	14.85	**14.43**	15.08	14.85
	Overall	19.28	20.09	37.10	**15.27**	27.49	13.34	13.44	13.35	**12.15**

Comparison results in terms of RMSE(A, A^{gt}) for the abundance maps: four of our semi-supervised methods (with around 0.4% of labeled pixels) are compared with five (unsupervised) blind unmixing methods on four publicly available datasets. For each row, the best results of unsupervised methods and our semi-supervised methods are bolded, respectively. The best of our methods achieves nearly 50% improvements over the unsupervised ones in most cases.

TABLE IV
SAD BETWEEN THE ESTIMATED SPECTRUM MATRIX AND THE GROUND TRUTH.

		Unsupervised Methods					*Our Semi-supervised Methods (0.4% training)*			
Dataset	***Class***	*GLNMF* [26]	*QMV* [20]	*GTVMBO* [30]	*MSC* [35]	*EGU* [36]	*GLU-OH*	*GLU-EXT*	*GRSU-OH*	*GRSU-EXT*
Jasper	Tree	5.63	2.66	13.95	**2.48**	8.49	7.49	2.50	**2.00**	4.70
	Water	**3.60**	5.02	22.43	16.60	14.64	22.21	22.66	**5.04**	22.27
	Dirt	6.53	**2.50**	8.37	3.80	6.68	3.99	**1.80**	1.89	2.54
	Road	43.94	**3.96**	6.61	18.88	5.16	2.41	2.05	**1.28**	2.20
	Overall	14.92	**3.55**	12.83	10.44	8.74	9.03	7.25	**2.55**	7.92
Samson	Soil	1.37	2.48	2.59	23.13	**1.35**	2.07	1.87	**1.44**	2.32
	Tree	**1.62**	3.04	5.53	2.03	2.29	4.12	**2.54**	3.20	2.56
	Water	10.48	32.94	21.40	45.95	**8.62**	9.50	30.96	**2.41**	31.45
	Overall	4.50	12.82	9.83	23.71	**4.09**	5.24	11.79	**2.36**	12.11
Urban4	Asphalt	7.36	150.64	**6.98**	43.06	7.62	6.35	**3.02**	6.25	3.08
	Grass	10.85	23.49	**6.66**	22.39	37.45	3.49	3.82	**3.35**	3.74
	Tree	9.11	7.52	**1.58**	5.67	4.86	7.02	**3.80**	6.35	4.39
	Roof	44.15	3.85	44.23	**2.59**	45.89	3.26	3.89	**3.22**	4.16
	Overall	17.87	46.37	**14.87**	18.43	23.96	5.04	**3.64**	4.79	3.83
Apex	Road	29.38	8.39	40.88	14.90	**5.61**	7.68	14.95	**4.25**	17.61
	Tree	**6.55**	11.86	24.21	7.88	7.41	10.15	**5.08**	8.03	6.62
	Roof	6.23	6.26	5.65	10.10	**4.86**	9.59	6.47	6.82	**4.30**
	Water	**9.70**	106.13	51.46	43.85	12.73	30.73	**11.33**	12.62	17.11
	Overall	12.96	33.16	30.55	19.19	**7.65**	14.54	9.46	**7.94**	11.41

Comparison results in terms of SAD(S, S^{gt}) for the endmember spectrum matrices S: four of our semi-supervised methods (with around 0.4% of labeled pixels) are compared with five (unsupervised) blind unmixing methods on four publicly available datasets. For each row, the best results of unsupervised methods and our semi-supervised methods are bolded, respectively.

thresholding the exact abundance map or an expert identifying the most significant endmember, which is more practical than the former (EXT) in real circumstances.

For each experiment, the active learning process starts with only one random pixel per material as an initial label set and terminates until around 0.4% of the total pixels are sampled according to the active learning algorithm (Algorithm 1). We fix $\epsilon = 10^{-3}$, $I_{\max} = 1000$ and $K = 50$ in the KNN graph construction (Section II-A). Note that K is chosen to be relatively small for computational benefits, while ensuring the connectivity of the constructed graph G, which is required in the calculation of the acquisition function (28). We select the optimal combination of the parameters $\alpha, \lambda, \gamma, \rho$ in the range of $\alpha \in \{10, 20, 50, 100\}$, $\lambda \in \{10^i, 5 \times 10^i | i = 0, 1, 2, 3\}$, $\gamma = \rho \in \{10^i | i = -2, -1, 0, 1, 2\}$ that yields the smallest sum of RMSE and SAD using 10% of randomly selected pixels as a validation set. We list the parameter choices, the amount of training data, and the acquisition function (Acq Fun) in the active learning approach for each dataset in Table I.

Table III and Table IV report the results of RMSE(A, A^{gt}) values and SAD(S, S^{gt}) values, respectively. For our methods, the '-OH' suffix refers to training on the one-hot pseudo labels, while the '-EXT' suffix refers to training on the exact abundance maps. In both tables, we highlight the best results in our four semi-supervised methods and the other five unsupervised methods, separately. By comparing the bold results in each row, the best of our methods achieves nearly 50% improvements over the competing unsupervised methods in most cases. There are only a few exceptions. For example, EGU has an outstanding performance of the SAD on the Apex dataset, which is slightly better than ours.

In addition, we observe that the winner of the five supervised methods scatters over Table III and Table IV. For example, QMV and GTVMBO perform the best on Jasper Ridge and Urban4, respectively, for both abundance and endmember estimations. For the Samson dataset, GTVMBO attains the best RMSE, while EGU has the best SAD. For Apex, MSC has the best RMSE, while EGU achieves the best SAD. Our methods, on the other hand, yield consistent performance in that GRSU is generally better than GLU. One exception is the Urban4 dataset, in which GRSU-EXT is slightly worse than GLU-EXT. Since GLU is the initialization of GRSU, we can see improvements of GRSU after the ADMM iterations, which are particularly significant on the SAD values of the Jasper Ridge, Samson, and Apex datasets. Furthermore, training on one-hot pseudo labels (OH) sometimes has a better performance than training on the exact abundance map (EXT), which implies that it is not necessary to require the exact abundance maps for the training process of our approaches.

We arrange the estimated abundance maps and endmembers in pairs for Jasper Ridge, Samson, Urban4, and Apex datasets, sequentially, showing in Figures 2–9. Note that we normalize each endmember (column) in the spectrum matrix S to have the unit norm, i.e., $\|\mathbf{s}_i\|_2 = \|\mathbf{s}_i^{\text{gt}}\|_2 = 1$. The labeled pixels that are selected by active learning are indicated in red dots on the ground truth abundance map.

The active learning approach can identify the distribution of each endmember by selecting a few representatives. Take the Road in the Jasper Ridge dataset for an example. As illustrated in the last row of Figure 2, the abundance map for the Road contains fine structures that are easily smeared out by other methods, while active learning can successfully identify pixels with high abundance values of this endmember to acquire labels. With those sampled labeled pixels, the estimated abundance maps' quality increases significantly compared to unsupervised methods. Another evident example is the endmember Roof in the Urban dataset (the last row of Figure 6). Both GLU-OH and GRSU-OH well preserve the contrast of the rectangular rooftop in the Roof abundance.

In conclusion, our methods demonstrate a significant improvement of approximately 50% in unmixing performance as measured by the RMSE of the abundance maps and the SAD of the mixing matrices, requiring only a minimum amount of supervision (e.g., 0.4% labeled pixels). It is important to note that exact abundance maps are not needed during the training process; instead, we leverage the practical and readily obtainable OH labels. Furthermore, while our GRSU methods may exhibit slower computation times due to the iterative nature of our optimization approach, they still maintain a speed advantage over neural network methods, such as MSC and EGU. This not only underlines the efficiency of our methods but also represents a competitive balance between computational speed and unmixing performance.

B. Discussion on the Number of Training Pixels

We discuss the influence of the number of labels sampled by active learning on our semi-supervised methods, GLU-OH, GLU-EXT, GRSU-OH, and GRSU-EXT. The active learning process starts with one random pixel per material as an initial label set, followed by Algorithm 1 (the active learning algorithm) until 5% of the total pixels are reached. We conduct experiments only on the Jasper Ridge and the Apex datasets for demonstration purposes. The corresponding parameters λ, γ, ρ are provided in Table I. The parameter α is designed to balance the numbers of the training pixels and total pixels. Since we have various numbers of training pixels in this experimental setting, we choose $\alpha_0 = 10$ and $\alpha = C_\alpha \alpha_0 / N_{\text{train}}$, where C_α is a constant depending on the dataset and N_{train} is the number of training pixels. In practice, $C_\alpha = 50$ for the Jasper Ridge dataset and $C_\alpha = 10$ for the Apex dataset.

Figure 10 shows the changes in the RMSE on abundance and SAD on the endmembers with respect to the number of labeled pixels obtained by active learning. Two abundance RMSE curves in Figure 10 (a) and (c) illustrate that our methods trained on one-hot pseudo labels (GLU-OH, GRSU-OH) improve in the very beginning and deteriorate with more labeled pixels (after 1%), which is attributed to the inaccurate information of the OH labels. At the beginning of active learning, the most representative pixels are selected, whose ground-truth abundance vectors might be close to a one-hot vector. When we incorporate more OH labels for unmixing, the OH labels are misleading as opposed to the exact abundance map. On the other hand, GLU-EXT and GRSU-EXT always benefit from increasing the number of labeled pixels increases,

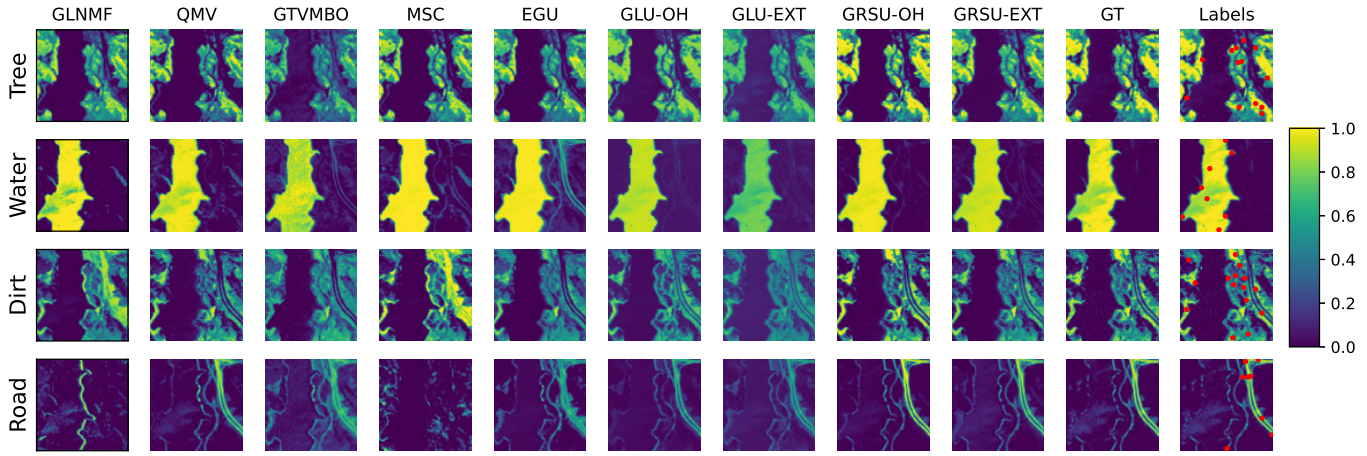


Fig. 2. Abundance maps estimated by different methods on the Jasper Ridge dataset. Each row of the plot matrix corresponds to an endmember of the dataset, including Tree, Water, Dirt, and Road. The first five columns are (unsupervised) blind unmixing methods and the following four columns are our semi-supervised methods. The last two columns are the ground truth and the label pixels selected by active learning (red dots) for our GLU and GRSU methods. Each red dot corresponds to a labeled pixel, which is enlarged for visual illustration. Notice that the Road abundance map contains fine structures that are easily smeared out by other methods, while active learning can successfully identify pixels with high abundance values for Road to acquire labels.

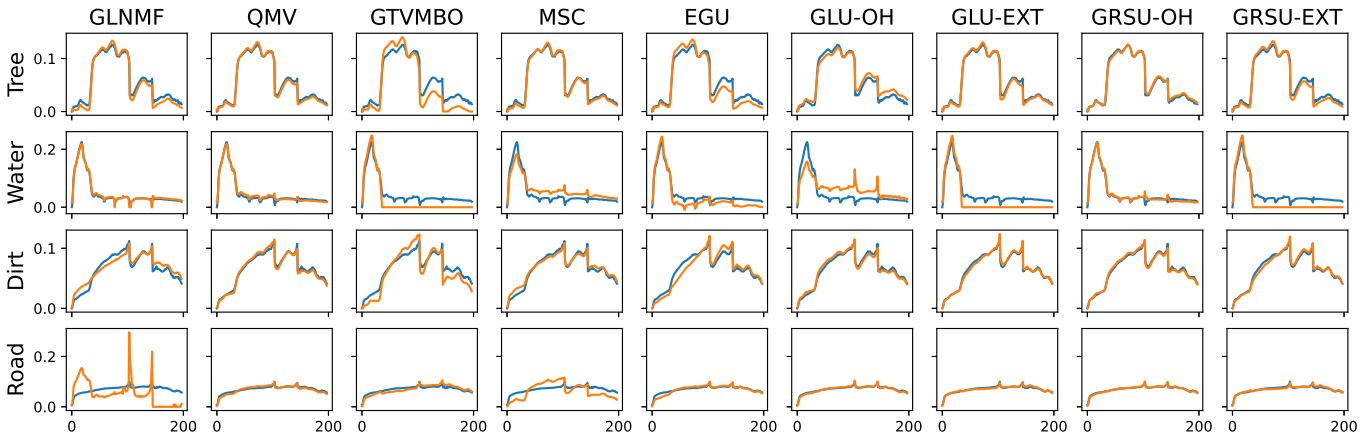


Fig. 3. Endermember matrix estimated by different methods (in orange) on the Jasper Ridge dataset with the ground truth (in blue). All the endmember vectors are normalized to have the unit norm. Each row of the plot matrix corresponds to an endmember of the dataset, including Tree, Water, Dirt, and Road. The first five columns are (unsupervised) blind unmixing methods and the following four columns are our semi-supervised methods.

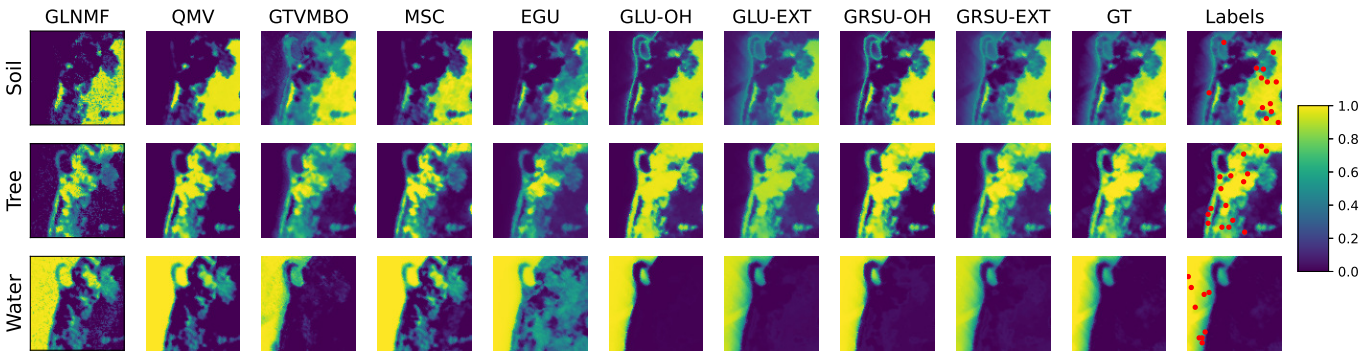


Fig. 4. Abundance maps estimated by different methods on the Samson dataset. Each row of the plot matrix corresponds to an endmember of the dataset, including Soil, Tree, and Water. The first five columns are (unsupervised) blind unmixing methods and the following four columns are our semi-supervised methods. The last two columns are the ground truth and the label pixels selected by active learning (red dots) for our GLU and GRSU methods. Each red dot corresponds to a labeled pixel, which is enlarged for visual illustration.

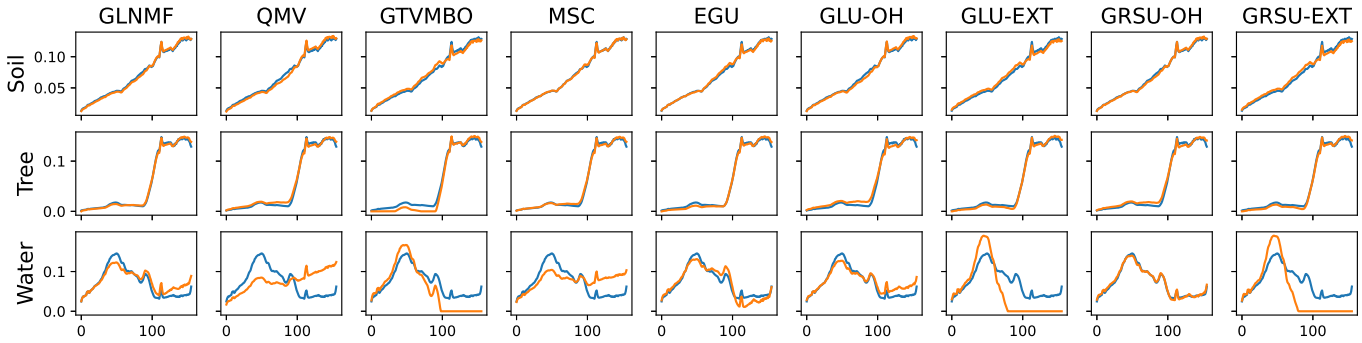


Fig. 5. Endermember matrix estimated by different methods (in orange) on the Samson dataset with the ground truth (in blue). All the endmember vectors are normalized to have the unit norm. Each row of the plot matrix corresponds to an endmember of the dataset, including Soil, Tree, and Water. The first five columns are (unsupervised) blind unmixing methods and the following four columns are our semi-supervised methods.

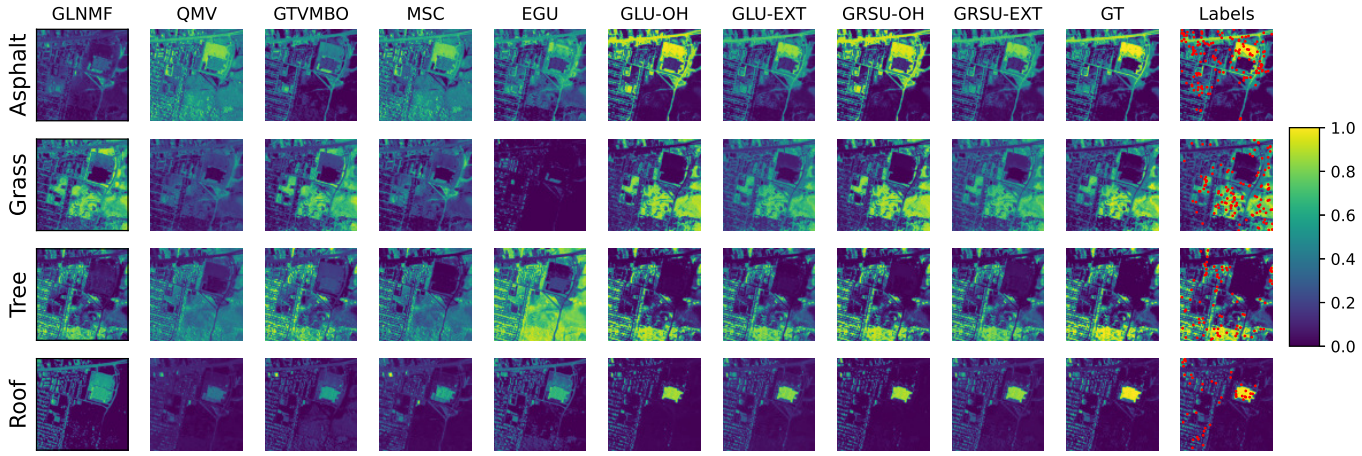


Fig. 6. Abundance maps estimated by different methods on the Urban4 dataset. Each row of the plot matrix corresponds to an endmember of the dataset, including Asphalt, Grass, Tree, and Roof. The first five columns are (unsupervised) blind unmixing methods and the following four columns are our semi-supervised methods. The last two columns are the ground truth and the label pixels selected by active learning (red dots) for our GLU and GRSU methods. Each red dot corresponds to a labeled pixel, which is enlarged for visual illustration. Note that both GLU-OH and GRSU-OH well preserve the contrast of the rectangular rooftop in the Roof abundance.

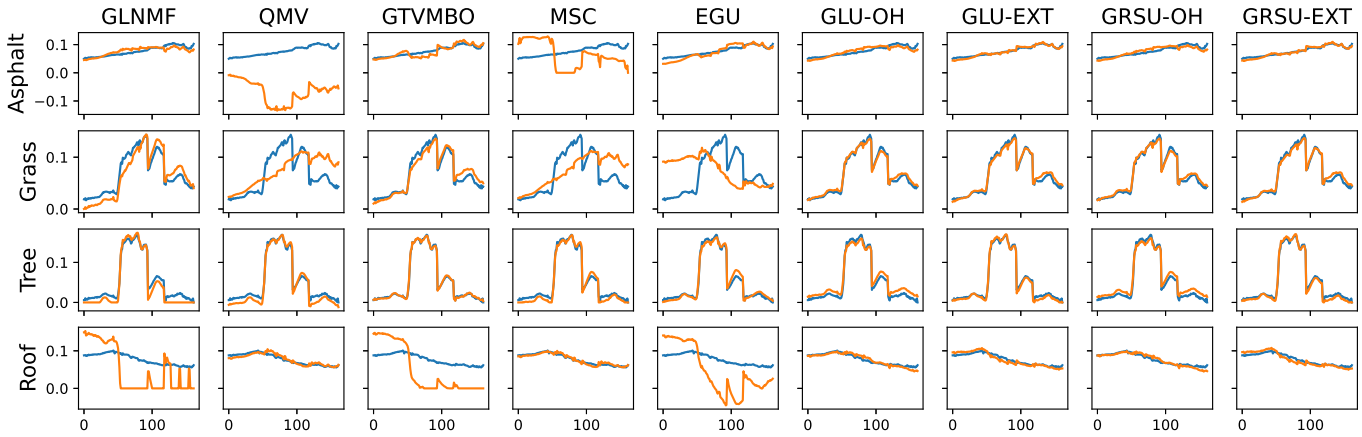


Fig. 7. Endermember matrix estimated by different methods (in orange) on the Urban4 dataset with the ground truth (in blue). All the endmember vectors are normalized to have the unit norm. Each row of the plot matrix corresponds to an endmember of the dataset, including Asphalt, Grass, Tree, and Roof. The first five columns are (unsupervised) blind unmixing methods and the following four columns are our semi-supervised methods.

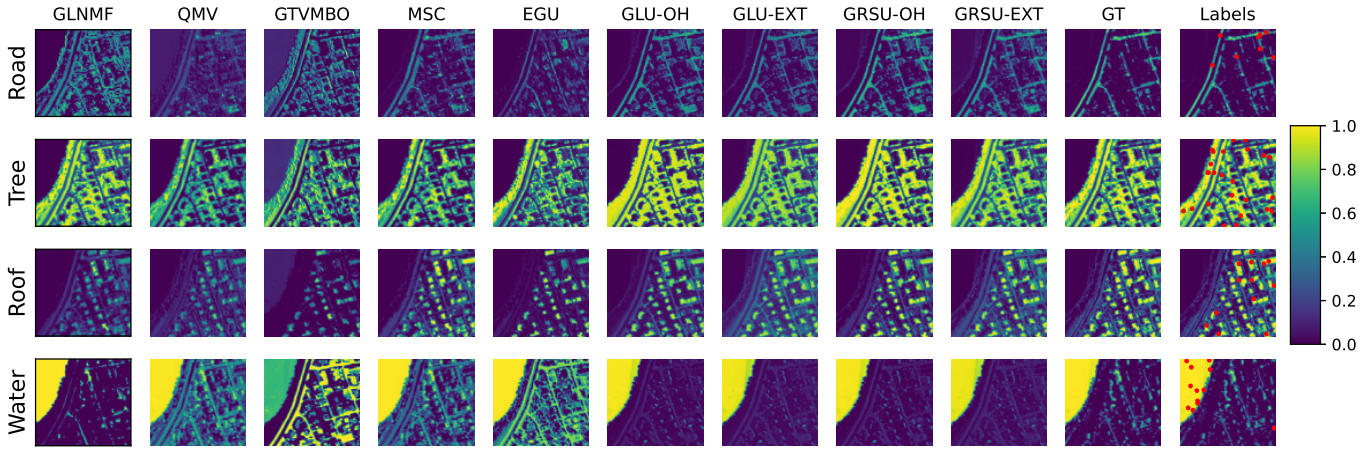


Fig. 8. Abundance maps estimated by different methods on the Apex dataset. Each row of the plot matrix corresponds to an endmember of the dataset, including Road, Tree, Roof, and Water. The first five columns are (unsupervised) blind unmixing methods and the following four columns are our semi-supervised methods. The last two columns are the ground truth and the label pixels selected by active learning (red dots) for our GLU and GRSU methods. Each red dot corresponds to a labeled pixel, which is enlarged for visual illustration.

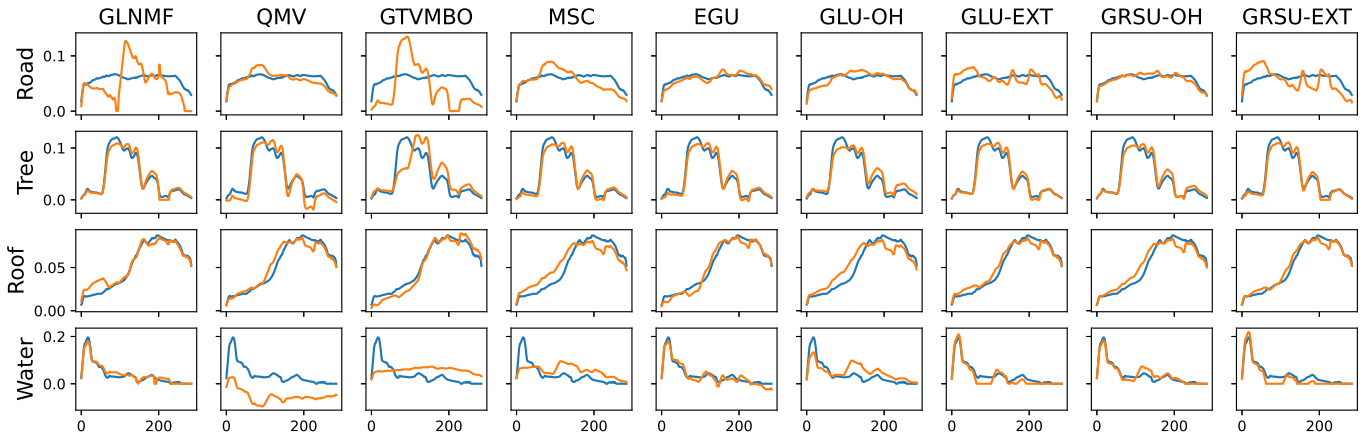


Fig. 9. Endermember matrix estimated by different methods (in orange) on the Apex dataset with the ground truth (in blue). All the endmember vectors are normalized to have the unit norm. Each row of the plot matrix corresponds to an endmember of the dataset, including Road, Tree, Roof, and Water. The first five columns are (unsupervised) blind unmixing methods and the following four columns are our semi-supervised methods.

but with a diminishing gain after around 3% of the label rate. In practice, we would like to use the one-hot pseudo labels for training, since it is much easier to obtain, but not exceeding 1% for the active learning process.

Compared to the RMSE curves, the endmember SAD curves are relatively more stable with respect to the increase in the number of training pixels. Practically, after 2% of the label rate, the SAD values do not change much for all our methods, which implies that only a small number of training pixels are needed to estimate the endmembers.

Overall for both GLU and GRSU, the increase in the training pixels does not always deem improvements in the performances. In fact, including more OH pseudo labels is not beneficial for unmixing. A label rate around 1% would be a good choice experimentally.

C. Robustness Study

In order to provide a quantitative validation of the robustness exhibited by our approach, we have undertaken an extensive investigation into the performances evaluated by RMSE (26) and SAD (27) of various methods applied to the Jasper Ridge dataset. We add different amounts of Gaussian white noise to this data matrix such that the resulting data has the signal-to-noise ratio (SNR) ranging from 5dB to 40dB with a 5dB increment. Note that a higher SNR value implies less noisy data.

As evidenced in Figure 11, the proposed GRSU method achieves the highest accuracy in estimating the abundance map A, no matter whether OH or EXT is considered under all the noise levels. The spectrum matrix S estimated by GRSU is the best for low SNR regime, and is comparable to QMV when the data is less noisy (SNR$>$15dB). This result is reasonable, as we only impose two simple constraints on S (i.e., nonnegativity and sum-to-one). On the other hand, the GLU method seems sensitive to noise. In circumstances where the SNR is relatively low, the RMSE and SAD values produced by GLU tend to be large. But we also observe a sharp decline in both RMSE and SAD by the GLU method with a gradual

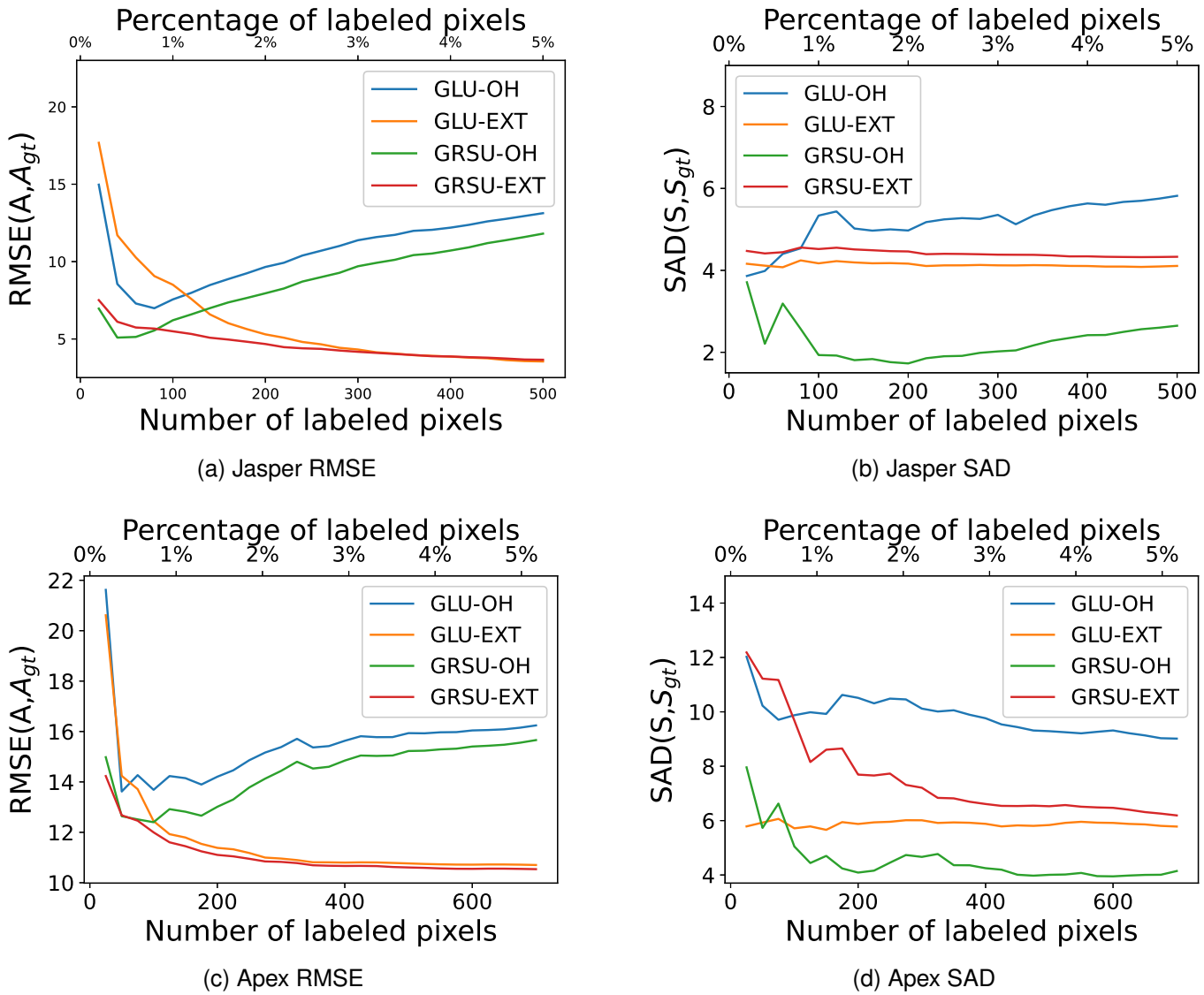
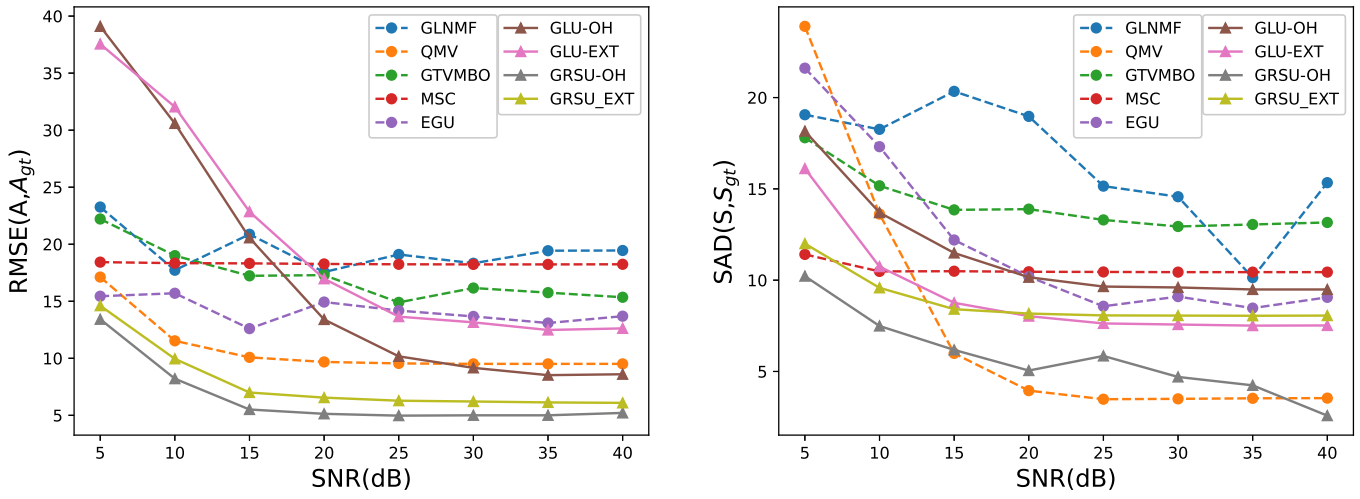


Fig. 10. RMSE and SAD curves with respect to the number of labeled pixels for the Jasper Ridge and the Apex datasets. For each plot, the x-axis on the top shows the percentage of labeled pixels, while the bottom one is the number of labeled pixels. In the active learning process, we apply the MCVOpt and VOpt acquisition functions for the Jasper Ridge and Apex datasets respectively. Each curve starts with only one random pixel per endmember and sample up to 5% of labeled pixels.

increase in SNR.

V. Conclusion

In this paper, we propose two semi-supervised hyperspectral models, the graph learning unmixing (GLU) and graph-regularized semi-supervised unmixing (GRSU). GLU applies the graph Laplace learning directly to solve the HSU problem by regarding the class probability as the abundance map. Initialized by GLU, GRSU estimates the abundance map A and endmember spectrum matrix S by solving an energy-minimizing problem via an iterative ADMM scheme. We extended the graph Laplace learning to a regularized version and explored the close-form solutions for efficient computation. This regularized graph Laplace learning is a subproblem in the ADMM iteration process. We conducted extensive experiments using four standard hyperspectral datasets to compare our semi-supervised methods with five state-of-the-art methods in hyperspectral blind unmixing. All the results demonstrated the proposed GLU and GRSU methods have a significant improvement with a minimum amount of supervision. Furthermore, both methods can take either the ground-truth abundance maps or the one-hot pseudo labels as the training information, the latter of which is much easier to obtain since it only requires determining the major endmember of each training pixel. According to our experiments, it is unnecessary to require the ground-truth abundance maps to feed in the semi-supervised framework. Sometimes, models trained on pseudo labels yield better performance than the exact values. We also discussed the influence of the number of training pixels on the model performance, revealing that a label rate of 1% is experimentally sufficient for satisfactory results. In addition, our GRSU method has great robustness over Gaussian white



(a) RMSE vs SNR (b) SAD vs SNR

Fig. 11. RMSE (for A) and SAD (for S) curves with respect to the SNR values of noisy input of the Jasper Ridge dataset corrupted by Gaussian white noise. In both panels, the performance of other blind methods is illustrated by dashed lines, whereas our proposed semi-supervised methods are represented by solid lines.

noise. The new methods have a great potential for real world problems since they do not need a ground truth abundance map and instead can work with pseudo-labels.

There are several promising directions following this work. First, scaling factor [1] and spectral variability [2] terms can be incorporated into our model to further enhance the unmixing performance. Second, Figure 11 (b) suggests a need for regularizations of the spectral matrix, in addition to nonnegativity and sum-to-one constraints, for improving the robustness of the proposed methods. Lastly, the current graph Laplacian learning solver can be replaced with neural networks or graph neural network techniques.

Appendix
Acquisition Function Formulas

To make the paper self-contained, we provide the formulas for the VOpt [51] and MCVopt [45] functions. Given a graph $G(\mathcal{X}, W)$, the formulas are based on a low-rank approximation of the graph Laplacian matrix L. With ordered eigenvalues of L as $0 = \lambda_1 < \lambda_2 \leq \ldots \leq \lambda_N$, the truncated decomposition of L with the smallest $M < N$ eigenvalues has the form $L \approx V\Lambda V^\top$, where $\Lambda \in \mathbb{R}^{M\times M}$ is a diagonal matrix with diagonal entries $\lambda_1, \lambda_2, \ldots, \lambda_M$ and $V \in \mathbb{R}^{N\times M}$ is the matrix of corresponding eigenvectors. Furthermore, a Gaussian correlation matrix C is defined by:

$$C = \left(\Lambda + V^\top \left(\frac{1}{\gamma^2} P^\top P\right) V\right)^{-1}, \tag{28}$$

where $P \in \mathbb{R}^{|\mathcal{X}_L|\times N}$ is a projection matrix onto the indices corresponding to the label set $\widehat{\mathcal{X}}$, and γ is a positive constant (we choose $\gamma = 0.1$ in this paper). When the graph G is connected, such a matrix C exists, i.e., the matrix $\Lambda + V^\top \left(\frac{1}{\gamma^2} P^\top P\right) V$ is invertible.

At each iteration in the active learning process with the current label set $\widehat{\mathcal{X}}$, graph Laplace learning gives the optimized node function $\mathbf{u}^*$ (and corresponding node matrix U^*) by minimizing the energy (3). Denoting the k^{th} column of $V^\top$ by $\mathbf{v}_k$, the VOpt and MCVOpt acquisition functions are given by:

$$\mathcal{A}_{\text{VOpt}}(\mathbf{x}_k) = \frac{1}{\gamma^2 + \mathbf{v}_k^\top C \mathbf{v}_k} \|C\mathbf{v}_k\|_2^2, \tag{29}$$

$$\mathcal{A}_{\text{MCVOpt}}(\mathbf{x}_k) = \frac{1}{\gamma^2 + \mathbf{v}_k^\top C \mathbf{v}_k} \|C\mathbf{v}_k\|_2^2 \|(U^*)^\top V \mathbf{v}_k - \mathbf{y}_k^*\|_2, \tag{30}$$

where $\mathbf{y}_k^*$ is the one-hot thresholding of $\mathbf{u}^*(\mathbf{x}_k)$.

References

[1] L. Drumetz, M.-A. Veganzones, S. Henrot, R. Phlypo, J. Chanussot, and C. Jutten, "Blind hyperspectral unmixing using an extended linear mixing model to address spectral variability," *IEEE Trans. Image Process.*, vol. 25, no. 8, pp. 3890–3905, 2016.

[2] D. Hong, N. Yokoya, J. Chanussot, and X. X. Zhu, "An augmented linear mixing model to address spectral variability for hyperspectral unmixing," *IEEE Trans. Image Process.*, vol. 28, no. 4, pp. 1923–1938, 2019.

[3] J. M. 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 Observ. Remote Sens.*, vol. 5, no. 2, pp. 354–379, 2012.

[4] R. Heylen, M. Parente, and P. Gader, "A review of nonlinear hyperspectral unmixing methods," *IEEE J. Sel. Topics Appl. Earth Obser. Remote Sens.*, vol. 7, no. 6, pp. 1844–1868, 2014.

[5] D. D. Lee and H. S. Seung, "Learning the parts of objects by non-negative matrix factorization," *Nature*, vol. 401, no. 6755, p. 788, 1999.

[6] X.-R. Feng, H.-C. Li, R. Wang, Q. Du, X. Jia, and A. Plaza, "Hyperspectral unmixing based on nonnegative matrix factorization: A comprehensive review," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 15, pp. 4414–4436, 2022.

[7] P. Sajda, S. Du, T. Brown, L. Parra, and R. Stoyanova, "Recovery of constituent spectra in 3D chemical shift imaging using nonnegative matrix factorization," in *4th International Symposium on Independent Component Analysis and Blind Signal Separation*, pp. 71–76, 2003.

[8] V. P. Pauca, J. Piper, and R. J. Plemmons, "Nonnegative matrix factorization for spectral data analysis," *Linear Algebra Appl.*, vol. 416, no. 1, pp. 29–47, 2006.

[9] A. Cichocki, R. Zdunek, A. H. Phan, and S. I. Amari, *Nonnegative matrix and tensor factorizations: applications to exploratory multi-way data analysis and blind source separation*. John Wiley & Sons, 2009.

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

[11] M. D. Iordache, J. M. Bioucas-Dias, and A. Plaza, "Sparse unmixing of hyperspectral data," *IEEE Trans. Geosci. Remote Sens.*, vol. 49, no. 6, pp. 2014–2039, 2011.

[12] W. He, H. Zhang, and L. Zhang, "Sparsity-regularized robust non-negative matrix factorization for hyperspectral unmixing," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 9, no. 9, pp. 4267–4279, 2016.

[13] 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, 2011.

[14] L. Drumetz, T. R. Meyer, J. Chanussot, A. L. Bertozzi, and C. Jutten, "Hyperspectral image unmixing with endmember bundles and group sparsity inducing mixed norms," *IEEE Trans Image Process*, vol. 28, no. 7, pp. 3435–3450, 2019.

[15] L. I. Rudin, S. J. Osher, and E. Fatemi, "Nonlinear total variation based noise removal algorithms," *Physica D: nonlinear phenomena*, vol. 60, no. 1-4, pp. 259–268, 1992.

[16] M. D. Iordache, J. M. Bioucas-Dias, and A. Plaza, "Total variation spatial regularization for sparse hyperspectral unmixing," *IEEE Trans. Geosci. Remote Sens.*, vol. 50, no. 11, pp. 4484–4502, 2012.

[17] W. He, H. Zhang, and L. Zhang, "Total variation regularized reweighted sparse nonnegative matrix factorization for hyperspectral unmixing," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 7, pp. 3909–3921, 2017.

[18] F. Xiong, Y. Qian, J. Zhou, and Y. Y. Tang, "Hyperspectral unmixing via total variation regularized nonnegative tensor factorization," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 4, pp. 2341–2357, 2018.

[19] Y. Yuan, Z. Zhang, and Q. Wang, "Improved collaborative non-negative matrix factorization and total variation for hyperspectral unmixing," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 13, pp. 998–1010, 2020.

[20] L. Zhuang, C.-H. Lin, M. A. Figueiredo, and J. M. Bioucas-Dias, "Regularization parameter selection in minimum volume hyperspectral unmixing," *IEEE Trans Geosci Remote Sens*, vol. 57, no. 12, pp. 9858–9877, 2019.

[21] A. L. Bertozzi and E. Merkurjev, "Graph-based optimization approaches for machine learning, uncertainty quantification and networks," in *Handbook of Numerical Analysis*, vol. 20, pp. 503–531, Elsevier, 2019.

[22] K. Benzi, V. Kalofolias, X. Bresson, and P. Vandergheynst, "Song recommendation with non-negative matrix factorization and graph total variation," in *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 2439–2443, 2016.

[23] D. I. Shuman, S. K. Narang, P. Frossard, A. Ortega, and P. Vandergheynst, "The emerging field of signal processing on graphs: Extending high-dimensional data analysis to networks and other irregular domains," *IEEE Signal Process. Mag.*, vol. 30, no. 3, pp. 83–98, 2013.

[24] A. Sandryhaila and J. M. Moura, "Discrete signal processing on graphs: Frequency analysis," *IEEE Trans. Signal Process.*, vol. 62, no. 12, pp. 3042–3054, 2014.

[25] F. Zhu, Y. Wang, S. Xiang, B. Fan, and C. Pan, "Structured sparse method for hyperspectral unmixing," *ISPRS J Photogramm Remote Sens*, vol. 88, pp. 101–118, 2014.

[26] X. Lu, H. Wu, Y. Yuan, P. Yan, and X. Li, "Manifold regularized sparse nmf for hyperspectral unmixing," *IEEE Trans Geosci Remote Sens*, vol. 51, no. 5, pp. 2815–2826, 2012.

[27] M. Li, F. Zhu, A. J. X. Guo, and J. Chen, "A graph regularized multilinear mixing model for nonlinear hyperspectral unmixing," *Remote Sens.*, vol. 11, no. 19, p. 2188, 2019.

[28] C. Fowlkes, S. Belongie, F. Chung, and J. Malik, "Spectral grouping using the Nyström method," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 2, pp. 214–225, 2004.

[29] J. Qin, H. Lee, J. T. Chi, Y. Lou, J. Chanussot, and A. L. Bertozzi, "Fast blind hyperspectral unmixing based on graph Laplacian," in *Workshop on Hyperspectral Imaging and Signal Processing: Evolution in Remote Sensing (WHISPERS)*, pp. 1–5, IEEE, 2019.

[30] J. Qin, H. Lee, J. T. Chi, L. Drumetz, J. Chanussot, Y. Lou, and A. L. Bertozzi, "Blind hyperspectral unmixing based on graph total variation regularization," *IEEE Trans Geosci Remote Sens*, vol. 59, no. 4, pp. 3338–3351, 2020.

[31] S. Arya, D. M. Mount, N. S. Netanyahu, R. Silverman, and A. Y. Wu, "An optimal algorithm for approximate nearest neighbor searching fixed dimensions," *Journal of the ACM*, vol. 45, no. 6, pp. 891–923, 1998.

[32] B. Chen, K. Miller, A. L. Bertozzi, and J. Schwenk, "Graph-based active learning for surface water and sediment detection in multispectral images," in *International Geoscience and Remote Sensing Symposium (IGARSS)*, IEEE, 2023.

[33] S. Ozkan, B. Kaya, and G. B. Akar, "Endnet: Sparse autoencoder network for endmember extraction and hyperspectral unmixing," *IEEE Trans Geosci Remote Sens*, vol. 57, no. 1, pp. 482–496, 2018.

[34] V. S. Deshpande, J. S. Bhatt, *et al.*, "A practical approach for hyperspectral unmixing using deep learning," *IEEE Geoscience and Remote Sensing Letters*, vol. 19, pp. 1–5, 2021.

[35] B. Rasti, B. Koirala, P. Scheunders, and J. Chanussot, "Misicnet: Minimum simplex convolutional network for deep hyperspectral unmixing," *IEEE Trans Geosci Remote Sens*, vol. 60, pp. 1–15, 2022.

[36] D. Hong, L. Gao, J. Yao, N. Yokoya, J. Chanussot, U. Heiden, and B. Zhang, "Endmember-guided unmixing network (egu-net): A general deep learning framework for self-supervised hyperspectral unmixing," *IEEE Trans. Neural Netw. Learn.*, vol. 33, no. 11, pp. 6518–6531, 2021.

[37] Z. Han, D. Hong, L. Gao, J. Yao, B. Zhang, and J. Chanussot, "Multimodal hyperspectral unmixing: Insights from attention networks," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, pp. 1–13, 2022.

[38] D. Hong, N. Yokoya, N. Ge, J. Chanussot, and X. X. Zhu, "Learnable manifold alignment (lema): A semi-supervised cross-modality learning framework for land cover and land use classification," *ISPRS J Photogramm Remote Sens*, vol. 147, pp. 193–205, 2019.

[39] D. Hong, N. Yokoya, J. Chanussot, J. Xu, and X. X. Zhu, "Learning to propagate labels on graphs: An iterative multitask regression framework for semi-supervised hyperspectral dimensionality reduction," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 158, pp. 35–49, 2019.

[40] S. Vijayashekhar, J. S. Bhatt, and B. Chattopadhyay, "Virtual dimensionality of hyperspectral data: Use of multiple hypothesis testing for controlling type-i error," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 13, pp. 2974–2985, 2020.

[41] J. S. Bhatt and M. V. Joshi, "Deep learning in hyperspectral unmixing: A review," in *IEEE International Geoscience and Remote Sensing Symposium*, pp. 2189–2192, IEEE, 2020.

[42] B. Zhang, Y. Wu, B. Zhao, J. Chanussot, D. Hong, J. Yao, and L. Gao, "Progress and challenges in intelligent remote sensing satellite systems," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 15, pp. 1814–1822, 2022.

[43] K. Miller, H. Li, and A. L. Bertozzi, "Efficient graph-based active learning with probit likelihood via Gaussian approximations," in *ICML Workshop on Experimental Design and Active Learning*, International Conference on Machine Learning (ICML), July 2020. arXiv: 2007.11126.

[44] K. Miller and A. L. Bertozzi, "Model-change active learning in graph-based semi-supervised learning," *arXiv preprint arXiv:2110.07739*, 2021.

[45] K. Miller, J. Mauro, J. Setiadi, X. Baca, Z. Shi, J. Calder, and A. L. Bertozzi, "Graph-based active learning for semi-supervised classification of SAR data," in *Algorithms for Synthetic Aperture Radar Imagery XXIX*, vol. 12095, pp. 126–139, SPIE, 2022.

[46] J. Li, J. M. Bioucas-Dias, and A. Plaza, "Semisupervised hyperspectral image segmentation using multinomial logistic regression with active learning," *IEEE Trans Geosci Remote Sens*, vol. 48, no. 11, pp. 4085–4098, 2010.

[47] S. Rajan, J. Ghosh, and M. M. Crawford, "An active learning approach to hyperspectral data classification," *IEEE Trans Geosci Remote Sens*, vol. 46, no. 4, pp. 1231–1242, 2008.

[48] J. M. Murphy and M. Maggioni, "Unsupervised clustering and active learning of hyperspectral images with nonlinear diffusion," *IEEE Trans Geosci Remote Sens*, vol. 57, no. 3, pp. 1829–1845, 2019.

[49] X. Cao, J. Yao, Z. Xu, and D. Meng, "Hyperspectral image classification with convolutional neural network and active learning," *IEEE Trans Geosci Remote Sens*, vol. 58, no. 7, pp. 4604–4616, 2020.

[50] B. Chen, K. Miller, A. L. Bertozzi, and J. Schwenk, "Batch active learning for multispectral and hyperspectral image segmentation using similarity graphs," *Communications on Applied Mathematics and Computation*, pp. 1–21, 2023.

[51] M. Ji and J. Han, "A variance minimization criterion to active learning on graphs," in *Artificial Intelligence and Statistics*, pp. 556–564, PMLR, 2012.

[52] Y. Qiao, C. Shi, C. Wang, H. Li, M. Haberland, X. Luo, A. M. Stuart, and A. L. Bertozzi, "Uncertainty quantification for semi-supervised multi-class classification in image processing and ego-motion analysis of body-worn videos," *Electronic Imaging*, vol. 2019, no. 11, pp. 264–1, 2019.

[53] M. Belkin, P. Niyogi, and V. Sindhwani, "Manifold regularization: A geometric framework for learning from labeled and unlabeled examples.," *J Mach Learn Res*, vol. 7, no. 11, 2006.

[54] U. Von Luxburg, "A tutorial on spectral clustering," *Statistics and computing*, vol. 17, no. 4, pp. 395–416, 2007.
[55] X. Zhu, Z. Ghahramani, and J. Lafferty, "Semi-supervised learning using gaussian fields and harmonic functions," in *Proceedings of the Twentieth International Conference on International Conference on Machine Learning*, ICML'03, p. 912–919, AAAI Press, 2003.
[56] J. Chapman, B. Chen, Z. Tan, J. Calder, K. Miller, and A. L. Bertozzi, "Novel batch active learning approach and its application on the synthetic aperture radar datasets," in *Algorithms for Synthetic Aperture Radar Imagery XXX*, vol. 12520, pp. 96–111, SPIE, 2023.
[57] W. Wang and M. A. Carreira-Perpinán, "Projection onto the probability simplex: An efficient algorithm with a simple proof, and an application," *arXiv preprint arXiv:1309.1541*, 2013.
[58] L. C. Evans, *Partial Differential Equations*. American Mathematical Society, 1998.
[59] B. Osting, C. D. White, and É. Oudet, "Minimal Dirichlet energy partitions for graphs," *SIAM J. Sci. Comput.*, vol. 36, no. 4, pp. A1635–A1651, 2014.
[60] J. Eckstein and D. P. Bertsekas, "On the Douglas—Rachford splitting method and the proximal point algorithm for maximal monotone operators," *Math. Program*, vol. 55, pp. 293–318, 1992.
[61] M. Hasanlou, "Remote sensing datasets." https://rslab.ut.ac.ir/data. Accessed: 2023-3-10.
[62] M. E. Schaepman, M. Jehle, A. Hueni, P. D'Odorico, A. Damm, J. Weyermann, F. D. Schneider, V. Laurent, C. Popp, F. C. Seidel, *et al.*, "Advanced radiometry measurements and earth science applications with the airborne prism experiment (apex)," *Remote Sensing of Environment*, vol. 158, pp. 207–219, 2015.
[63] J. M. Nascimento and J. M. Dias, "Vertex component analysis: A fast algorithm to unmix hyperspectral data," *IEEE Trans Geosci Remote Sens*, vol. 43, no. 4, pp. 898–910, 2005.

Bohan Chen received a B.S. degree in math from the school of mathematical sciences, Peking University, Beijing, China. He is currently a doctoral candidate in Mathematics at the University of California, Los Angeles. He received the UC-National Laboratory In-Residence Graduate Fellowship in 2021. After that, he conducted his research at both UCLA and Los Alamos National Laboratory (LANL). His research interests include optimal control problems, image processing, graph semi-supervised learning, and active learning.

Yifei Lou holds a joint position in the Department of Mathematics and the School of Data Science and Society (SDSS) at the University of North Carolina at Chapel Hill. She was on the faculty in the Mathematical Sciences Department at the University of Texas Dallas from 2014 to 2023, first as an Assistant Professor and then as an Associate Professor. She received her Ph.D. in Applied Math from the University of California Los Angeles (UCLA) in 2010. After graduation, she was a postdoctoral fellow at the School of Electrical and Computer Engineering Georgia Institute of Technology, followed by another postdoc training at the Department of Mathematics, University of California Irvine from 2012-2014. Dr. Lou received the National Science Foundation CAREER Award in 2019. Her research interests include compressive sensing and its applications, image analysis (medical, hyperspectral, and seismic imaging), and (nonconvex) optimization algorithms.

Andrea L. Bertozzi, *Member, IEEE* completed all her degrees in mathematics at Princeton. She is an Applied Mathematician with expertise in nonlinear partial differential equations and graphical models for machine learning. She was an L. E. Dickson Instructor and an NSF Postdoctoral Fellow at the University of Chicago from 1991 to 1995. She was the Maria Geoppert-Mayer Distinguished Scholar with the Argonne National Laboratory from 1995 to 1996. She was on the faculty at Duke University from 1995 to 2004, first as an Associate Professor of mathematics and then as a Professor of mathematics and physics. She has been on the faculty at UCLA since 2003 where she now holds the position of Distinguished Professor of Mathematics and Mechanical and Aerospace Engineering, along with the Betsy Wood Knapp Chair for Innovation and Creativity. She was elected to the American Academy of Arts and Sciences in 2010 and to the Fellows of the Society of Industrial and Applied Mathematics (SIAM) in 2010. She became a Fellow of the American Mathematical Society in 2013 and the American Physical Society in 2016. In 2018, she was elected to the U.S. National Academy of Sciences. Her honors include the Sloan Research Fellowship in 1995, the Presidential Early Career Award for Scientists and Engineers in 1996, and the SIAM Kovalevsky Prize in 2009, and the SIAM Kleinman Prize in 2019. She received the SIAM Outstanding Paper Prize in 2014 with A. Flenner, for her work on geometric graphbased algorithms for machine learning. She received the Simons Math + X Investigator Award in 2017.

Jocelyn Chanussot, *Fellow, IEEE* (M'04–SM'04–F'12) received the M.Sc. degree in electrical engineering from the Grenoble Institute of Technology (Grenoble INP), Grenoble, France, in 1995, and the Ph.D. degree from the Université de Savoie, Annecy, France, in 1998. Since 1999, he has been with Grenoble INP, where he is currently a Professor of signal and image processing. His research interests include image analysis, hyperspectral remote sensing, data fusion, machine learning and artificial intelligence. He has been a visiting scholar at Stanford University (USA), KTH (Sweden) and NUS (Singapore). Since 2013, he is an Adjunct Professor of the University of Iceland. In 2015-2017, he was a visiting professor at the University of California, Los Angeles (UCLA). He holds the AXA chair in remote sensing and is an Adjunct professor at the Chinese Academy of Sciences, Aerospace Information research Institute, Beijing. Dr. Chanussot is the founding President of IEEE Geoscience and Remote Sensing French chapter (2007-2010) which received the 2010 IEEE GRS-S Chapter Excellence Award. He has received multiple outstanding paper awards. He was the Vice-President of the IEEE Geoscience and Remote Sensing Society, in charge of meetings and symposia (2017-2019). He was the General Chair of the first IEEE GRSS Workshop on Hyperspectral Image and Signal Processing, Evolution in Remote sensing (WHISPERS). He was the Chair (2009-2011) and Cochair of the GRS Data Fusion Technical Committee (2005-2008). He was a member of the Machine Learning for Signal Processing Technical Committee of the IEEE Signal Processing Society (2006-2008) and the Program Chair of the IEEE International Workshop on Machine Learning for Signal Processing (2009). He is an Associate Editor for the IEEE Transactions on Geoscience and Remote Sensing, the IEEE Transactions on Image Processing and the Proceedings of the IEEE. He was the Editor-in-Chief of the IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing (2011-2015). In 2014 he served as a Guest Editor for the IEEE Signal Processing Magazine. He is a Fellow of the IEEE, an ELLIS Fellow, a Fellow of the Asia-Pacific Artificial Intelligence Association, a member of the Institut Universitaire de France (2012-2017) and a Highly Cited Researcher (Clarivate Analytics/Thomson Reuters, since 2018).