Research Article

Double Regression-Based Sparse Unmixing for Hyperspectral Images

Shuaiyang Zhang , Wenshen Hua , Gang Li, Jie Liu, Fuyu Huang, and Qianghui Wang

Department of Electronic and Optical Engineering, Army Engineering University of PLA, Shijiazhuang 050000, China

Correspondence should be addressed to Wenshen Hua; huaoptics@163.com

Received 24 February 2021; Revised 6 July 2021; Accepted 14 August 2021; Published 6 September 2021

Academic Editor: Penghai Wu

Copyright © 2021 Shuaiyang Zhang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Sparse unmixing has attracted widespread attention from researchers, and many effective unmixing algorithms have been proposed in recent years. However, most algorithms improve the unmixing accuracy at the cost of large calculations. Higher unmixing accuracy often leads to higher computational complexity. To solve this problem, we propose a novel double regression-based sparse unmixing model (DRSUM), which can obtain better unmixing results with lower computational complexity. DRSUM decomposes the complex objective function into two simple formulas and completes the unmixing process through two sparse regressions. The unmixing result of the first sparse regression is added as a constraint to the second. DRSUM is an open model, and we can add different constraints to improve the unmixing accuracy. In addition, we can perform appropriate preprocessing to further improve the unmixing results. Under this model, a specific algorithm called double regression-based sparse unmixing via K -means (DRSUM _{K -means}) is proposed. The improved K -means clustering algorithm is first used for preprocessing, and then we impose single sparsity and joint sparsity (using $l_{2,0}$ norm to control the sparsity) constraints on the first and second sparse unmixing, respectively. To meet the sparsity requirement, we introduce the row-hard-threshold function to solve the $l_{2,0}$ norm directly. Then, DRSUM _{K -means} can be efficiently solved under alternating direction method of multipliers (ADMM) framework. Simulated and real data experiments have proven the effectiveness of DRSUM _{K -means}.

1. Introduction

The hyperspectral remote sensing technology has shown great development and application potential in recent years [1–5]. However, due to spatial resolution and spatial complexity limitations, there are often many mixed pixels in the images, which seriously affect the processing and application of hyperspectral data. Therefore, spectral unmixing technology for mixed pixels has received widespread attention, which aims to identify the spectral features (endmembers) and estimate the corresponding abundance [6]. The linear unmixing model [7], which assumes that mixed pixels are linear combinations of endmembers, has been widely used for its simplicity. Under this model, unmixing algorithms are mainly categorized into three: statistic based, geometry based, and sparse regression based.

Sparse unmixing [8] is a semisupervised unmixing algorithm, which assumes that the mixed pixels are linear combinations of only a few endmembers from the spectral library. It does not need to extract endmembers and

estimate the number of endmembers. Numerous effective sparse unmixing algorithms have been proposed in recent years. The sparse unmixing algorithm via variable splitting and augmented Lagrangian (SUnSAL) [8] is a classic sparse unmixing algorithm. SUnSAL introduces new ideas into spectral unmixing. But the unmixing performance of SUnSAL is affected by the mutual correlation of spectral libraries, and the sparsity degree is insufficient. Collaborative SUnSAL (CLSUnSAL) [9] imposes joint sparsity constraint on hyperspectral images, which uses the $l_{2,0}$ norm to control sparsity. However, to avoid the nondeterministic polynomial hard (NP-hard) problem, CLSUnSAL uses convex relaxation strategy to solve the $l_{2,0}$ norm, which makes the sparsity insufficient in many scenarios. Reweighted sparse unmixing algorithm adopts the reweighting strategy to enhance the sparsity [10]. Collaborative sparse unmixing algorithm using l_0 norm (CSUnL0) [11] solves the $l_{2,0}$ norm directly and gets better sparsity than many convex relaxation algorithms.

In addition to sparsity, how to make full use of spatial information in the hyperspectral images is also an important direction to improve unmixing performance. Sparse unmixing via variable splitting augmented Lagrangian and total variation (SUnSAL-TV) [12] introduces a total variation regularizer to utilize the spatial information. Spectral-spatial weighted sparse unmixing (S²WSU) [13] introduces spatial information by spatial weighting factor. Reference [14] combines total variation regularizer and joint-sparse-blocks to improve unmixing accuracy. Reference [15] proposed row-sparsity spectral unmixing via total variation (RSSUn-TV) and achieved promising results. Reference [16] uses the $l_1 - l_2$ sparsity model and total variation regularizer to depict the sparse characteristic. The high spatial correlation of hyperspectral images is associated with the low column rank, and the low-rank constraint can make full use of the global structure of data. Therefore, the low-rank property can also be used to exploit spatial information. The alternating direction sparse and low-rank unmixing algorithm (ADSpLRU) [17] imposes the low-rank constraint on single sparsity model. Reference [18] incorporates the low-rank constraint into sparse unmixing to suppress noises and preserve more details. Reference [19] proposed nonconvex joint-sparsity and low-rank unmixing with dictionary pruning. Reference [20] combines the joint-sparse-blocks with low-rank constraint. Reference [21] proposed nonlocal low-rank prior sparse unmixing algorithm and achieved promising results.

The above algorithms can all obtain promising results. However, higher unmixing accuracy often leads to higher computational complexity. To solve this problem, we propose a novel double regression-based sparse unmixing model (DRSUM). DRSUM decomposes the complex unmixing process into two simple sparse regressions, which can reduce the computational complexity. An important contribution of this paper is that the proposed DRSUM is an open model and framework. Therefore, we can add different constraints on the first and second sparse regressions to improve unmixing accuracy. In addition, we can perform appropriate preprocessing to further improve the unmixing results. Under this model, a specific algorithm called double regression-based sparse unmixing via K -means (DRSUM _{K -means}) is proposed. We first use the improved K -means clustering algorithm [22] for preprocessing, which can classify pixels into different categories. Pixels belonging to the same category have the same active endmember set. Therefore, we use the mean vector of each category instead of all pixels for the first sparse unmixing. In other words, the number of pixels participating in the first sparse unmixing is the number of categories, which can greatly reduce the computational complexity. Then, we impose single sparsity constraint to unmix these mean vectors. The unmixing results of mean vectors are considered as abundance estimate of all pixels in the corresponding categories. After the first unmixing, we take the result as a constraint and impose joint sparsity (using $l_{2,0}$ norm to control the sparsity) constraint on the original hyperspectral image for the second unmixing. However, it is NP-hard to solve the $l_{2,0}$ norm directly [23]. Most algorithms adopt the convex relaxation strategy to solve the $l_{2,0}$ norm at the expense of unmixing

accuracy and sparsity. To address this issue, we introduce the row-hard-threshold function to solve the $l_{2,0}$ norm directly, which can improve the unmixing accuracy and sparsity. Simulated and real data experiments have proven that the proposed DRSUM _{K -means} algorithm can obtain better unmixing results.

2. Sparse Unmixing

Sparse unmixing is aimed at finding an optimal linear combination of endmembers from the spectral library for mixed pixels. Let $Y \in \mathbb{R}^{L \times n}$ represent a hyperspectral image, which contains n pixels and L spectral bands. The mixed model can be written as

$$Y = AX + N, \quad (1)$$

where $A \in \mathbb{R}^{L \times m}$ represents the known spectral library, which contains a total of m atoms (endmembers). $X \in \mathbb{R}^{m \times n}$ denotes the abundance matrix, which represents the abundance of these m endmembers in the hyperspectral image. $N \in \mathbb{R}^{L \times n}$ is the error term. However, the number of endmembers in the real hyperspectral image is far less than that of the endmembers in the spectral library. Most endmembers of the spectral library do not appear in the hyperspectral image. Therefore, most of the elements in the abundance matrix X are zero. In other words, there are only a few elements in X that are nonzero [24]. Therefore, the abundance matrix X is sparse, and we can impose sparsity constraint to solve equation (1). In addition, the abundance nonnegative constraint (ANC) and the abundance sum-to-one constraint (ASC) should also be met. It has been proven in [8] that the ANC will automatically impose a generalized ASC. Therefore, we only impose ANC and relax the ASC. Then, the sparse unmixing problem can be written as:

$$\min_X \frac{1}{2} \|Y - AX\|_F^2 + \lambda \|X\|_{1,0} \text{ s.t. } X \geq 0, \quad (2)$$

where $\|\cdot\|_F$ represents the Frobenius norm, and λ denotes the regularization parameter. $\|X\|_{1,0}$ represents the number of nonzero elements in X . $\geq$ denotes element-wise comparison. It is well known that solving equation (2) directly is NP-hard. The SUnSAL algorithm relaxes the $\|X\|_{1,0}$ norm to $\|X\|_{1,1}$ norm to solve it. SUnSAL is single-pixel sparse unmixing algorithm, and it is seriously affected by the mutual correlation of spectral libraries. CLSUnSAL imposes collaborative sparsity constraint to improve unmixing accuracy, and the unmixing problem is

$$\min_X \frac{1}{2} \|Y - AX\|_F^2 + \lambda \|X\|_{2,1} \text{ s.t. } X \geq 0, \quad (3)$$

where $\|X\|_{2,1} = \sum_{i=1}^m \|X_{i\cdot}\|_2$ and $X_{i\cdot}$ represent the i -th row of X . It is worth noting that $\|X\|_{2,1}$ is a convex relaxation of $\|X\|_{2,0}$ to avoid NP-hard problem. However, this will reduce the sparsity constraint and unmixing accuracy. Although SUnSAL and CLSUnSAL are faster than many algorithms,

their unmixing accuracy is insufficient. To obtain higher unmixing accuracy with low computational complexity, the double regression-based sparse unmixing model is proposed.

3. Double Regression-Based Sparse Unmixing Model

3.1. DRSUM. The core idea of DRSUM is to decompose the complex unmixing process into two simple sparse regressions. We can add different constraints on the first and second sparse unmixing to improve unmixing accuracy. In addition, we can perform appropriate preprocessing to further improve the unmixing results. The first sparse unmixing problem is as follows:

$$\min_{X_1} \frac{1}{2} \|Y - AX_1\|_F^2 + g(X_1) \text{ s.t. } X_1 \geq 0, \quad (4)$$

where Y denotes the hyperspectral image, and X_1 is the corresponding abundance matrix. $g(X_1)$ denotes the constraint item. After the first sparse unmixing, the obtained abundance matrix X_1 is added as a constraint to the second unmixing process. The second sparse unmixing problem is

$$\min_X \frac{1}{2} \|Y - AX\|_F^2 + \frac{\alpha}{2} \|X_1 - X\|_F^2 + f(X) \text{ s.t. } X \geq 0, \quad (5)$$

where α denotes the regularization parameter, and $f(X)$ denotes the constraint for X . There are no specific restrictions on $g(X_1)$ and $f(X)$. In different application scenarios, $g(X_1)$ and $f(X)$ can be different types of constraints to guarantee the unmixing accuracy, such as sparsity constraints, low-rank constraints, graph regularization constraints, total variation regularizer, nonlocal similarity, and image edge information. Therefore, DRSUM is flexible and has wide applicability.

3.2. DRSUM_{K-means}

3.2.1. Preprocessing. DRSUM is an open model and framework. Under this model, we can perform appropriate preprocessing and add different constraints to get different algorithms. In this paper, we first use the improved K -means clustering algorithm to classify the hyperspectral image. The K -means algorithm is simple and fast. It has only one parameter to be determined: the number of categories K . It should be noted that the original K -means algorithm classifies pixels only based on spectral distance, which ignores spatial information in the hyperspectral image. To improve classification accuracy, we consider both spectral distance d_1 and spatial distance d_2 to classify pixels. Therefore, the distance calculation formula for pixels i and j becomes

$$d_1(i, j) = \|y_i - y_j\|_2^2, \quad (6)$$

$$d_2(i, j) = \sqrt{(a_i - a_j)^2 + (b_i - b_j)^2}, \quad (7)$$

$$d'_1(i, j) = \frac{d_1(i, j)}{\max(d_1)}, d'_2(i, j) = \frac{d_2(i, j)}{\max(d_2)}, \quad (8)$$

$$D(i, j) = \sqrt{(d'_1(i, j))^2 + \rho(d'_2(i, j))^2}, \quad (9)$$

where y_i and y_j denote the spectral vectors. (a_i, b_i) and (a_j, b_j) denote the position of pixels i and j , respectively. To effectively control the weight, $d_1(i, j)$ and $d_2(i, j)$ are normalized to obtain $d'_1(i, j)$ and $d'_2(i, j)$. $\max(d_1)$ and $\max(d_2)$ denote the maximum value of spectral distance and spatial distance for any two pixels. ρ denotes the weight between spectral distance and spatial distance. To simplify the calculation, ρ is set to 1 in this paper. In each iteration, every pixel is assigned to the nearest cluster center. After each round of classification is completed, the cluster centers are updated by the average of the spectral vectors and position coordinates of all pixels in the corresponding categories. The classification process is completed when the cluster centers no longer change. More details about the K -means algorithm can be seen in reference [22].

After preprocessing, the pixels are classified into K categories. Pixels belonging to the same category have the same active endmember set. Therefore, we can use the mean vector of each category instead of all pixels for the first unmixing, which can greatly reduce the computational complexity. In other words, the number of pixels participating in the first sparse unmixing is the number of categories. In addition, the improved K -means clustering algorithm considers both spectral and spatial information for classification, which can improve the information utilization of hyperspectral images.

3.2.2. First Sparse Unmixing. After preprocessing, we calculate the mean vector of each category and impose single sparsity constraint to unmix these mean vectors. Then, we can get the following unmixing problem:

$$\min_{X_M} \frac{1}{2} \|Y_M - AX_M\|_F^2 + \lambda_1 \|X_M\|_{1,1} \text{ s.t. } X_M \geq 0, \quad (10)$$

where $Y_M = [y_1, y_2, \dots, y_K] \in R^{L \times K}$ denotes mean vectors of the K categories. X_M denotes the corresponding abundance matrix and $\|X_M\|_{1,1} = \sum_{i=1}^m \sum_{j=1}^K |x_{i,j}|$. λ_1 denotes the regularization parameter. Before solving equation (10), we introduce the indicator function to denote the nonnegative constraint, and the unmixing problem becomes

$$\min_{X_M} \frac{1}{2} \|Y_M - AX_M\|_F^2 + \lambda_1 \|X_M\|_{1,1} + I_{R^+}(X_M), \quad (11)$$

where $I_{R^+}(X_M)$ represents the indicator function, i.e., $I_{R^+}(X_M) = 0$ if $X_M \in R^+$ and $I_{R^+}(X_M) = +\infty$ otherwise. Then, equation (11) can be efficiently solved under alternating direction method of multipliers (ADMM) framework [25]. Set $V_1 = AX_M$, $V_2 = X_M$, $V_3 = X_M$, and then equation (11) can be written as

$$\begin{aligned} \min_{X_M, V_1, V_2, V_3} \quad & \frac{1}{2} \|Y_M - V_1\|_F^2 + \lambda_1 \|V_2\|_{1,1} + l_{R+}(V_3) \quad \text{s.t. } V_1 \\ & = AX_M, V_2 = X_M, V_3 = X_M. \end{aligned} \quad (12)$$

Set

$$g(X_M, V) = \frac{1}{2} \|Y_M - V_1\|_F^2 + \lambda_1 \|V_2\|_{1,1} + l_{R+}(V_3), \quad (13)$$

$$V \equiv (V_1, V_2, V_3), \quad (14)$$

$$B = \text{diag}(-I), \quad (15)$$

$$G = [A, I, I]^T. \quad (16)$$

Then, equation (12) becomes

$$\min_{X_M, V} g(X_M, V) \text{ s.t. } GX_M + BV = 0. \quad (17)$$

The following Lagrangian function can be constructed by equation (17)

$$L(X_M, V, D) \equiv g(X_M, V) + \frac{\mu}{2} \|GX_M + BV - D\|_F^2, \quad (18)$$

where $\mu > 0$ denotes a positive constant, and D/μ denotes the Lagrangian multiplier of the constraint $GX_M + BV = 0$. Under the ADMM framework, each variable is independent when updated. First, expand equation (18) to get

$$\begin{aligned} L(X_M, V_1, V_2, V_3, D_1, D_2, D_3) = & \frac{1}{2} \|Y_M - V_1\|_F^2 + \lambda_1 \|V_2\|_{1,1} \\ & + l_{R+}(V_3) + \frac{\mu}{2} \|AX_M - V_1 - D_1\|_F^2 \\ & + \frac{\mu}{2} \|X_M - V_2 - D_2\|_F^2 + \frac{\mu}{2} \|X_M - V_3 - D_3\|_F^2. \end{aligned} \quad (19)$$

When updating each variable, only the items related to it need to be considered. The objective optimization functions of variables X_M , V_1 , V_2 , and V_3 are

$$\begin{aligned} X_M^{(i+1)} = \underset{X_M}{\text{argmin}} \quad & \frac{\mu}{2} \|AX_M - V_1^{(i)} - D_1^{(i)}\|_F^2 \\ & + \frac{\mu}{2} \|X_M - V_2^{(i)} - D_2^{(i)}\|_F^2 + \frac{\mu}{2} \|X_M - V_3^{(i)} - D_3^{(i)}\|_F^2, \end{aligned} \quad (20)$$

$$V_1^{(i+1)} = \underset{V_1}{\text{argmin}} \frac{1}{2} \|Y_M - V_1\|_F^2 + \frac{\mu}{2} \|AX_M^{(i+1)} - V_1 - D_1^{(i)}\|_F^2, \quad (21)$$

$$V_2^{(i+1)} = \underset{V_2}{\text{argmin}} \lambda_1 \|V_2\|_{1,1} + \frac{\mu}{2} \|X_M^{(i+1)} - V_2 - D_2^{(i)}\|_F^2, \quad (22)$$

$$V_3^{(i+1)} = \underset{V_3}{\text{argmin}} l_{R+}(V_3) + \frac{\mu}{2} \|X_M^{(i+1)} - V_3 - D_3^{(i)}\|_F^2. \quad (23)$$

The corresponding solutions are

$$X_M^{(i+1)} = (A^T A + 2I)^{-1} (A^T (V_1^{(i)} + D_1^{(i)}) + V_2^{(i)} + D_2^{(i)} + V_3^{(i)} + D_3^{(i)}), \quad (24)$$

$$V_1^{(i+1)} = \frac{1}{1 + \mu} [Y_M + \mu (AX_M^{(i+1)} - D_1^{(i)})], \quad (25)$$

$$V_2^{(i+1)} = \text{soft} \left(X_M^{(i+1)} - D_2^{(i)}, \frac{\lambda_1}{\mu} \right), \quad (26)$$

$$V_3^{(i+1)} = \max \left(X_M^{(i+1)} - D_3^{(i)}, 0 \right), \quad (27)$$

where soft denotes the soft-threshold function $\text{soft}(y, \tau) = \text{sign}(y) \max\{|y| - \tau, 0\}$. Then, the final algorithm flow for the first sparse unmixing can be seen in Algorithm 1.

3.2.3. Second Sparse Unmixing. After the first sparse unmixing, we get the abundance estimate X_M of these mean vectors. Then, the abundance matrix $X_M \in \mathbb{R}^{m \times K}$ for these mean vectors is converted to the abundance matrix $X_1 \in \mathbb{R}^{m \times n}$ for the entire image. Specifically, the pixels in the same category have the same abundance estimate as the category's mean vector. Then, X_1 is added as a constraint for the second sparse unmixing, and we impose joint sparsity constraint on the original hyperspectral image. It is well known that solving the $l_{2,0}$ norm directly will cause NP-hard problem. Most algorithms use the $l_{2,1}$ norm instead of $l_{2,0}$ norm to avoid NP-hard problem. However, the $l_{2,1}$ norm just represents the sum of row-norms, which cannot meet the sparsity requirement sometimes. To address this issue, we introduce the row-hard-threshold function to solve the $l_{2,0}$ norm directly. Then, we can get the following unmixing problem:

$$\min_X \frac{1}{2} \|Y - AX\|_F^2 + \frac{\alpha}{2} \|X_1 - X\|_F^2 + \lambda_2 \|X\|_{2,0} \text{ s.t. } X \geq 0, \quad (28)$$

where λ_2 and α represent the regularization parameters, and the $l_{2,0}$ norm is defined as follows

$$\|X\|_{2,0} = \sum_{i=1}^m 1(\|X_{i\cdot}\|_2 > 0), \quad (29)$$

where $X_{i\cdot}$ denotes the i -th row of X . We also use the ADMM method to solve equation (28). Set $V_1 = AX$, $V_2 = X$, and $V_3 = X$. Then equation (28) can be written as

$$\begin{aligned} \min_{X, V_1, V_2, V_3} \quad & \frac{1}{2} \|Y - V_1\|_F^2 + \frac{\alpha}{2} \|X_1 - X\|_F^2 + \lambda_2 \|V_2\|_{2,0} + l_{R+}(V_3) \\ \text{s.t. } & V_1 = AX, V_2 = X, V_3 = X. \end{aligned} \quad (30)$$

We can construct the following Lagrangian function using the same method as equation (19)

1. Input: mean vectors matrix Y_M , spectral library A , constant μ .
2. Select parameters: λ_1 .
3. Initialization: $X_M^{(0)}, V^{(0)}, D^{(0)}$ and set $i = 0$.
4. Repeat:

$$X_M^{(i+1)} = (A^T A + 2I)^{-1} (A^T (V_1^{(i)} + D_1^{(i)}) + V_2^{(i)} + D_2^{(i)} + V_3^{(i)} + D_3^{(i)})$$

$$V_1^{(i+1)} = (1/1 + \mu) [Y_M + \mu (A X_M^{(i+1)} - D_1^{(i)})]$$

$$V_2^{(i+1)} = \text{soft}(X_M^{(i+1)} - D_2^{(i)}, (\lambda_1/\mu))$$

$$V_3^{(i+1)} = \max(X_M^{(i+1)} - D_3^{(i)}, 0)$$

$$D_1^{(i+1)} = D_1^{(i)} - A X_M^{(i+1)} + V_1^{(i+1)}$$

$$D_2^{(i+1)} = D_2^{(i)} - X_M^{(i+1)} + V_2^{(i+1)}$$

$$D_3^{(i+1)} = D_3^{(i)} - X_M^{(i+1)} + V_3^{(i+1)}$$

$$i = i + 1$$
5. Until the Stopping Condition Is Met

ALGORITHM 1: ADMM pseudocode for the first sparse unmixing.

$$\begin{aligned}
 L(X, V_1, V_2, V_3, D_1, D_2, D_3) = & \frac{1}{2} \|Y - V_1\|_F^2 + \frac{\alpha}{2} \|X_1 - X\|_F^2 \\
 & + \lambda_2 \|V_2\|_{2,0} + l_{R+}(V_3) + \frac{\mu}{2} \|AX - V_1 - D_1\|_F^2 \\
 & + \frac{\mu}{2} \|X - V_2 - D_2\|_F^2 + \frac{\mu}{2} \|X - V_3 - D_3\|_F^2.
 \end{aligned} \quad (31)$$

The iterative update formulas of variables X and V_1 are as follows

$$\begin{aligned}
 X^{(i+1)} = & (A^T A + (\alpha + \mu)I)^{-1} (A^T (V_1^{(i)} + D_1^{(i)}) \\
 & + V_2^{(i)} + D_2^{(i)} + V_3^{(i)} + D_3^{(i)} + \alpha X_1),
 \end{aligned} \quad (32)$$

$$V_1^{(i+1)} = \frac{1}{1 + \mu} [Y + \mu (A X^{(i+1)} - D_1^{(i)})]. \quad (33)$$

Before solving variable V_2 , we first remove items independent of it and get

$$V_2^{(i+1)} = \underset{V_2}{\operatorname{argmin}} \lambda_2 \|V_2\|_{2,0} + \frac{\mu}{2} \|X^{(i+1)} - V_2 - D_2^{(i)}\|_F^2. \quad (34)$$

Inspired by CSUnL0 algorithm [11], we introduce the row-hard-threshold function to solve equation (34). Define

$$(\text{RH}_t(X))_{i,j} = \begin{cases} x_{i,j} & \sum_j x_{i,j}^2 > t, \\ 0 & \sum_j x_{i,j}^2 \leq t, \end{cases} \quad (35)$$

where $(\text{RH}_t(X))_{i,j}$ represents the j -th element in the i -th row of the row-hard-threshold function $\text{RH}_t(X)$. $x_{i,j}$ denotes the j -th element in the i -th row of X and t denotes the threshold. It has been proved in reference [11] that $H = \text{RH}_t(X)$ is the closed-form solution for

$$\underset{H}{\operatorname{mint}} \|H\|_{2,0} + \|X - H\|_F^2. \quad (36)$$

Equation (34) can be written as follows

$$V_2^{(i+1)} = \underset{V_2}{\operatorname{argmin}} \frac{\mu}{2} \left(\frac{2\lambda_2}{\mu} \|V_2\|_{2,0} + \|X^{(i+1)} - V_2 - D_2^{(i)}\|_F^2 \right). \quad (37)$$

Then, we can get the iterative update formula of V_2 from equation (36) and (37).

$$V_2^{(i+1)} = \text{RH}_{2\lambda_2/\mu}(X^{(i+1)} - D_2^{(i)}). \quad (38)$$

The iterative update formulas of other variables are similar to those in the first sparse unmixing. Then, the final algorithm flow for the second sparse unmixing can be seen in Algorithm 2.

4. Simulated Data Experiments

In this section, we use two sets of simulated hyperspectral data to validate the effectiveness of the proposed DRSU $M_{K-\text{means}}$ algorithm. The famous USGS spectral library [26], which contains 498 substances and 224 bands, was used in this paper. Its spectral range is from 0.4 to 2.5 μm . For simplicity, we only use a subset $A_1 \in \mathbb{R}^{224 \times 240}$ of the USGS spectral library. The angle between any endmembers (Em) in A_1 is greater than 4.44 degrees. The DRSU $M_{K-\text{means}}$ algorithm was compared with five state-of-the-art algorithms: SUNSAL [8], CLSUnSAL [9], SUNSAL-TV [12], CSUnL0 [11], and $S^2\text{WSU-W1}$ [13]. The metrics signal-to-reconstruction error (SRE) and reconstruction overall root-mean-square error (rRMSE) [27] were used to evaluate the unmixing effect of these algorithms. Let x_i denote the true abundance vector and $\hat{x}_i$ an estimate of x_i . The metric SRE can be defined as follows

1. Input: hyperspectral image Y , spectral library A , abundance matrix X_1 , constant μ .
2. Select parameters: λ_2 and α .
3. Initialization: $X^{(0)}$, $V^{(0)}$, $D^{(0)}$ and set $i = 0$.
4. Repeat:

$$X^{(i+1)} = (A^T A + (\alpha + \mu)I)^{-1} (A^T (V_1^{(i)} + D_1^{(i)}) + V_2^{(i)} + D_2^{(i)} + V_3^{(i)} + D_3^{(i)} + \alpha X_1)$$

$$V_1^{(i+1)} = (1/\mu) [Y + \mu (AX^{(i+1)} - D_1^{(i)})]$$

$$V_2^{(i+1)} = \text{RH}_{2\lambda_2/\mu}(X^{(i+1)} - D_2^{(i)})$$

$$V_3^{(i+1)} = \max(X^{(i+1)} - D_3^{(i)}, 0)$$

$$D_1^{(i+1)} = D_1^{(i)} - AX^{(i+1)} + V_1^{(i+1)}$$

$$D_2^{(i+1)} = D_2^{(i)} - X^{(i+1)} + V_2^{(i+1)}$$

$$D_3^{(i+1)} = D_3^{(i)} - X^{(i+1)} + V_3^{(i+1)}$$

$$i = i + 1$$
- 5: **until** the stopping condition is met.

ALGORITHM 2: ADMM pseudocode for the second sparse unmixing.

$$\text{SRE}(\text{dB}) = 10 \log_{10} \left(\frac{E(\|x_i\|_2^2)}{E(\|x_i - \hat{x}_i\|_2^2)} \right), \quad (39)$$

where $E(\cdot)$ represents the expectation function. The larger the SRE (dB), the higher the unmixing accuracy. The metric rRMSE is defined by

$$\text{rRMSE} = \sqrt{\frac{1}{L n} \|Y - \hat{Y}\|_F^2}, \quad (40)$$

where $\hat{Y} = A\hat{X}$ represents the reconstruction image, and $\hat{X}$ represents the estimated abundance matrix. n and L represent the number of pixels and spectral bands, respectively. The smaller the rRMSE, the higher the unmixing accuracy.

4.1. Generation of Simulated Data

4.1.1. Simulated Data Cube 1 (DC1). DC1 contains 75×75 pixels. The abundance satisfied the ANC and ASC. Five endmembers were randomly selected from A_1 for this experiment: Jarosite GDS101, Anorthite HS349.3B, Calcite WS272, Microcline HS82.3B, and Howlite GDS155. Figure 1 shows the true abundance map of each endmember. To be more realistic and detect the antinoise ability of each algorithm, DC1 was contaminated with white Gaussian noise for three signal-to-noise ratio (SNR) levels: 20, 30, and 40 dB.

4.1.2. Simulated Data Cube 2 (DC2). A more complex dataset (DC2), which contains 100×100 pixels, was used for this experiment. We also used the A_1 spectral library and randomly selected 9 spectral signatures from it as the endmembers: Wollastonite HS348.3B, Heulandite GDS3, Mordenite GDS18, Almandine WS478, Nepheline HS19.3, Barite HS79.3B, Grossular WS485, Brookite HS443.2B, and Albite HS324.3B. Their corresponding abundance conforms to the Dirichlet distribution and is shown in Figure 2. We can see from Figure 2 that the endmember distribution in DC2 is more complicated than that in DC1. Both steep and smooth transitions exist in DC2, which is more in line

with the real distribution of objects. In addition, DC2 was also contaminated by white Gaussian noise with SNRs of 20, 30, and 40 dB.

4.2. Three Processes Analysis of $\text{DRSUM}_{K\text{-means}}$. In this section, we will analyze the three processes of $\text{DRSUM}_{K\text{-means}}$: preprocessing, the first sparse unmixing, and the second sparse unmixing. We first use the improved K -means clustering algorithm to classify pixels into different categories. The purpose of preprocessing is to improve the unmixing accuracy and reduce the computational complexity. The choice of K value is an important factor of preprocessing. If the value of K is too small, each category will contain more pixels. There may be large differences between pixels belonging to the same category. The standard deviation of each category will become larger. Then, pixels of the same category may have different active endmember set. In other words, it is inappropriate to use the mean vector of each category instead of all pixels for the first sparse unmixing. If the value of K is too large, the utilization of spatial information will decrease. Pixels with the same active endmember set may be classified into different categories. This will also affect the unmixing accuracy. Therefore, there should be a balance for the parameter K .

After preprocessing, we calculate the mean vector of each category and impose single sparsity constraint on these mean vectors for the first sparse unmixing. If there are large differences between pixels belonging to the same category, the results of the first sparse unmixing will be quite different from those of the real situation. Generally speaking, the larger the number of categories, the smaller the difference between pixels in the same category. However, as the number of categories increases, the utilization of spatial information will decrease and the calculation time will increase. Therefore, the classification results have a great influence on the first sparse unmixing.

After the first sparse unmixing, we get the abundance estimate X_M of these mean vectors. The abundance matrix $X_M \in \mathbb{R}^{m \times K}$ for these mean vectors is converted to the abundance matrix $X_1 \in \mathbb{R}^{m \times n}$ for the entire image. Then, X_1 is

FIGURE 1: True abundance maps of the endmembers (Em) in DC1.

FIGURE 2: True abundance maps of the endmembers (Em) in DC2.

added as a constraint for the second sparse unmixing, and we impose joint sparsity constraint on the original hyperspectral image. The abundance matrix X_1 can be regarded as a reference for the second sparse unmixing. The closer X_1 is to the true abundance, the higher the unmixing accuracy. Therefore, the first unmixing results can directly affect the second unmixing accuracy.

4.3. Parameters Selection. The $\text{DRSUM}_{K\text{-means}}$ algorithm contains three processes. We can know from Section 4.2 that the previous process has a great influence on the latter one. However, these three processes still have a certain degree of independence. Therefore, their parameters can be selected independently. When selecting some parameters, other parameters maintain appropriate values. The parameters that lead to the highest SRE values are selected. According to the algorithm flow, the number of categories K is first selected from the interval $[5, 200]$ with a step length of 5 for simulated data experiments. For real data experiment, the step length and interval are set to 10 and $[200, 700]$, respectively. The parameters λ_1 , λ_2 , and α are temporarily set to 0.01, 0.01, and 10, respectively. After K is selected, the parameter λ_1 is selected from the range $\{0.0001, 0.0005, 0.001, 0.005, 0.01, 0.05, 0.1\}$. Finally, the parameters λ_2 and α are selected from the ranges $\{0.0005, 0.001, 0.005, 0.01, 0.05, 0.1, 0.5, 1\}$ and $\{5, 10, 15, 20, 25, 30, 35, 40, 45,$

$50\}$, respectively. In addition, the constant μ is set to 0.01 for all experiments. Figure 3 presents the influence of parameters on the performance of $\text{DRSUM}_{K\text{-means}}$ for DC1 with SNR 30 dB. We can see that the parameters K , λ_1 , λ_2 , and α all have a great influence on the SRE value, which indicates that the preprocessing, the first, and second sparse unmixing can affect the unmixing effect greatly.

4.4. Comparison with Other Algorithms. In this section, we will compare the proposed $\text{DRSUM}_{K\text{-means}}$ algorithm with several state-of-the-art algorithms: SUnSAL [8], CLSUnSAL [9], SUnSAL-TV [12], CSUnL0 [11], and $S^2\text{WSU-W1}$ [13]. The parameters of all algorithms were constantly adjusted to obtain their optimal performance. The SRE and rRMSE values, the optimal parameter values, and the required time for all algorithms are shown in Tables 1 and 2. We can see that the $\text{DRSUM}_{K\text{-means}}$ algorithm has the highest SRE values and lowest rRMSE values under all the SNRs. Both CSUnL0 and $S^2\text{WSU-W1}$ algorithms can also achieve promising results. However, they are sensitive to noise and their unmixing accuracy is low when SNR is 20 dB. The SUnSAL-TV algorithm takes much more time than $\text{DRSUM}_{K\text{-means}}$, and its unmixing accuracy is still worse than that of $\text{DRSUM}_{K\text{-means}}$. Although the SUnSAL and CLSUnSAL algorithms are faster than $\text{DRSUM}_{K\text{-means}}$, their unmixing accuracy is much worse than that of $\text{DRSUM}_{K\text{-means}}$.

FIGURE 3: SRE (dB) as a function of parameters in $\text{DRSUM}_{K\text{-means}}$ for DC1 with SNR 30 dB. (a) Parameter K , (b) parameter λ_1 , (c) parameters λ_2 and α .

Figures 4 and 5 present the estimated abundance maps for DC1 and DC2 with an SNR of 30 dB, respectively. For space considerations, we only selected five endmembers from DC2 for comparison. It can be seen from Figure 4 that the SUnSAL, CLSUnSAL, CSUnL0, and S²WSU-W1 algorithms have more impurities in the background region than SUnSAL-TV and $\text{DRSUM}_{K\text{-means}}$ algorithms. The number of square region of SUnSAL-TV is less than that of $\text{DRSUM}_{K\text{-means}}$, especially for Em2. We can see from Figure 5 that the estimated abundance maps of SUnSAL and CLSUnSAL algorithms are quite different from the real ones. The estimated abundance maps of SUnSAL-TV, CSUnL0, S²WSU-W1, and $\text{DRSUM}_{K\text{-means}}$ algorithms are similar. However, it can still be seen that $\text{DRSUM}_{K\text{-means}}$ obtains higher unmixing accuracy than other algorithms, especially for Em1 and Em2. Therefore, the simulated data experiments

can validate the effectiveness of the proposed $\text{DRSUM}_{K\text{-means}}$ algorithm.

4.5. Computational Cost Analysis. To analyze the computational complexity of the proposed $\text{DRSUM}_{K\text{-means}}$ algorithm, all algorithms were run in MATLAB R2018a on a laptop equipped with an Intel Core 7 processor (2.3 GHz main frequency) and 16 GB of memory. The maximum number of iterations and the tolerance for all algorithms are set to 1000 and 10^{-4} , respectively. Tables 1 and 2 show the running time of all algorithms for DC1 and DC2. We can see that SUnSAL-TV takes the most time among all algorithms. S²WSU-W1 and CSUnL0 algorithms take much more time than $\text{DRSUM}_{K\text{-means}}$. Although $\text{DRSUM}_{K\text{-means}}$ takes a little more time than SUnSAL and CLSUnSAL algorithms, the unmixing effect of $\text{DRSUM}_{K\text{-means}}$ is much better

TABLE 1: SRE (dB), rRMSE, time (s), and parameters of each algorithm for DC1.

SNR	Criteria	SUnSAL	CLSUnSAL	SUnSAL-TV	CSUnL0	S ² WSU-W1	DRSUM _{K-means}
20 dB	SRE (dB)	4.5610	8.5435	9.4239	5.3514	7.8621	14.5899
	rRMSE	0.0053	0.0047	0.0045	0.0052	0.0049	0.0039
	Time (s)	3.95	4.14	88.07	28.25	30.32	5.33
	Parameters	$\lambda = 0.58$	$\lambda = 16$	$\lambda = 0.05$ $\lambda_{TV} = 0.05$	$a_0 = 0.04$	$\lambda = 0.2$	$K = 35, \lambda_1 = 0.005$ $\alpha = 40, \lambda_2 = 0.5$
30 dB	SRE (dB)	8.9168	11.1263	14.4408	13.5834	15.5173	17.2511
	rRMSE	0.0046	0.0043	0.0040	0.0041	0.0037	0.0035
	Time (s)	3.04	5.58	87.64	26.71	29.17	5.81
	Parameters	$\lambda = 0.09$	$\lambda = 2.8$	$\lambda = 0.007$ $\lambda_{TV} = 0.01$	$a_0 = 0.01$	$\lambda = 0.01$	$K = 90, \lambda_1 = 0.005$ $\alpha = 20, \lambda_2 = 0.05$
40 dB	SRE (dB)	12.6214	16.3085	21.1759	23.0446	24.7366	26.0369
	rRMSE	0.0042	0.0036	0.0031	0.0030	0.0028	0.0026
	Time (s)	3.28	5.62	85.92	33.31	31.45	6.19
	Parameters	$\lambda = 0.05$	$\lambda = 0.1$	$\lambda = 0.001$ $\lambda_{TV} = 0.005$	$a_0 = 0.005$	$\lambda = 0.005$	$K = 120, \lambda_1 = 0.001$ $\alpha = 10, \lambda_2 = 0.01$

TABLE 2: SRE (dB), rRMSE, time (s), and parameters of each algorithm for DC2.

SNR	Criteria	SUnSAL	CLSUnSAL	SUnSAL-TV	CSUnL0	S ² WSU-W1	DRSUM _{K-means}
20 dB	SRE (dB)	4.2673	3.4011	10.9791	6.8322	9.1641	18.7322
	rRMSE	0.0044	0.0046	0.0037	0.0042	0.0039	0.0019
	Time (s)	5.80	13.51	114.62	49.84	52.36	10.82
	Parameters	$\lambda = 0.1$	$\lambda = 2.2$	$\lambda = 0.01$ $\lambda_{TV} = 0.03$	$a_0 = 0.05$	$\lambda = 0.05$	$K = 35, \lambda_1 = 0.005$ $\alpha = 40, \lambda_2 = 0.5$
30 dB	SRE (dB)	10.4529	7.3505	17.8078	17.4521	19.2785	21.3605
	rRMSE	0.0038	0.0041	0.0022	0.0022	0.0017	0.0013
	Time (s)	7.74	16.04	115.17	51.06	53.49	12.36
	Parameters	$\lambda = 0.01$	$\lambda = 0.3$	$\lambda = 0.005$ $\lambda_{TV} = 0.007$	$a_0 = 0.005$	$\lambda = 0.03$	$K = 100, \lambda_1 = 0.005$ $\alpha = 5, \lambda_2 = 0.01$
40 dB	SRE (dB)	13.3724	12.7233	20.6522	21.2153	23.8638	25.1639
	rRMSE	0.0031	0.0033	0.0015	0.0013	0.00011	0.0009
	Time (s)	6.33	11.28	116.34	46.59	51.08	12.68
	Parameters	$\lambda = 0.005$	$\lambda = 0.05$	$\lambda = 0.001$ $\lambda_{TV} = 0.005$	$a_0 = 0.001$	$\lambda = 0.007$	$K = 150, \lambda_1 = 0.001$ $\alpha = 5, \lambda_2 = 0.005$

than that of them. Therefore, the DRSUM_{K-means} algorithm can obtain promising results with low computational complexity. It should be noted that the time required by SUnSAL-TV is several times longer than that of DRSUM_{K-means}. Therefore, though DRSUM_{K-means} has more parameters to be set, the time required by DRSUM_{K-means} for selecting parameters is still less than that of SUnSAL-TV.

4.6. Convergence Analysis. After preprocessing, we impose single sparsity constraint on these mean vectors for the first sparse unmixing. From equation (10), we can know that the first sparse unmixing is similar to the SUnSAL algorithm. Reference [8] has proved that the SUnSAL algorithm is convergent. Therefore, we can conclude that the first sparse unmixing is also convergent. From equation (28), we can

know that the result of the first sparse unmixing and the $l_{2,0}$ norm is introduced as constraints for the second sparse unmixing. It is difficult to give the theoretical convergence analysis of equation (28) directly. Figure 6 presents the convergence curves of equation (28) for DC1 and DC2 with different SNRs. We can see from Figure 6 that the SRE value no longer changes when the number of iterations reaches a certain value. Therefore, the residuals and the maximum number of iterations can be used as iteration stop conditions of DRSUM_{K-means}. In all experiments, the tolerance and the maximum number of iterations for DRSUM_{K-means} are set to 10^{-4} and 1000, respectively. In addition, it is worth noting that though convergence is not theoretically guaranteed, and the DRSUM_{K-means} algorithm exhibits a robust convergence behavior.

FIGURE 4: Abundance maps for DC1 (from top to bottom: Em1 to Em5) with SNR = 30 dB.

5. Real Data Experiment

The public Cuprite dataset (<http://aviris.jpl.nasa.gov/html/aviris.freedata.html>) was used for this experiment. It is representative and has been widely used for hyperspectral unmixing experiments. The Cuprite dataset was collected by the Airborne Visible Infrared Imaging Spectrometer in 1997. To simplify the calculation, we used a 250×191 subset of the Cuprite dataset, similarly as in [28–30]. It contains 224 spectral bands, but after removing the low-SNR and high-water-absorption bands (1–2, 105–115, 150–170, 223–224), 188 spectral bands were retained. The whole USGS spectral library $A \in \mathbb{R}^{224 \times 498}$ was used. Figure 7 shows the distribution of different substances in Cuprite dataset. It was produced in 1995 by the USGS, and it is an important reference to qualitatively evaluate the unmixing performance. We selected three representative endmembers (buddingtonite, alunite, and chalcedony) from this dataset and compared the abundance maps estimated by each algorithm with those generated by USGS software Tricorder 3.3², which are presented in Figure 8. The parameters of SUNSAL,

CLSUNSAL, SUNSAL-TV, CSUnL0, and S^2 WSU-W1 are set according to references [9, 11, 13]. Table 3 presents the parameter values, processing time, and rRMSE values of each algorithm for the Cuprite dataset.

We can see from Figure 8 that the backgrounds of SUNSAL and SUNSAL-TV algorithms are not as pure and clear as those of other algorithms, especially for the mineral buddingtonite. There are also more oversmoothing and blurred edges phenomenon in the abundance maps of SUNSAL-TV. CLSUNSAL and CSUnL0 algorithms have worse abundance maps than other algorithms for the mineral chalcedony. The unmixing effects of S^2 WSU-W1 and DRSUM_{K-means} algorithms are similar, but it can still be seen that DRSUM_{K-means} has higher abundance for the mineral chalcedony and is closer to the true distribution. In addition, it can be seen from Table 3 that the DRSUM_{K-means} algorithm has the lowest rRMSE value among all algorithms. That is to say, DRSUM_{K-means} has the highest unmixing accuracy for the real dataset. Therefore, the real dataset can also prove the effectiveness of the proposed DRSUM_{K-means} algorithm.

FIGURE 5: Abundance maps for DC2 (from top to bottom: Em1 to Em5) with SNR = 30 dB.

FIGURE 6: Convergence curves of DRSUM_{K-means} for DC1 and DC2. (a) DC1 and (b) DC2.

FIGURE 7: Distribution map of different substances for Cuprite dataset.

FIGURE 8: Abundance maps estimated for (from top to bottom) alunite, buddingtonite, and chalcedony.

TABLE 3: The rRMSE values, time (s), and parameters of each algorithm for Cuprite dataset.

Criteria	SUnSAL	CLSUnSAL	SUnSAL-TV	CSUnL0	S ² WSU-W1	DRSUM _{K-means}
rRMSE	$8.7e-5$	$8.3e-5$	$5.9e-5$	$7.8e-5$	$3.6e-5$	$2.1e-5$
Time (s)	217.3	194.5	1441.4	631.5	824.7	322.5
Parameters	$\lambda = 0.001$	$\lambda = 0.01$	$\lambda = 0.001$ $\lambda_{TV} = 0.001$	$a_0 = 6e-5$	$\lambda = 0.002$	$K = 390$, $\lambda_1 = 0.001$ $\alpha = 5$, $\lambda_2 = 0.001$

6. Conclusions

In this paper, a novel double regression-based sparse unmixing model (DRSUM) was proposed. DRSUM decomposes the complex unmixing process into two simple sparse regressions, which can reduce the computational complexity. DRSUM is an open model and framework. In different application scenarios, we can add different types of constraints into DRSUM to guarantee the unmixing accuracy. In addition, we can perform appropriate preprocessing to further improve the unmixing results. Under this model, a specific algorithm called double regression-based sparse unmixing via K -means (DRSUM_{K-means}) is proposed. By appropriate preprocessing, DRSUM_{K-means} can comprehensively utilize spatial information, spectral information, and sparsity. Simulated and real data experiments have proven that DRSUM_{K-means} can obtain better unmixing results with lower computational complexity.

Data Availability

The MATLAB code and dataset used to support the findings of this study have been deposited in the Zenodo repository (doi:10.5281/zenodo.4052818).

Conflicts of Interest

The authors declare that there is no conflict of interest regarding the publication of this paper.

Acknowledgments

This research was supported by the Department of Electronic and Optical Engineering, Shijiazhuang Campus, AEU.

References

- [1] G. Shaw and D. Manolakis, "Signal processing for hyperspectral image exploitation," *IEEE Signal Processing Magazine*, vol. 19, no. 1, pp. 12–16, 2002.
- [2] D. Landgrebe, "Hyperspectral image data analysis," *IEEE Signal Processing Magazine*, vol. 19, no. 1, pp. 17–28, 2002.
- [3] R. C. Murdock, O. A. Khan, T. J. Lamkin, S. M. Hussain, and N. Kelley-Loughnane, "Hybridization state detection of DNA-functionalized gold nanoparticles using hyperspectral imaging," *International Journal of Optics*, vol. 2017, Article ID 8427459, 12 pages, 2017.
- [4] D. Zhang, T. Tiyip, J. Ding et al., "Quantitative estimating salt content of saline soil using laboratory hyperspectral data treated by fractional derivative," *Journal of Spectroscopy*, vol. 2016, Article ID 1081674, 11 pages, 2016.
- [5] W. K. Ma, J. Bioucas-Dias, T. H. Chan et al., "A signal processing perspective on hyperspectral unmixing: insights from remote sensing," *IEEE Signal Processing Magazine*, vol. 31, no. 1, pp. 67–81, 2014.
- [6] N. Keshava and J. F. Mustard, "Spectral unmixing," *IEEE Signal Processing Magazine*, vol. 19, no. 1, pp. 44–57, 2002.
- [7] J. M. Bioucas-Dias, A. Plaza, N. Dobigeon et al., "Hyperspectral unmixing overview: geometrical, statistical, and sparse regression-based approaches," *IEEE journal of selected topics in applied earth observations and remote sensing*, vol. 5, no. 2, pp. 354–379, 2012.
- [8] M.-D. Iordache, J. M. Bioucas-Dias, and A. Plaza, "Sparse unmixing of hyperspectral data," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 49, no. 6, pp. 2014–2039, 2011.
- [9] M.-D. Iordache, J. M. Bioucas-Dias, and A. Plaza, "Collaborative sparse regression for hyperspectral unmixing," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 52, no. 1, pp. 341–354, 2013.
- [10] C. Y. Zheng, H. Li, Q. Wang, and C. P. Chen, "Reweighted sparse regression for hyperspectral unmixing," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 4, no. 1, pp. 479–488, 2015.
- [11] Z. Shi, T. Shi, M. Zhou, and X. Xu, "Collaborative sparse hyperspectral unmixing using $sl_{\{0\}}$ norm," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 56, no. 9, pp. 5495–5508, 2018.
- [12] M.-D. Iordache, J. M. Bioucas-Dias, and A. Plaza, "Total variation spatial regularization for sparse hyperspectral unmixing," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 50, no. 11, pp. 4484–4502, 2012.
- [13] S. Zhang, J. Li, H.-C. Li, C. Deng, and A. Plaza, "Spectral-spatial weighted sparse regression for hyperspectral image unmixing," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 56, no. 6, pp. 3265–3276, 2018.
- [14] J. Huang, T.-Z. Huang, X.-L. Zhao, and L. J. Deng, "Joint-sparse-blocks regression for total variation regularized hyperspectral unmixing," *IEEE Access*, vol. 7, pp. 138779–138791, 2019.
- [15] J. J. Wang, T. Z. Huang, J. Huang, H. X. Dou, L. J. Deng, and X. L. Zhao, "Row-sparsity spectral unmixing via total variation," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 12, no. 12, pp. 5009–5022, 2019.
- [16] L. Sun, W. Ge, Y. Chen, J. Zhang, and B. Jeon, "Hyperspectral unmixing employing l_1 - l_2 sparsity and total variation regularization," *International Journal of Remote Sensing*, vol. 39, no. 19, pp. 6037–6060, 2018.
- [17] P. V. Giampouras, K. E. Themelis, A. A. Rontogiannis, and K. D. Koutroumbas, "Simultaneously sparse and low-rank abundance matrix estimation for hyperspectral image unmixing," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 54, no. 8, pp. 4775–4789, 2016.

- [18] Y. Xu, "Sparse abundance estimation with low-rank reconstruction for hyperspectral unmixing," *International Journal of Remote Sensing*, vol. 41, no. 17, pp. 6805–6830, 2020.
- [19] H. Han, G. Wang, M. Wang et al., "Hyperspectral unmixing via nonconvex sparse and low-rank constraint," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 13, pp. 5704–5718, 2020.
- [20] J. Huang, T. Z. Huang, L. J. Deng, and X. L. Zhao, "Joint-sparse-blocks and low-rank representation for hyperspectral unmixing," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 57, no. 4, pp. 2419–2438, 2018.
- [21] Y. Zheng, F. Wu, H. J. Shim, and L. Sun, "Sparse unmixing for hyperspectral image with nonlocal low-rank prior," *Remote Sensing*, vol. 11, no. 24, 2019.
- [22] T. Kanungo, D. M. Mount, N. S. Netanyahu, C. D. Piatko, R. Silverman, and A. Y. Wu, "An efficient k-means clustering algorithm: analysis and implementation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 7, pp. 881–892, 2002.
- [23] B. K. Natarajan, "Sparse approximate solutions to linear systems," *SIAM Journal on Computing*, vol. 24, no. 2, pp. 227–234, 1995.
- [24] R. Ammanouil, A. Ferrari, C. Richard, and D. Mary, "Blind and fully constrained unmixing of hyperspectral images," *IEEE Transactions on Image Processing*, vol. 23, no. 12, pp. 5510–5518, 2014.
- [25] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. E. Kstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Found Trends Mach Learn*, vol. 3, no. 1, pp. 1–122, 2010.
- [26] R. G. Clark, R. Swayze, K. E. Livo et al., "USGS digital spectral library Splib06a: digital data series 231," September 2007. Available online: <http://speclab.cr.usgs.gov/spectral.lib06>.
- [27] T. Ahmad, R. B. Lyngdoh, A. S. Raha, P. K. Gupta, and A. Misra, "Four-directional spatial regularization for sparse hyperspectral unmixing," *Journal of Applied Remote Sensing*, vol. 14, no. 4, 2020.
- [28] R. A. Borsoi, T. Imbiriba, J. C. M. Bermudez, and C. Richard, "A fast multiscale spatial regularization for sparse hyperspectral unmixing," *IEEE Geoscience and Remote Sensing Letters*, vol. 16, no. 4, pp. 598–602, 2019.
- [29] H. Li, R. Feng, L. Wang, Y. Zhong, and L. Zhang, "Super-pixel-based reweighted low-rank and total variation sparse unmixing for hyperspectral remote sensing imagery," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 59, no. 1, pp. 629–647, 2020.
- [30] S. Zhang, J. Li, K. Liu, C. Deng, L. Liu, and A. Plaza, "Hyperspectral unmixing based on local collaborative sparse regression," *IEEE Geoscience and Remote Sensing Letters*, vol. 13, no. 5, pp. 631–635, 2016.