



A two-step iterative algorithm for sparse hyperspectral unmixing via total variation[☆]

Jin-Ju Wang, Ting-Zhu Huang*, Jie Huang, Liang-Jian Deng

School of Mathematical Sciences, University of Electronic Science and Technology of China, Chengdu, Sichuan 611731, PR China



ARTICLE INFO

Article history:

Received 8 December 2018

Revised 17 January 2021

Accepted 1 February 2021

Available online 26 February 2021

Keywords:

Sparse unmixing

Hyperspectral images

Two-step iterative strategy

Total variation (TV)

Weighted $\ell_{2,1}$ regularization

ABSTRACT

Sparse hyperspectral unmixing is a hot topic in the field of remote sensing. Its goal is to find an optimal spectral subset, from a large spectral library, to properly model the mixed pixels in hyperspectral images. Sparse unmixing via variable splitting augmented Lagrangian and total variation (SUnSAL-TV) incorporates a TV regularizer into sparse unmixing, achieving a promising unmixing performance. Note that, SUnSAL-TV is solved by the framework of the alternating direction method of multipliers (ADMM). In this paper, we first propose a weighted collaborative sparse unmixing via TV model, named as WCSU-TV, for hyperspectral unmixing. Then a two-step iterative strategy, based on ADMM, is designed to solve the proposed model. Its key idea is to compute the current solution by a linear combination of the results of two previous iterates, instead of only using current solution in classic ADMM. Experiments on simulated and real hyperspectral data illustrate the effectiveness of the proposed approach.

© 2021 Elsevier Inc. All rights reserved.

1. Introduction

Hyperspectral images contain a lot both spectral information and spatial information. It has been widely used in military target detection and other fields [1–3]. Due to the spatial resolution of sensors and the diversity of ground objects, many pixels in the hyperspectral image contain more than one material, and thus are viewed as mixed pixels [4]. In this case, spectral unmixing, decomposing mixed pixels into a collection of constituent materials called *endmembers* and their corresponding fractions called *abundances*, becomes an important technology for investigating hyperspectral images [5].

To illustrate the mixing process of mixed pixels, both linear and nonlinear mixture models are studied for hyperspectral unmixing [4,6,7]. The linear mixture model (LMM) assumes that the spectral response of a pixel can be modeled as a linear combination of the endmembers, weighted by their fractional abundances. It has been widely used for practical advantages in simplicity and feasibility. Therefore, our proposed unmixing method are based on the LMM.

Recently, replacing the collection of endmembers selected from images with known spectral libraries has fostered a large amount of researches [8–12]. It avoids assuming the existence of pure endmembers and estimating the number of endmembers [6]. Since the number of spectral signatures in known spectral libraries is much larger than the number of endmembers in the mixed pixels, the sparsity is incorporated into spectral unmixing. Then the sparse unmixing scheme,

[☆] This research is supported in part by NSFC (Grant No. 61772003), in part by Key Projects of Applied Basic Research in Sichuan Province (Grant No. 2020YJ0216), in part by National Key Research and Development Program of China (Grant No. 2020YFA0714001), and in part by the Fundamental Research Funds for the Central Universities (Grant No. ZYGX2019J093). (Corresponding authors: Ting-Zhu Huang, Jie Huang)

* Corresponding author.

E-mail addresses: jinjuwang123@163.com (J.-J. Wang), tingzhuhuang@126.com (T.-Z. Huang), huangjie_uestc@uestc.edu.cn (J. Huang).

termed as SUnSAL, is proposed under the assumption that there are only a few endmembers in a mixed pixel compared with the large spectral library [8]. In addition, to promote the sparsity in the solution, an ℓ_p regularization is utilized to replace the classic ℓ_1 regularization in SUnSAL [10]. Furthermore, an iterative weighted ℓ_1 regularization is developed to obtain more accurate abundance estimations [13].

Besides the sparsity assumption, spatial information is utilized for a better hyperspectral unmixing performance [9,12,14–18]. Typically, the TV regularization is included into the SUnSAL model, which is termed as SUnSAL-TV, to exploit the spatial-contextual information [12]. It requires that both the mixing materials and the associated abundances should be similar for neighboring pixels, which is a rather strict assumption in some cases [19]. A collaborative sparse scheme, termed as CLSUnSAL, relaxes this assumption of the TV regularization [9]. It assumes that adjacent pixels are composed of similar materials but do not necessarily have similar abundance for each material.

Subsequently, nonlocal sparse unmixing (NLSU) adopts the nonlocal means as a spatial regularizer for sparse unmixing, exploiting similar patterns and structures in the abundance maps [15]. Although NLSU takes advantage of high-order structural information, the neighborhood of the pixel changes randomly, limiting the continuity of spectral information. In addition, sparse unmixing using spectral a priori information (SUnSPI) integrates the priori knowledge of the hyperspectral data into the framework of hyperspectral unmixing [16].

Furthermore, the low rankness of the abundances is used to exploit the high correlation of the endmembers existing in hyperspectral images [17,18]. In addition, to model second-order photon scattering effects, various bilinear mixture models (BMMs) have been introduced to capture such nonlinearities [19–21]. For example, the generalized bilinear model (GBM) generalizes the LMM by introducing bilinear terms that take into account the multipath effects [22]. These unmixing algorithms exploit both sparsity and spatial information and provide promising unmixing performance.

In this paper, we propose a weighted collaborative sparse unmixing via TV model, named as WCSU-TV, for hyperspectral unmixing. Recall that a hyperspectral image only contains a few endmembers in the spectral library. It means that the abundance matrix includes a amount of zero rows (called collaborative sparsity). Then, the proposed model introduces an iterative weighted $\ell_{2,1}$ regularization to describe the collaborative sparsity of the abundance matrix. Meanwhile, we also adopt the TV term to exploit the neighboring correlation in hyperspectral images. To solve the proposed model, we design a two-step iterative strategy, under the framework of *alternating direction method of multipliers* (ADMM). Its key idea is to compute the current solution by a linear combination of the results of previous two-step iterates instead of only the former iteration. We call the proposed algorithm as two-step WCSU-TV (TSWCSU-TV).

The structure of this paper is outlined as follows. Section II introduces some sparse unmixing models. Section III proposes the given model and designs a new unmixing algorithm to solve it. The effectiveness of the proposed algorithm is demonstrated by both simulated experiments in Section IV and a real data test in Section V. Finally, Section VI gives some conclusions.

2. Sparse unmixing model

Given a hyperspectral image $\mathbf{Y} = [y_1, y_2, \dots, y_n] \in \mathbb{R}^{L \times n}$ with L bands and n pixels and a spectral library $\mathbf{A} \in \mathbb{R}^{L \times m}$ with m endmembers, the LMM can be formulated as

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

where $\mathbf{X} = [x_1, x_2, \dots, x_n] \in \mathbb{R}^{m \times n}$ is the abundance matrix and $\mathbf{N} \in \mathbb{R}^{L \times n}$ is the system noise. In the literature, \mathbf{X} often satisfies the *abundance non-negativity constraint* (ANC, i.e., $\mathbf{X} \geq \mathbf{0}$) and the *abundance sum-to-one constraint* (ASC, i.e., $\mathbf{1}_m^T \mathbf{X} = \mathbf{1}_n^T$) [23].

Notice that, the number of endmembers in the spectral library \mathbf{A} is often much larger than the number of endmembers in a mixed pixel, then it follows that \mathbf{X} is sparse. In this case, sparsity becomes an important constraint for the hyperspectral unmixing problem [8]. It has been also revealed that the accuracy and solution stability of the unmixing models can be improved with other priori information [12,17,24].

Typically, a sparse unmixing model can be expressed as

$$\min_{\mathbf{X}} \frac{1}{2} \|\mathbf{AX} - \mathbf{Y}\|_F^2 + \lambda \|\mathbf{X}\|_0 \quad s.t. \quad \mathbf{X} \geq \mathbf{0}, \quad (2)$$

where $\|\cdot\|_F$ represents the Frobenius norm, $\lambda \geq 0$ is the regularization parameter, and $\|\mathbf{X}\|_0$ represents the ℓ_0 quasi-norm of \mathbf{X} . It is known that the ℓ_0 minimization problem is non-convex and non-Lipschitz, which is therefore difficult to solve [25]. One popular strategy is to transform the ℓ_0 minimization problem into an ℓ_p ($0 < p < 1$) one in recent years [26–28]. And by exploiting the spatial information in hyperspectral images, we attain the following general model:

$$\min_{\mathbf{X}} \frac{1}{2} \|\mathbf{AX} - \mathbf{Y}\|_F^2 + \lambda \|\mathbf{X}\|_{p,1} + \lambda_{TV} \mathbf{TV}(\mathbf{X}) \quad s.t. \quad \mathbf{X} \geq \mathbf{0}, \quad (3)$$

where $\lambda \geq 0$, $\lambda_{TV} \geq 0$ are regularization parameters and $\mathbf{TV}(\mathbf{X})$ is the TV operator of the abundance maps. We note that the anisotropic version and the isotropic version are two types of the TV regularizers. As said in [29,30], the isotropic version behaves better in remaining the regions with non-zero abundances, while the anisotropic version behaves better in removing small regions dominated by non-zeros abundances. Considering the existence of the noise in hyperspectral images, we adopt the anisotropic TV regularizer, similarly as in [12,14].

In the following, with different values of λ , λ_{TV} , and p , we obtain different unmixing models:

1. Let $p = 1$ and $\lambda_{TV} = 0$, then the model in (3) becomes the SUnSAL model in [8], where $\|\mathbf{X}\|_{1,1} = \sum_{j=1}^n \|x_j\|_1$, and x_j represents the j -th column of the abundance matrix \mathbf{X} . It promotes sparsity in each column of \mathbf{X} .
2. Let $p = 2$ and $\lambda_{TV} = 0$, then the model in (3) becomes the CLSUnSAL model in [9], where $\|\mathbf{X}\|_{2,1} = \sum_{k=1}^m \|x^k\|_2$, and x^k represents the k -th row of \mathbf{X} , for $k = 1, \dots, m$. It promotes the sparsity of \mathbf{X} along the line.
3. Let $p = 1$ and $\lambda_{TV} \neq 0$, then the model in (3) becomes the SUnSAL-TV model in [12]. It promotes not only the sparsity in each column but also the piecewise constant transitions in each abundance map.

3. Proposed unmixing algorithm

3.1. WCSU-TV

The TV regularizer assumes that two neighboring pixels have similar fractional abundances for the same endmember, which improves the unmixing performance (see [12,14,18]). Recall that in SUnSAL-TV, the TV regularization is combined with the ℓ_1 regularization. In many practical cases, however, a hyperspectral image always contains a small number of endmembers [9]. It says that the $\ell_{2,1}$ regularization, instead of the ℓ_1 one, can be used to promote the collaborative sparsity of the abundance matrix. Therefore, in this paper, we combine the $\ell_{2,1}$ regularization with the TV term, obtaining the following model:

$$\min_{\mathbf{X}} \frac{1}{2} \|\mathbf{AX} - \mathbf{Y}\|_F^2 + \lambda \|\mathbf{X}\|_{2,1} + \lambda_{TV} \mathbf{TV}(\mathbf{X}) \quad s.t. \quad \mathbf{X} \geq \mathbf{0}. \quad (4)$$

However, the $\ell_{2,1}$ norm is a convex relaxation of the $\ell_{2,0}$ norm and is hard to guarantee the sparsity. Thus, we adopt a weighting matrix to enhance the collaborative sparsity of the abundance matrix. It is as follows:

$$\|\mathbf{W} \odot \mathbf{X}\|_{2,1} = \sum_{k=1}^m w^k \|x^k\|_2,$$

where \odot represents the element-wise multiplication (i.e., Hadamard product) of two variables, $\mathbf{W} = \text{diag}(w^1, \dots, w^m)$, and $w^k, k = 1, \dots, m$ is the weighting factor of the i th row in \mathbf{X} . When $w^k = \frac{1}{\|x^k\|_2 + \epsilon}$, $k = 1, \dots, m$, then

$$w^i \|x^k\|_2 = \frac{\|x^k\|_2}{\|x^k\|_2 + \epsilon} \begin{cases} = 0, & \text{if } \|x^k\|_2 = 0, \\ \approx 1, & \text{if } \|x^k\|_2 \neq 0, \end{cases}$$

which is close to the definition of the $\ell_{2,0}$ norm. Then, for hyperspectral unmixing, we obtain a weighted collaborative sparse unmixing via TV model, named WCSU-TV, as follows:

$$\min_{\mathbf{X}} \frac{1}{2} \|\mathbf{AX} - \mathbf{Y}\|_F^2 + \lambda \|\mathbf{W} \odot \mathbf{X}\|_{2,1} + \lambda_{TV} \mathbf{TV}(\mathbf{X}) \quad s.t. \quad \mathbf{X} \geq \mathbf{0}. \quad (5)$$

Finally, we recall that in the literature, it is effective to introduce weighting factors to enforce the sparsity of the nonzero coefficients of the solution; see [11,13,31]. Especially, the weighted ℓ_1 regularization [32] and the weighted $\ell_{2,1}$ regularization [24] have been used for the hyperspectral unmixing problem.

3.2. TSWCSU-TV

We now solve the proposed model in (5). Recall that the ADMM is a convex optimization algorithm, which is a variant of the classic augmented Lagrangian method. It solves convex optimization problems by breaking them into smaller pieces, so that each subproblem is easier to handle [33]. It has recently found wide applications in a number of areas such as machine learning and image processing; see [34–38] and references therein. Note that, the current solution in ADMM only depends on the results of former iteration. For linear inverse problems, a fast iterative shrinkage-thresholding algorithm (FISTA) computes the current solution by a linear combination of the two previous iterates [39]. Similar strategies have been also used for image restoration in [40,41]. Inspired by those algorithms, we introduce a two-step iterative strategy under the ADMM framework to solve (5) to further exploit the iteration information. That means, the current solution of each subproblem in ADMM is computed by a linear combination of the results of two previous iterates.

We begin with applying the ADMM idea to (5). To this end, we first equivalently rewrite (5) by some substitutions, see as follows,

$$\begin{aligned} \min_{\mathbf{X}, \mathbf{V}_1, \mathbf{V}_2, \mathbf{V}_3, \mathbf{V}_4, \mathbf{V}_5} \quad & \frac{1}{2} \|\mathbf{V}_1 - \mathbf{Y}\|_F^2 + \lambda \|\mathbf{W} \odot \mathbf{V}_2\|_{2,1} + \lambda_{TV} \|\mathbf{V}_4\|_{1,1} + \iota_{R_+}(\mathbf{V}_5) \\ \text{s.t.} \quad & \mathbf{A}\mathbf{X} = \mathbf{V}_1 \\ & \mathbf{X} = \mathbf{V}_2 \\ & \mathbf{X} = \mathbf{V}_3 \\ & \mathbf{H}\mathbf{V}_3 = \mathbf{V}_4 \\ & \mathbf{X} = \mathbf{V}_5, \end{aligned} \quad (6)$$

where \mathbf{H} includes two linear operators computing the horizontal and vertical differences, respectively [12] and $\iota_{R_+}(\mathbf{X}) = \sum_{i=1}^n \iota_{R_+}(x_i)$ represents the non-negative constraint of $\mathbf{X} \geq \mathbf{0}$. For $x_i \geq 0$, $\iota_{R_+}(x_i) = 0$, otherwise, $\iota_{R_+}(x_i) = \infty$. Define

$$\mathbf{V} = \begin{pmatrix} \mathbf{V}_1 \\ \mathbf{V}_2 \\ \mathbf{V}_3 \\ \mathbf{V}_4 \\ \mathbf{V}_5 \end{pmatrix}, \mathbf{G} = \begin{pmatrix} \mathbf{A} \\ \mathbf{I} \\ \mathbf{I} \\ \mathbf{0} \\ \mathbf{I} \end{pmatrix}, \mathbf{B} = \begin{pmatrix} -\mathbf{I} & 0 & 0 & 0 & 0 \\ 0 & -\mathbf{I} & 0 & 0 & 0 \\ 0 & 0 & -\mathbf{I} & 0 & 0 \\ 0 & 0 & \mathbf{H} & -\mathbf{I} & 0 \\ 0 & 0 & 0 & 0 & -\mathbf{I} \end{pmatrix},$$

and

$$g(\mathbf{V}) = \frac{1}{2} \|\mathbf{V}_1 - \mathbf{Y}\|_F^2 + \lambda \|\mathbf{W} \odot \mathbf{V}_2\|_{2,1} + \lambda_{TV} \|\mathbf{V}_4\|_{1,1} + l_{R_+}(\mathbf{V}_5).$$

Then, we reformulate (6) to a compact form

$$\min_{\mathbf{X}, \mathbf{V}} g(\mathbf{V}) \quad s.t. \quad \mathbf{GX} + \mathbf{BV} = \mathbf{0}.$$

To solve the above model by ADMM, here we need to introduce Lagrange multipliers

$$\mathbf{D} = \begin{pmatrix} \mathbf{D}_1 \\ \mathbf{D}_2 \\ \mathbf{D}_3 \\ \mathbf{D}_4 \\ \mathbf{D}_5 \end{pmatrix},$$

and define

$$\mathbf{L}(\mathbf{X}, \mathbf{V}, \mathbf{D}) = g(\mathbf{V}) + \frac{\mu}{2} \|\mathbf{GX} + \mathbf{BV} - \mathbf{D}\|_F^2,$$

where $\mu > 0$ is a penalty parameter. Then, we propose a two-step iterative strategy under the ADMM framework, that means, to minimize $\mathbf{L}(\mathbf{X}, \mathbf{V}, \mathbf{D})$ with respect to \mathbf{X} and \mathbf{V} and update \mathbf{D} as the following framework:

$$\begin{cases} \mathbf{X}^{(t+1)} \leftarrow \arg \min_{\mathbf{X}} \mathbf{L}(\mathbf{X}, \mathbf{V}^{(t)}, \mathbf{D}^{(t)}) \\ \hat{\mathbf{X}}^{(t+1)} \leftarrow \alpha \mathbf{X}^{(t+1)} + (1 - \alpha) \mathbf{X}^{(t)} \quad (0 < \alpha < 1) \\ \mathbf{V}^{(t+1)} \leftarrow \arg \min_{\mathbf{V}} \mathbf{L}(\hat{\mathbf{X}}^{(t+1)}, \mathbf{V}, \mathbf{D}^{(t)}) \\ \mathbf{D}^{(t+1)} \leftarrow \mathbf{D}^{(t)} - \mathbf{G}\hat{\mathbf{X}}^{(t+1)} - \mathbf{B}\mathbf{V}^{(t+1)}. \end{cases}$$

In the following, we compute each $\mathbf{X}, \mathbf{V}, \mathbf{D}$ -subproblems by the above-mentioned framework. For the \mathbf{X} subproblem, first, the optimization problem is

$$\mathbf{X}^{(t+1)} \leftarrow \arg \min_{\mathbf{X}} \frac{\mu}{2} \|\mathbf{AX} - \mathbf{V}_1^{(t)} - \mathbf{D}_1^{(t)}\|_F^2 + \frac{\mu}{2} \|\mathbf{X} - \mathbf{V}_2^{(t)} - \mathbf{D}_2^{(t)}\|_F^2 + \frac{\mu}{2} \|\mathbf{X} - \mathbf{V}_3^{(t)} - \mathbf{D}_3^{(t)}\|_F^2 + \frac{\mu}{2} \|\mathbf{X} - \mathbf{V}_5^{(t)} - \mathbf{D}_5^{(t)}\|_F^2.$$

It is easy to obtain that

$$\mathbf{X}^{(t+1)} = (\mathbf{A}^T \mathbf{A} + 3\mathbf{I})^{-1} (\mathbf{A}^T (\mathbf{V}_1^{(t)} + \mathbf{D}_1^{(t)}) + (\mathbf{V}_2^{(t)} + \mathbf{D}_2^{(t)}) + (\mathbf{V}_3^{(t)} + \mathbf{D}_3^{(t)}) + (\mathbf{V}_5^{(t)} + \mathbf{D}_5^{(t)})).$$

Then we have $\hat{\mathbf{X}}^{(t+1)} \leftarrow \alpha \mathbf{X}^{(t+1)} + (1 - \alpha) \mathbf{X}^{(t)}$, for $0 < \alpha < 1$.

Next, we decouple the \mathbf{V} subproblem into five parts. To compute \mathbf{V}_1 , we have the optimization problem

$$\mathbf{V}_1^{(t+1)} \leftarrow \arg \min_{\mathbf{V}_1} \frac{1}{2} \|\mathbf{V}_1 - \mathbf{Y}\|_F^2 + \frac{\mu}{2} \|\mathbf{A}\hat{\mathbf{X}}^{(t+1)} - \mathbf{V}_1 - \mathbf{D}_1^{(t)}\|_F^2.$$

A simple calculation gives

$$\mathbf{V}_1^{(t+1)} \leftarrow \frac{1}{1 + \mu} (\mathbf{Y} + \mu (\mathbf{A}\hat{\mathbf{X}}^{(t+1)} - \mathbf{D}_1^{(t)})).$$

Then, \mathbf{V}_2 is computed by solving the optimization problem

$$\mathbf{V}_2^{(t+1)} \leftarrow \arg \min_{\mathbf{V}_2} \lambda \|\mathbf{W} \odot \mathbf{V}_2\|_{2,1} + \frac{\mu}{2} \|\hat{\mathbf{X}}^{(t+1)} - \mathbf{V}_2 - \mathbf{D}_2^{(t)}\|_F^2.$$

Here, we adopt a reweighting strategy. That is, the current weighting matrix \mathbf{W} , i.e., $\mathbf{W}^{(t)}$, is computed from the matrices of the former iteration:

$$\mathbf{W}^{(t)} = \text{diag} \left(\frac{1}{\|(\hat{\mathbf{X}}^{(t+1)} - \mathbf{D}_2^{(t)})(1, :)\|_2 + \varepsilon}, \dots, \frac{1}{\|(\hat{\mathbf{X}}^{(t+1)} - \mathbf{D}_2^{(t)})(m, :)\|_2 + \varepsilon} \right),$$

in which $\varepsilon > 0$ is a stabilization parameter and $\mathbf{B}(i, :)$ represents the i th row of \mathbf{B} for any matrix \mathbf{B} . Thus, we get

$$\mathbf{V}_2^{(t+1)} \leftarrow \mathbf{T}_{row}^{VST} (\hat{\mathbf{X}}^{(t+1)} - \mathbf{D}_2^{(t)}, \frac{\lambda}{\mu} \mathbf{W}^{(t)}),$$

where the vector-soft thresholding (VST) operator $\mathbf{T}_{\text{row}}^{\text{VST}}(\mathbf{b}, \tau)$ is defined as

$$\mathbf{T}_{\text{row}}^{\text{VST}}(\mathbf{b}, \tau) \leftarrow \mathbf{b} \frac{\max\{\|\mathbf{b}\|_2 - \tau, 0\}}{\max\{\|\mathbf{b}\|_2 + \tau, 0\} + \tau}, \quad \mathbf{b} \in \mathbb{R}^n.$$

It denotes the row-by-row implementation of the *vectorial soft-threshold* function [42,43].

In order to compute \mathbf{V}_3 , we have

$$\mathbf{V}_3^{(t+1)} \leftarrow \arg \min_{\mathbf{V}_3} \frac{\mu}{2} \|\hat{\mathbf{X}}^{(t+1)} - \mathbf{V}_3 - \mathbf{D}_3^{(t)}\|_F^2 + \frac{\mu}{2} \|\mathbf{H}\mathbf{V}_3 - \mathbf{V}_4^{(t)} - \mathbf{D}_4^{(t)}\|_F^2.$$

It is easy to obtain that

$$\mathbf{V}_3^{(t+1)} \leftarrow (\mathbf{H}^T \mathbf{H} + \mathbf{I})^{-1} (\mathbf{H}^T (\mathbf{V}_4^{(t)} + \mathbf{D}_4^{(t)}) + (\hat{\mathbf{X}}^{(t+1)} - \mathbf{D}_3^{(t)})).$$

For the \mathbf{V}_4 subproblem, we have

$$\mathbf{V}_4^{(t+1)} \leftarrow \arg \min_{\mathbf{V}_4} \lambda_{TV} \|\mathbf{V}_4\|_{1,1} + \frac{\mu}{2} \|\mathbf{H}\mathbf{V}_3^{(t)} - \mathbf{V}_4 - \mathbf{D}_4^{(t)}\|_F^2,$$

Then it is easy to get

$$\mathbf{V}_4^{(t+1)} \leftarrow \text{soft}(\mathbf{D}_4^{(t)} - \mathbf{H}\mathbf{V}_3^{(t+1)}, \frac{\lambda_{TV}}{\mu}),$$

where $\text{soft}(u, \tau)$, defined by

$$\text{soft}(u, \tau) \leftarrow \text{sign}(u) \max\{|u| - \tau, 0\},$$

is a component-wise application of the *soft-threshold* function in [44,45].

Finally, \mathbf{V}_5 is computed by solving the following optimization problem:

$$\mathbf{V}_5^{(t+1)} \leftarrow \arg \min_{\mathbf{V}_5} \iota_{R_+}(\mathbf{V}_5) + \frac{\mu}{2} \|\hat{\mathbf{X}}^{(t+1)} - \mathbf{V}_5 - \mathbf{D}_5^{(t)}\|_F^2,$$

and the solution is

$$\mathbf{V}_5^{(t+1)} \leftarrow \max(\hat{\mathbf{X}}^{(t+1)} - \mathbf{D}_5^{(t)}, \mathbf{0}).$$

Hence, the optimization for \mathbf{V} is performed as:

$$\begin{cases} \mathbf{V}_1^{(t+1)} \leftarrow \frac{1}{1+\mu} (\mathbf{Y} + \mu(\mathbf{A}\hat{\mathbf{X}}^{(t+1)} - \mathbf{D}_1^{(t)})) \\ \mathbf{V}_2^{(t+1)} \leftarrow \mathbf{T}_{\text{row}}^{\text{VST}}(\hat{\mathbf{X}}^{(t+1)} - \mathbf{D}_2^{(t)}, \frac{\lambda}{\mu} \mathbf{W}^{(t)}) \\ \mathbf{V}_3^{(t+1)} \leftarrow (\mathbf{H}^T \mathbf{H} + \mathbf{I})^{-1} (\mathbf{A}^T (\mathbf{V}_4^{(t)} + \mathbf{D}_4^{(t)}) + (\hat{\mathbf{X}}^{(t+1)} - \mathbf{D}_3^{(t)})) \\ \mathbf{V}_4^{(t+1)} \leftarrow \text{soft}(\mathbf{D}_4^{(t)} - \mathbf{H}\mathbf{V}_3^{(t+1)}, \frac{\lambda_{TV}}{\mu}) \\ \mathbf{V}_5^{(t+1)} \leftarrow \max(\hat{\mathbf{X}}^{(t+1)} - \mathbf{D}_5^{(t)}, \mathbf{0}). \end{cases}$$

Finally, we update all Lagrange multipliers as follows

$$\begin{cases} \mathbf{D}_1^{(t+1)} \leftarrow \mathbf{D}_1^{(t)} - (\mathbf{A}\hat{\mathbf{X}}^{(t+1)} - \mathbf{V}_1^{(t+1)}) \\ \mathbf{D}_2^{(t+1)} \leftarrow \mathbf{D}_2^{(t)} - (\hat{\mathbf{X}}^{(t+1)} - \mathbf{V}_2^{(t+1)}) \\ \mathbf{D}_3^{(t+1)} \leftarrow \mathbf{D}_3^{(t)} - (\hat{\mathbf{X}}^{(t+1)} - \mathbf{V}_3^{(t+1)}) \\ \mathbf{D}_4^{(t+1)} \leftarrow \mathbf{D}_4^{(t)} - (\hat{\mathbf{X}}^{(t+1)} - \mathbf{V}_4^{(t+1)}). \end{cases}$$

To make it more clearly, we summarize the proposed algorithm, named as two-step WCSU-TV (TSWCSU-TV) algorithm in [Algorithm 1](#).

4. Synthetic data experiments

In this section, we demonstrate the effectiveness of TSWCSU-TV on two synthetic hyperspectral data cubes. We will compare the proposed algorithm with three state-of-the-art unmixing algorithms: SUoSAL [8], CLSUnSAL [9], and SUoSAL-TV [12]. All tests are implemented on the platform of Windows 7 and MATLAB (R2016a) with an Intel Core i5-4590, 3.30GHz and 8GB RAM.

4.1. Generation of synthetic data

We generate two simulated hyperspectral data cubes with the following two spectral libraries. The first spectral library, $\mathbf{A}_1 \in \mathbb{R}^{224 \times 240}$, is generated by selecting 240 materials (different mineral types) from the United States Geological Survey (USGS) library¹ randomly. It includes spectral signatures with reflectance values given in 224 spectral bands and distributed

¹ Available online: <http://speclab.cr.usgs.gov/spectral.lib06>

Algorithm 1 The TSWCSU-TV algorithm.

01) **Initialization:** $t = 0, \mathbf{X}_0, \mathbf{V}_0, \mathbf{D}_0, \varepsilon > 0, \alpha \in (0, 1)$
 02) Set $\lambda, \lambda_{TV}, \mu > 0$, and maximum number of iterations
Repeat:

- 03) $\mathbf{X}^{(t+1)} \leftarrow (\mathbf{A}^T \mathbf{A} + 3\mathbf{I})^{-1} (\mathbf{A}^T (\mathbf{V}_1^{(t)} + \mathbf{D}_1^{(t)}) + (\mathbf{V}_2^{(t)} + \mathbf{D}_2^{(t)}))$
- 04) $\hat{\mathbf{X}}^{(t+1)} = \alpha \mathbf{X}^{(t+1)} + (1 - \alpha) \mathbf{X}^{(t)}$
- 05) $\mathbf{V}_1^{(t+1)} \leftarrow \frac{1}{1+\mu} (\mathbf{Y} + \mu (\hat{\mathbf{X}}^{(t+1)} - \mathbf{D}_1^{(t)}))$
- 06) $\mathbf{W}^{(t)} = \text{diag} \left(\frac{1}{\|(\hat{\mathbf{X}}^{(t+1)} - \mathbf{D}_2^{(t)})(1,:) \|_2 + \varepsilon}, \dots, \frac{1}{\|(\hat{\mathbf{X}}^{(t+1)} - \mathbf{D}_2^{(t)})(m,:) \|_2 + \varepsilon} \right)$
- 07) $\mathbf{V}_2^{(t+1)} \leftarrow \mathbf{T}_{row}^{VST} (\hat{\mathbf{X}}^{(t+1)} - \mathbf{D}_2^{(t)}, \frac{\lambda}{\mu} \mathbf{W}^{(t)})$
- 08) $\mathbf{V}_3^{(t+1)} \leftarrow (\mathbf{H}^T \mathbf{H} + \mathbf{I})^{-1} (\mathbf{A}^T (\mathbf{V}_4^{(t)} + \mathbf{D}_4^{(t)}) + (\hat{\mathbf{X}}^{(t+1)} - \mathbf{D}_3^{(t)}))$
- 09) $\mathbf{V}_4^{(t+1)} \leftarrow \text{soft}(\mathbf{D}_4^{(t)} - \mathbf{H}\mathbf{V}_3^{(t+1)}, \frac{\lambda_{TV}}{\mu})$
- 10) $\mathbf{V}_5^{(t+1)} \leftarrow \max(\hat{\mathbf{X}}^{(t+1)} - \mathbf{D}_5^{(t)}, \mathbf{0})$

Update the Lagrange multipliers:

- 11) $\mathbf{D}_1^{(t+1)} \leftarrow \mathbf{D}_1^{(t)} - (\mathbf{A} \hat{\mathbf{X}}^{(t+1)} - \mathbf{V}_1^{(t+1)})$
- 12) $\mathbf{D}_2^{(t+1)} \leftarrow \mathbf{D}_2^{(t)} - (\hat{\mathbf{X}}^{(t+1)} - \mathbf{V}_2^{(t+1)})$
- 13) $\mathbf{D}_3^{(t+1)} \leftarrow \mathbf{D}_3^{(t)} - (\hat{\mathbf{X}}^{(t+1)} - \mathbf{V}_3^{(t+1)})$
- 14) $\mathbf{D}_4^{(t+1)} \leftarrow \mathbf{D}_4^{(t)} - (\hat{\mathbf{X}}^{(t+1)} - \mathbf{V}_4^{(t+1)})$

Update Number of iterations:

- 15) $t \leftarrow t + 1$

Until some termination condition is met.

uniformly in the interval $0.4 \mu\text{m} - 2.5 \mu\text{m}$. The other spectral library matrix, $\mathbf{A}_2 \in \mathbb{R}^{100 \times 120}$, is generated by using a library of 262 spectral signatures generally found on satellites, from the National Aeronautics and Space Administration Johnson Space Center (NASA JSC)[46], with 100 spectral bands. Then two data cubes are built as follows:

1. *Simulated Data Cube 1* (DC1). We select five spectral signatures randomly from \mathbf{A}_1 as the endmembers to simulate the DC1 with 75×75 pixels and 224 bands per pixel. Fig. 1 shows the true fractional abundances for each of the five end-members.
2. *Simulated Data Cube 1* (DC2). Nine signatures are randomly chosen, and a 100×100 pixels datacube of true observations is generated. Fig. 2 shows the corresponding true abundances of the endmembers.

After generating DC1 and DC2 following the procedure described above, we contaminate the scene by adding Gaussian noise with the signal to noise ratio (SNR), i.e.,

$$\text{SNR} = \frac{\mathbb{E}[\|\mathbf{Ax}\|^2]}{\mathbb{E}[\|\mathbf{N}\|_2^2]},$$

of 30 dB, 40 dB, and 50 dB, respectively. Here, \mathbb{E} denotes the expected value. We use the signal to reconstruction error (SRE), measured in dB:

$$\text{SRE(dB)} = 10 \log_{10} \frac{\mathbb{E}[\|x\|_2^2]}{\mathbb{E}[\|x - \hat{x}\|_2^2]},$$

to measure the quality of the reconstruction of spectral mixtures [8]. Here, x denotes the true fractional abundances of endmembers, and \hat{x} is the computed fractional abundances of endmembers by the proposed algorithm. Generally, the higher the SRE (dB) values, the better the unmixing performance. In addition, the regularization parameters of each algorithm are tuned to their highest SRE (dB) values. We choose optimal λ and λ_{TV} from a finite set {0.001, 0.005, 0.01, 0.05, 0.1, 0.5} and optimal μ from {0.01, 0.05, 0.1, 0.5} for both DC1 and DC2 data. All possible combinations of these parameters are considered. For each experiment, the maximum iteration number is set to 300.

4.2. Experiments and results

In the following, four experiments are conducted to evaluate the proposed algorithm on the above-mentioned DC1 and DC2.

4.2.1. The two-step iterative algorithm analysis

The first experiment is to choose a suitable α for TSWCSU-TV. We consider four different values of α : $\alpha = 1, \frac{1}{2}, \frac{1}{3}$, and $\frac{1}{4}$, for both DC1 and DC2. We recall that TSWCSU-TV with $\alpha = 1$ is equivalent to solving WCSU-TV with classic ADMM.

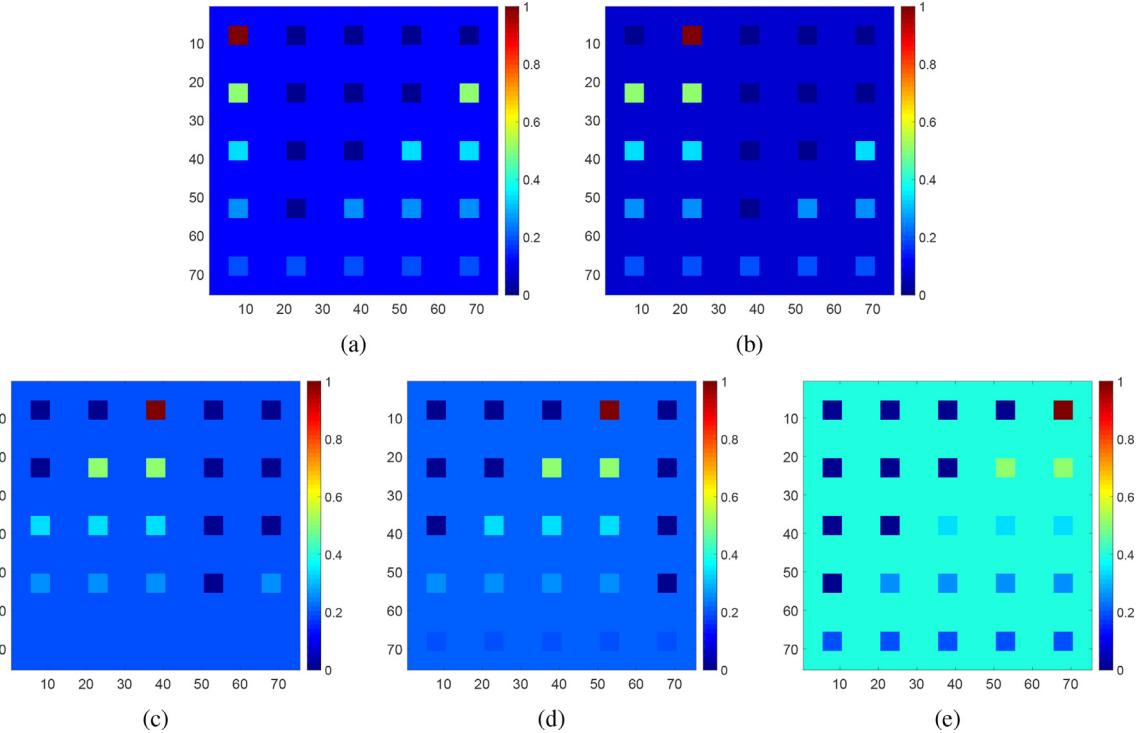


Fig. 1. True fractional abundances of five endmembers for DC1. (a) Endmember #1; (b) Endmember #2; (c) Endmember #3; (d) Endmember #4; (e) Endmember #5.

Table 1
SRE(dB) values by TSWCSU-TV with different values of α for DC1 and DC2 (optimal parameters for which the reported values were achieved are indicated in the parentheses).

Data cube	SNR (dB)	TSWCSU-TV			
		$\alpha = 1$ (WCSU-TV)	$\alpha = \frac{1}{2}$	$\alpha = \frac{1}{3}$	$\alpha = \frac{1}{4}$
DC1	30	19.43 ($\lambda = 0.001$, $\lambda_{TV} = 0.001$)	20.99 ($\lambda = 0.5$, $\lambda_{TV} = 0.01$)	21.41 ($\lambda = 0.5$, $\lambda_{TV} = 0.01$)	21.48 ($\lambda = 0.5$, $\lambda_{TV} = 0.01$)
		28.57 ($\lambda = 0.5$, $\lambda_{TV} = 0.001$)	27.85 ($\lambda = 0.5$, $\lambda_{TV} = 0.001$)	29.10 ($\lambda = 0.5$, $\lambda_{TV} = 0.001$)	29.97 ($\lambda = 0.5$, $\lambda_{TV} = 0.001$)
	40	30.45 ($\lambda = 0.001$, $\lambda_{TV} = 0.001$)	32.32 ($\lambda = 0.005$, $\lambda_{TV} = 0.001$)	35.96 ($\lambda = 0.005$, $\lambda_{TV} = 0.001$)	36.78 ($\lambda = 0.005$, $\lambda_{TV} = 0.001$)
		31.12 ($\lambda = 0.001$, $\lambda_{TV} = 0.001$)	33.00 ($\lambda = 0.005$, $\lambda_{TV} = 0.001$)	36.68 ($\lambda = 0.005$, $\lambda_{TV} = 0.001$)	37.55 ($\lambda = 0.005$, $\lambda_{TV} = 0.001$)
	50	31.80 ($\lambda = 0.001$, $\lambda_{TV} = 0.001$)	33.68 ($\lambda = 0.005$, $\lambda_{TV} = 0.001$)	37.36 ($\lambda = 0.005$, $\lambda_{TV} = 0.001$)	38.23 ($\lambda = 0.005$, $\lambda_{TV} = 0.001$)
		32.48 ($\lambda = 0.001$, $\lambda_{TV} = 0.001$)	34.36 ($\lambda = 0.005$, $\lambda_{TV} = 0.001$)	38.04 ($\lambda = 0.005$, $\lambda_{TV} = 0.001$)	38.90 ($\lambda = 0.005$, $\lambda_{TV} = 0.001$)
DC2	30	11.94 ($\lambda = 0.1$, $\lambda_{TV} = 0.005$)	11.66 ($\lambda = 0.1$, $\lambda_{TV} = 0.005$)	12.28 ($\lambda = 0.1$, $\lambda_{TV} = 0.005$)	12.26 ($\lambda = 0.1$, $\lambda_{TV} = 0.005$)
		23.46 ($\lambda = 0.5$, $\lambda_{TV} = 0.001$)	22.83 ($\lambda = 0.5$, $\lambda_{TV} = 0.001$)	24.27 ($\lambda = 0.5$, $\lambda_{TV} = 0.001$)	23.46 ($\lambda = 0.5$, $\lambda_{TV} = 0.001$)
	40	24.79 ($\lambda = 0.5$, $\lambda_{TV} = 0.001$)	27.18 ($\lambda = 0.5$, $\lambda_{TV} = 0.001$)	28.13 ($\lambda = 0.1$, $\lambda_{TV} = 0.001$)	27.70 ($\lambda = 0.1$, $\lambda_{TV} = 0.001$)
		25.47 ($\lambda = 0.5$, $\lambda_{TV} = 0.001$)	27.85 ($\lambda = 0.5$, $\lambda_{TV} = 0.001$)	29.05 ($\lambda = 0.1$, $\lambda_{TV} = 0.001$)	28.53 ($\lambda = 0.1$, $\lambda_{TV} = 0.001$)
	50	26.15 ($\lambda = 0.5$, $\lambda_{TV} = 0.001$)	28.53 ($\lambda = 0.5$, $\lambda_{TV} = 0.001$)	29.92 ($\lambda = 0.1$, $\lambda_{TV} = 0.001$)	29.40 ($\lambda = 0.1$, $\lambda_{TV} = 0.001$)
		26.83 ($\lambda = 0.5$, $\lambda_{TV} = 0.001$)	29.21 ($\lambda = 0.5$, $\lambda_{TV} = 0.001$)	30.79 ($\lambda = 0.1$, $\lambda_{TV} = 0.001$)	30.27 ($\lambda = 0.1$, $\lambda_{TV} = 0.001$)

Table 1 shows SRE(dB) values obtained by TSWCSU-TV with different values of α for DC1 and DC2. As shown in **Table 1**, we see that the SRE values obtained by TSWCSU-TV with $\alpha = \frac{1}{3}$ and $\frac{1}{4}$ are generally higher than those with $\alpha = 1$ for both DC1 and DC2 with examined noise levels. It shows the effectiveness of the two-step strategy for solving optimization problem (5). In addition, the difference between the highest SRE values with $\alpha = \frac{1}{4}$ and $\alpha = \frac{1}{3}$ are within 1dB for DC1 and DC2. To verify the effectiveness of the two-step strategy, **Fig. 3** shows the $\|G\mathbf{x}^{(t)} + \mathbf{b}\mathbf{v}^{(t)}\|_F$ of TSWCSU-TV for different values of α with the SNR = 30 dB on DC1 and DC2. According to **Fig. 3**, TSWCSU-TV with $\alpha = \frac{1}{3}$ has faster convergence speed than with other values. In this case, we set $\alpha = \frac{1}{3}$ to obtain a comparable performance in the following experiments.

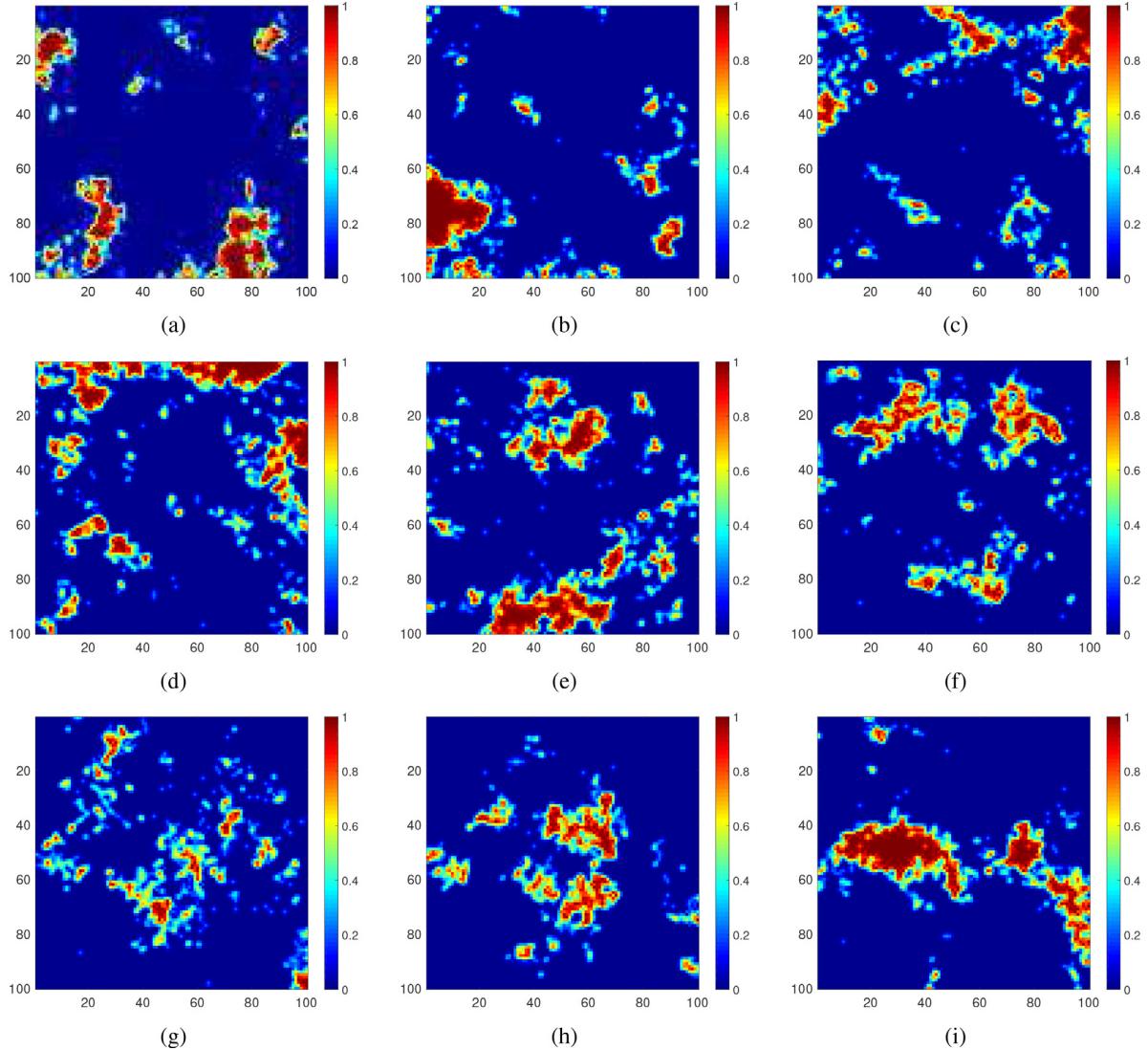


Fig. 2. True fractional abundances of nine endmembers for DC2. (a) Endmember #1; (b) Endmember #2; (c) Endmember #3; (d) Endmember #4; (e) Endmember #5; (f) Endmember #6; (g) Endmember #7; (h) Endmember #8; (i) Endmember #9.

Table 2

SRE(dB) values of TSWCSU-TV with different weighting coefficients on DC1 and DC2 (optimal parameters for which the reported values were achieved are indicated in the parentheses).

Algorithm	DC1			DC2		
	SNR = 30 dB	SNR = 40 dB	SNR = 50 dB	SNR = 30 dB	SNR = 40 dB	SNR = 50 dB
TSWCSU-TV with weighting W	21.41 ($\lambda = 0.5$, $\lambda_{TV} = 0.01$)	29.10 ($\lambda = 0.5$, $\lambda_{TV} = 0.001$)	35.96 ($\lambda = 0.005$, $\lambda_{TV} = 0.001$)	12.28 ($\lambda = 0.1$, $\lambda_{TV} = 0.005$)	24.27 ($\lambda = 0.5$, $\lambda_{TV} = 0.001$)	28.13 ($\lambda = 0.1$, $\lambda_{TV} = 0.001$)
TSWCSU-TV without weighting W	15.13 ($\lambda = 0.1$, $\lambda_{TV} = 0.005$)	23.04 ($\lambda = 0.1$, $\lambda_{TV} = 0.001$)	28.31 ($\lambda = 0.05$, $\lambda_{TV} = 0.001$)	11.59 ($\lambda = 0.05$, $\lambda_{TV} = 0.001$)	21.26 ($\lambda = 0.1$, $\lambda_{TV} = 0.001$)	25.75 ($\lambda = 0.1$, $\lambda_{TV} = 0.005$)

4.2.2. The weighting coefficients analysis

Here, we discuss the performance of TSWCSU-TV with different weighting coefficients. In Table 2, we show the SRE values of TSWCSU-TV with different weighting coefficients on DC1 and DC2. From Table 2, TSWCSU-TV with weighting obtains higher SRE values than TSWCSU-TV without weighting in all cases. It means that the reweighted strategy is helpful for improving the unmixing performance.

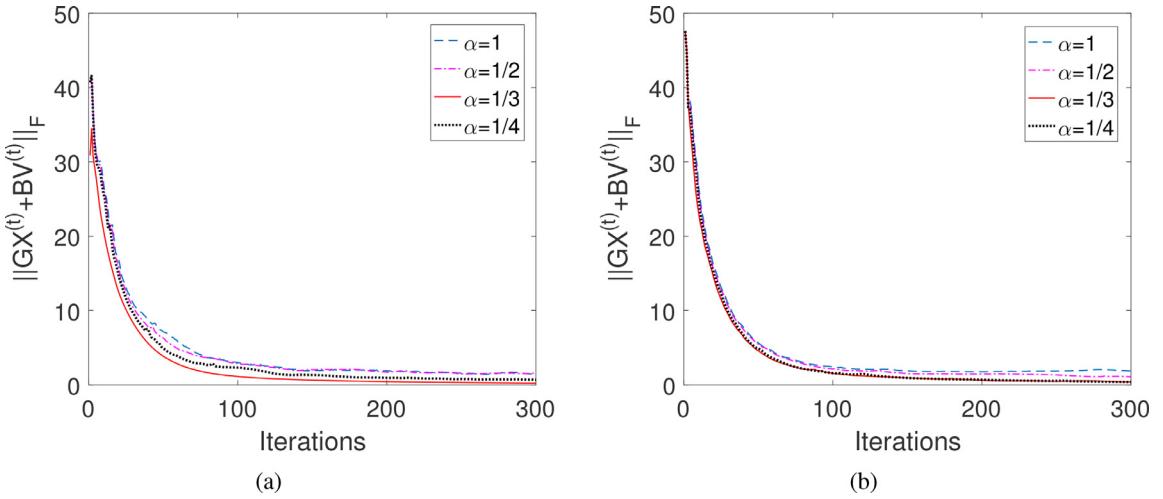


Fig. 3. Convergence curves of TSWCSU-TV for different values of α for (a) DC1 and (b) DC2 with SNR = 30 dB.

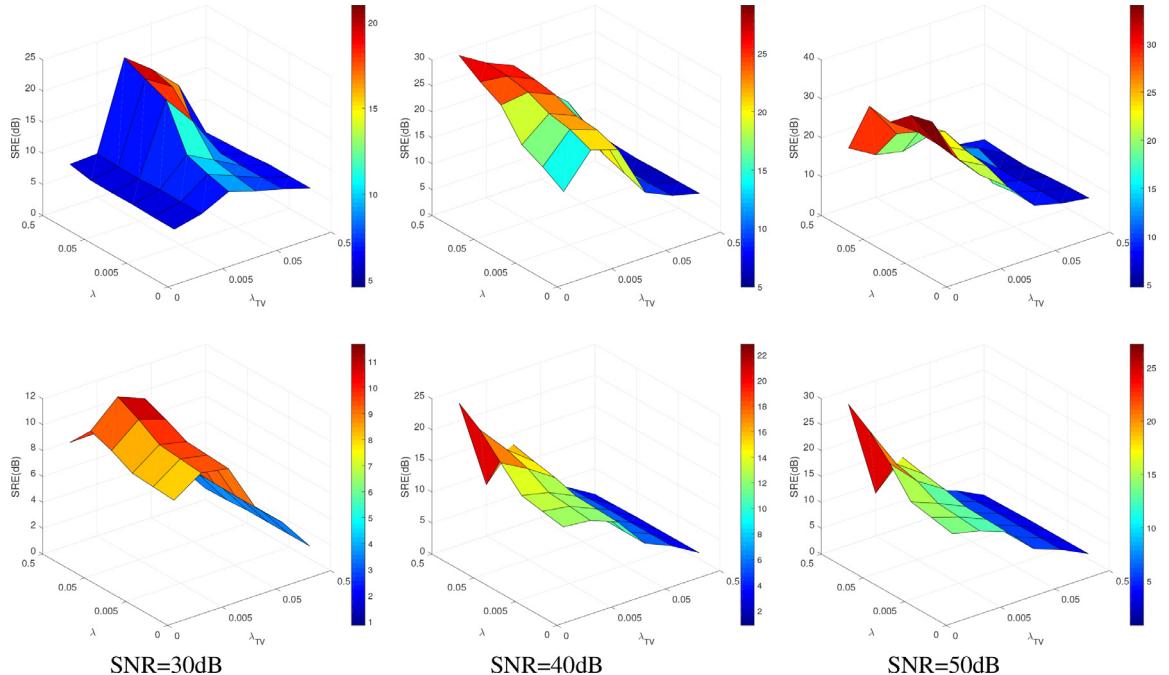


Fig. 4. SRE(dB) as a function of parameters λ and λ_{TV} for (Top row) DC1 and (Bottom row) DC2 with different SNRs.

4.2.3. Parameter settings

This experiment discusses the regularization parameter selection in TSWCSU-TV. Fig. 4 shows SRE(dB) as a function of parameters λ and λ_{TV} for DC1 and DC2 with different SNR levels. As the SNR gets lower, both the optimal λ and optimal λ_{TV} increase. From Fig. 4, we see that the best unmixing performances are often achieved with λ_{TV} less than 0.01 and λ greater than 0.1.

4.2.4. Comparison with other algorithms

We now compare the proposed unmixing algorithm with three state-of-the-art sparse unmixing algorithms for DC1 and DC2. Table 3 shows the SRE values obtained by SUNSAL, CLSUnSAL, SUNSAL-TV, and TSWCSU-TV under different SNRs. The optimal regularization parameters of each algorithm are also provided. According to Table 3, CLSUnSAL provides higher SRE values than SUNSAL for DC1 with three noise levels. For DC2, however, SUNSAL is better than CLSUnSAL. Clearly, SUNSAL-TV and TSWCSU-TV provide higher SRE values than both CLSUnSAL and SUNSAL, and the TSWCSU-TV algorithm obtains the best SREs for both DC1 and DC2.

Table 3

SRE(dB) values by different unmixing algorithms for DC1 and DC2 (optimal parameters for which the reported values were achieved are indicated in the parentheses).

Data cube	SNR(dB)	SUnSAL	CLSUnSAL	SUnSAL-TV	TSWCSU-TV
DC1	30	6.12 ($\lambda = 0.5$)	6.30 ($\lambda = 0.05$)	13.68 ($\lambda = 0.01, \lambda_{TV} = 0.01$)	21.41 ($\lambda = 0.5, \lambda_{TV} = 0.01$)
	40	11.04 ($\lambda = 0.01$)	15.79 ($\lambda = 0.5$)	21.44 ($\lambda = 0.01, \lambda_{TV} = 0.01$)	29.10 ($\lambda = 0.5, \lambda_{TV} = 0.001$)
	50	20.71 ($\lambda = 0.005$)	23.47 ($\lambda = 0.05$)	27.88 ($\lambda = 0.001, \lambda_{TV} = 0.001$)	35.96 ($\lambda = 0.005, \lambda_{TV} = 0.001$)
DC2	30	8.16 ($\lambda = 0.05$)	5.52 ($\lambda = 0.01$)	11.34 ($\lambda = 0.05, \lambda_{TV} = 0.001$)	12.28 ($\lambda = 0.1, \lambda_{TV} = 0.005$)
	40	13.59 ($\lambda = 0.01$)	10.65 ($\lambda = 0.005$)	20.18 ($\lambda = 0.005, \lambda_{TV} = 0.001$)	24.27 ($\lambda = 0.5, \lambda_{TV} = 0.001$)
	50	17.95 ($\lambda = 0.001$)	15.13 ($\lambda = 0.05$)	23.89 ($\lambda = 0.005, \lambda_{TV} = 0.001$)	28.13 ($\lambda = 0.1, \lambda_{TV} = 0.001$)

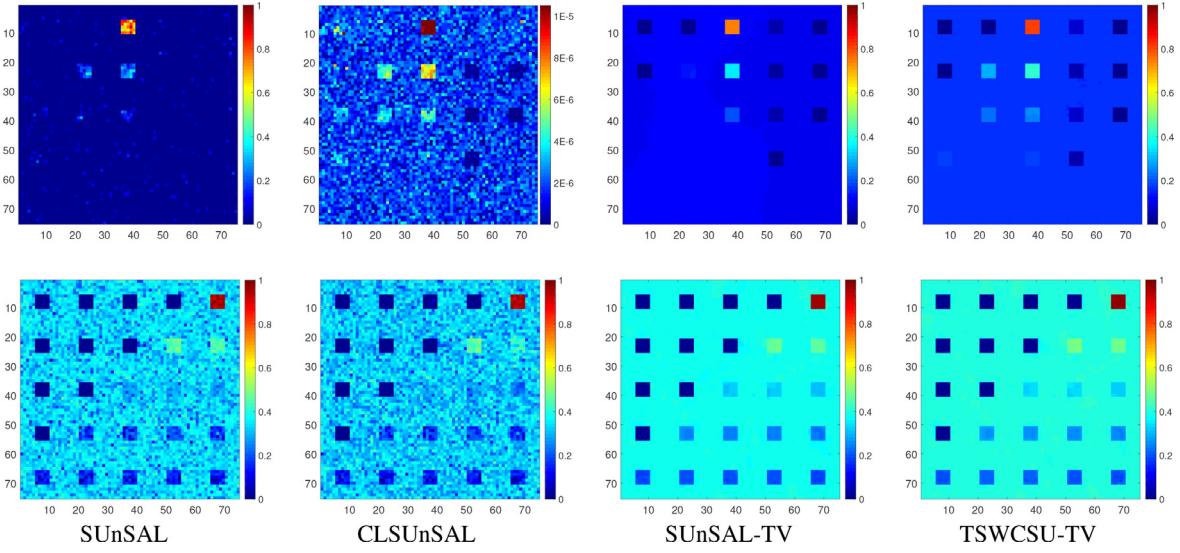


Fig. 5. Estimated abundance maps by different unmixing algorithms for (Top row) endmembers #3 and (Bottom row) #5 for DC1 with SNR = 30 dB.

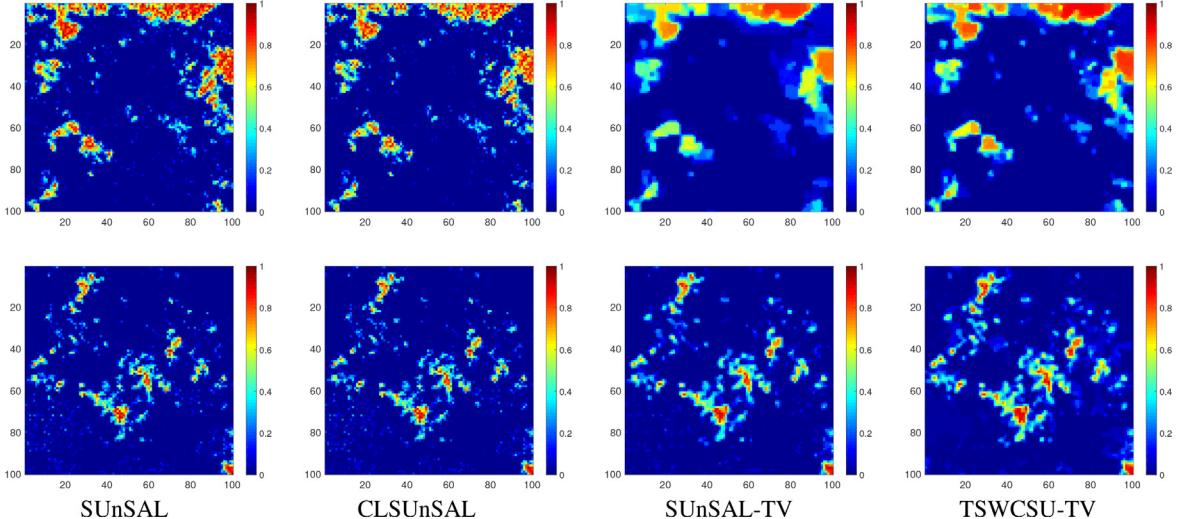


Fig. 6. Estimated abundance maps by different unmixing algorithms for (Top row) endmembers #4 and (Bottom row) #7 for DC2 with SNR = 30 dB.

To further verify the effectiveness of TSWCSU-TV, Figs. 5 and 6 show the estimated abundance maps by four comparing algorithms for endmembers #3 and #5 for DC1 and endmembers #4 and #7 for DC2 with SNR = 30 dB, respectively. From Fig. 5, we see that all four algorithms delineate most square regions in the abundance maps. The backgrounds of SUnSAL and CLSUnSAL, however, is full of noise. Owing to the TV term, both SUnSAL-TV and TSWCSU-TV provide much smoother background. Clearly, TSWCSU-TV provides better results for endmember #3 with more squares delineated. It is also clear

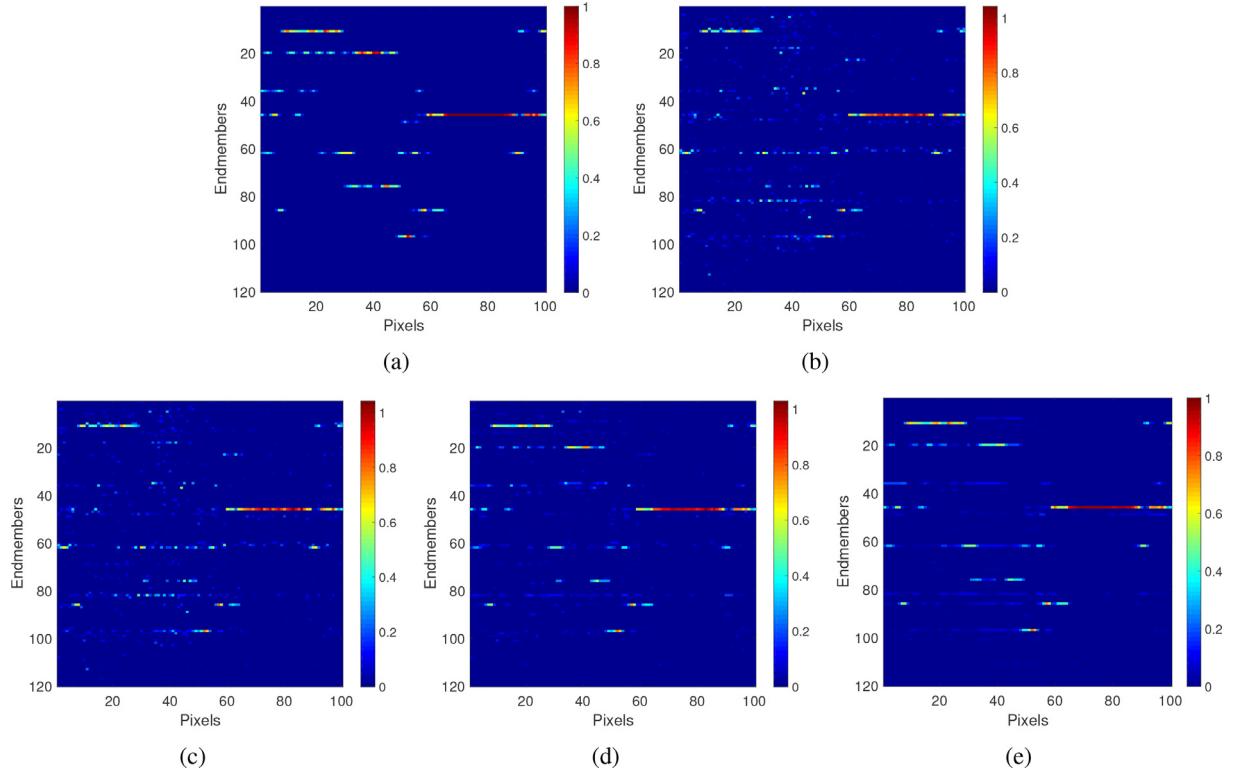


Fig. 7. True and estimated abundances by different algorithms for 100 selected pixels in DC2 with SNR of 30 dB. (a) True. (b) SUUnSAL. (c) CLSUUnSAL. (d) SUUnSAL-TV. (e) TSWCSU-TV.

from Fig. 6 that TSWCSU-TV gives not only the highest abundances for the respective endmembers but also the smoothest background.

Fig. 7 shows true and estimated abundances by different algorithms for 100 selected pixels in DC2 with SNR of 30 dB. The graphs for other pixels are similar, so we omit them here. From Fig. 7, although the lines for respective endmembers estimated by SUUnSAL and CLSUUnSAL are generally similar to the ground truth. There are still many low abundance values estimated for endmembers which are not actually present in the image. On the contrary, the estimated abundances of SUUnSAL-TV and TSWCSU-TV are more accurate and have a better visual effect. Clearly, TSWCSU-TV provides the most similar abundances to the ground truth than other three competing algorithms.

4.2.5. Convergence analysis

The goal of this experiment is to numerically analyze the convergence of SUUnSAL, CLSUUnSAL, SUUnSAL-TV and TSWCSU-TV. To this end, Fig. 8 plots the $\|\mathbf{GX}^{(t)} + \mathbf{BV}^{(t)}\|_F$ versus the iteration number for DC1 and DC2 with SNR of 30 dB. It is shown in Fig. 8 that the convergence curves tend to be stable when the number of iteration is less than 300. It is therefore sufficient to set the maximum iteration number to be 300. In addition, Fig. 8 also shows that TSWCSU-TV has faster convergence speed than SUUnSAL, CLSUUnSAL, and SUUnSAL-TV for both DC1 and DC2.

5. Real data experiment

For real data experiments, we use the well-known Airborne Visible Infrared Imaging Spectrometer (AVIRIS) imaging spectrometer to collect hyperspectral data from the Cuprite region of Nevada, USA, as shown in Fig. 9. The portion used in experiments corresponds to a 350×350 -pixel subset of the sector labeled as *f970619t01p02_r02_s03.a.rfl* in the online data². The spectrum contains 188 bands with a wavelength range of $0.4 \mu\text{m} - 2.5 \mu\text{m}$ and a spectral resolution of 10 nm. In the experiments, the USGS mineral spectrum library $\mathbf{A} \in \mathbb{R}^{188 \times 240}$ was used. Here, the low SNR and the absorption phase of the water vapor are eliminated, and finally 188 bands are left for subsequent processing. The Tetracorder 4.4 software³ was used to inspect the different minerals in the mine to obtain the distribution of each mineral. Here, the maximum number

² <http://aviris.jpl.nasa.gov/html/aviris.freedata.html>

³ <http://speclab.cr.usgs.gov/PAPER/tetracorder>

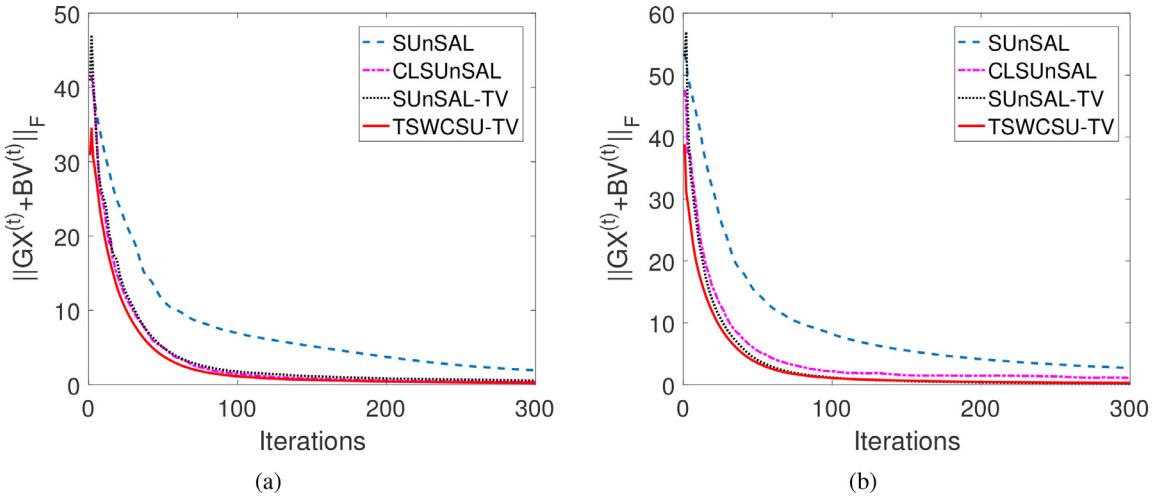


Fig. 8. Convergence curves of different unmixing algorithms for (a) DC1 and (b) DC2 with SNR = 30 dB.

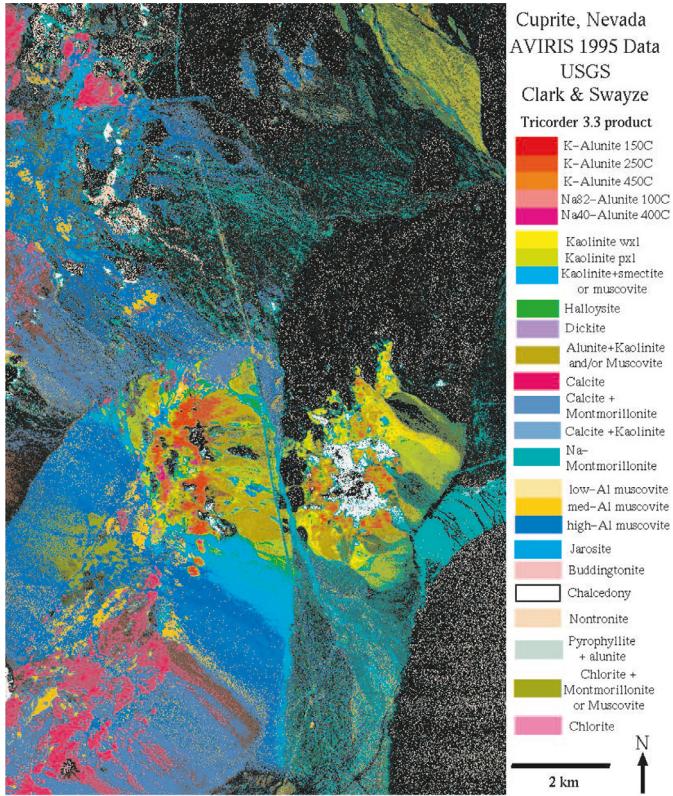


Fig. 9. USGS map showing the location of different minerals in the Cuprite mining district in Nevada. The map is available online at http://speclab.cr.usgs.gov/cuprite95.tif.2.2um_map.gif.

of iterations is set to 300. Similarly, as in [12], we set the regularization parameters $\lambda = \lambda_{TV} = 0.001$ for SUnSAL, CLSUnSAL, SUnSAL-TV, and TSWCSU-TV.

Fig. 10 shows a qualitative comparison among the fractional abundance maps of the three considered minerals (i.e., Alunite, Chalcedony and Muscovite) in the hyperspectral scene. Generally, the highest abundances estimated by the sparse unmixing algorithms correspond with those pixels belonging to the respective class of minerals. It is clear from Fig. 10 that the estimations of the four algorithms are similar to those of the Tetracorder software product, indicating the effectiveness of the sparse unmixing algorithms. It is also worth noting that the abundances estimated by TSWCSU-TV are generally comparable or higher in the regions classified as respective minerals in comparison with SUnSAL, CLSUnSAL, and SUnSAL-

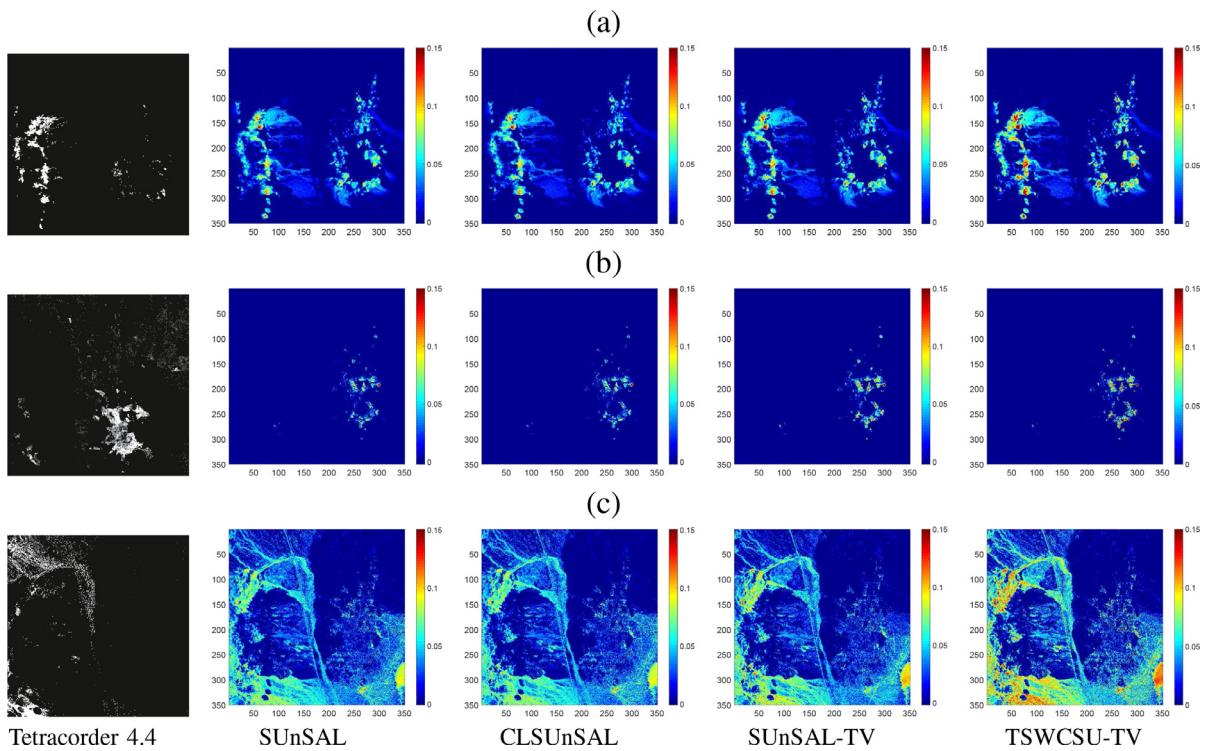


Fig. 10. Estimated abundance maps by Tetracorder 4.4 and different unmixing algorithms for the AVIRIS Cuprite scene for three minerals: (a) Alunite; (b) Chalcedony; (c) Muscovite.

TV. This observation is more obvious for the abundances corresponding to Alunite and Muscovite than the abundance of Chalcedony.

Therefore, we qualitatively conclude that TSWCSU-TV offers promising results for the unmixing real hyperspectral data.

6. Conclusion

In this paper, we have proposed a weighted collaborative sparse unmixing via TV model for hyperspectral unmixing. To solve the proposed model, we design a two-step strategy algorithm based on ADMM. Its key idea is to compute the current solution by a linear combination of the results of two previous iterates, instead of only using current solution in classic ADMM. Experimental results have demonstrated the effectiveness of the proposed WCSU-TV model and the two-step iterative strategy for the hyperspectral unmixing problem. Note that the TV regularizer may lead to oversmooth abundance maps in some cases. In the future, instead of the TV regularization, we will study a reweighted spatial regularizer to exploit the spatial information for hyperspectral unmixing.

Supplementary material

Supplementary material associated with this article can be found, in the online version, at [10.1016/j.amc.2021.126059](https://doi.org/10.1016/j.amc.2021.126059)

CRediT authorship contribution statement

Jin-Ju Wang: Methodology, Software, Writing - original draft. **Ting-Zhu Huang:** Visualization, Resources, Supervision. **Jie Huang:** Conceptualization, Writing - review & editing, Visualization. **Liang-Jian Deng:** Writing - review & editing, Visualization.

References

- [1] D. Landgrebe, Hyperspectral image data analysis, *IEEE Signal Process. Mag.* 19 (1) (2002) 17–28.
- [2] C.I. Chang, *Hyperspectral Imaging: Techniques for Spectral Detection and Classification*, Plenum Publishing Co., 2003.
- [3] M.E. Winter, N-FINDR: An algorithm for fast autonomous spectral endmember determination in hyperspectral data, *Proc. SPIE Int. Soc. Opt. Eng.* 3753 (1999) 266–275.
- [4] N. Keshava, J.F. Mustard, Spectral unmixing, *IEEE Signal Process. Mag.* 19 (1) (2002) 44–57.

- [5] A. Plaza, P. Martinez, R. Perez, J. Plaza, A quantitative and comparative analysis of endmember extraction algorithms from hyperspectral data, *IEEE Trans. Geosci. Remote Sens.* 42 (3) (2004) 650–663.
- [6] J.M. Bioucas-Dias, A. Plaza, N. Dobigeon, M. Parente, Q. Du, P. Gader, J. Chanussot, Hyperspectral unmixing overview: geometrical, statistical, and sparse regression-based approaches, *IEEE J. Select. Top. Appl. Earth Observat. Remote Sens.* 5 (2) (2012) 354–379.
- [7] J.M. Bioucas-Dias, M.A.T. Figueiredo, Alternating direction algorithms for constrained sparse regression: application to hyperspectral unmixing, in: *Proceedings of the Workshop on Hyperspectral Image Signal Processing: Evolution in Remote Sensing*, 2010, pp. 1–4.
- [8] M. Iordache, J.M. Bioucas-Dias, A. Plaza, Sparse unmixing of hyperspectral data, *IEEE Trans. Geosci. Remote Sens.* 49 (6) (2011) 2014–2039.
- [9] M. Iordache, J.M. Bioucas-Dias, A. Plaza, Collaborative sparse regression for hyperspectral unmixing, *IEEE Trans. Geosci. Remote Sens.* 52 (1) (2013) 341–354.
- [10] J. Sigurdsson, M. Ulfarsson, J.R. Sveinsson, Hyperspectral unmixing with ℓ_q regularization, *IEEE Trans. Geosci. Remote Sens.* 52 (11) (2014) 6793–6806.
- [11] C.Y. Zheng, H. Li, Q. Wang, C.L.P. Chen, Reweighted sparse regression for hyperspectral unmixing, *IEEE Trans. Geosci. Remote Sens.* 54 (1) (2015) 479–488.
- [12] M. Iordache, J.M. Bioucas-Dias, A. Plaza, Total variation spatial regularization for sparse hyperspectral unmixing, *IEEE Trans. Geosci. Remote Sens.* 50 (11) (2012) 4484–4502.
- [13] Z.B. Wu, Z.H. Wei, L. Sun, J.J. Liu, Hyperspectral unmixing based on iterative weighted ℓ_1 regularization, *J. Nanj. Univ. Sci. Technol.* 35 (4) (2011) 431–435.
- [14] X.L. Zhao, F. Wang, T.Z. Huang, M.K. Ng, R.J. Plemons, Deblurring and sparse unmixing for hyperspectral images, *IEEE Trans. Geosci. Remote Sens.* 51 (7) (2013) 4045–4058.
- [15] Y. Zhong, R. Feng, L. Zhang, Non-local sparse unmixing for hyperspectral remote sensing imagery, *IEEE J. Select. Top. Appl. Earth Observat. Remote Sens.* 7 (6) (2014) 1889–1909.
- [16] W. Tang, Z.W. Shi, Y. Wu, C.S. Zhang, Sparse unmixing of hyperspectral data using spectral a priori information, *IEEE Trans. Geosci. Remote Sens.* 53 (2) (2015) 770–783.
- [17] P.V. Giampouras, K.E. Themelis, A.A. Rontogiannis, K.D. Kourtoumbas, Simultaneously sparse and low-rank abundance matrix estimation for hyperspectral image unmixing, *IEEE Trans. Geosci. Remote Sens.* 54 (8) (2016) 4775–4789.
- [18] M. Rizkinia, M. Okuda, Joint local abundance sparse unmixing for hyperspectral images, *Remote Sens. (Basel)* 9 (12) (2017) 1224.
- [19] Q. Qu, N.M. Nasrabadi, T.D. Tran, Abundance estimation for bilinear mixture models via joint sparse and low-rank representation, *IEEE Trans. Geosci. Remote Sens.* 52 (7) (2014) 4404–4423.
- [20] Y. Altmann, N. Dobigeon, J. Tourneret, Bilinear models for nonlinear unmixing of hyperspectral images, in: *Proceedings of the 3rd Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing (WHISPERS)*, 2011, pp. 1–4.
- [21] P. Gader, D. Dranishnikov, A. Zare, J. Chanussot, A sparsity promoting bilinear unmixing model, in: *Proceedings of the 4th Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing (WHISPERS)*, 2012, pp. 1–4.
- [22] C. Li, Y. Ma, J. Huang, X.G. Mei, C.Y. Liu, J.Y. Ma, GBM-based unmixing of hyperspectral data using bound projected optimal gradient method, *IEEE Geosci. Remote Sens. Lett.* 13 (7) (2016) 952–956.
- [23] D.C. Heinz, C.I. Chang, Fully constrained least squares linear spectral mixture analysis method for material quantification in hyperspectral imagery, *IEEE Trans. Geosci. Remote Sens.* 39 (2001) 529–545.
- [24] J. Huang, T.Z. Huang, L.J. Deng, X.L. Zhao, Joint-sparse-blocks and low-rank representation for hyperspectral unmixing, *IEEE Trans. Geosci. Remote Sens.* 57 (4) (2018) 2419–2438.
- [25] Z. Zhang, Y. Xu, J. Yang, X. Li, D. Zhang, A survey of sparse representation: algorithms and applications, *IEEE Access* 3 (2015) 490–530.
- [26] F. Chen, Y. Zhang, Sparse hyperspectral unmixing based on constrained $\ell_p - \ell_2$ optimization, *IEEE Geosci. Remote Sens. Lett.* 10 (5) (2013) 1142–1146.
- [27] E.J. Candès, T. Tao, Decoding by linear programming, *IEEE Trans. Inf. Theory* 51 (12) (2005) 4203–4215.
- [28] E.J. Candès, T. Tao, Near-optimal signal recovery from random projections: universal encoding strategies? *IEEE Trans. Inf. Theory* 52 (12) (2006) 5406–5425.
- [29] M.D. Iordache, A Sparse Regression Approach to Hyperspectral Unmixing, Universidade Techica de Lisboa Instituto Superior Techico, 2011 Ph.D. thesis.
- [30] M. Nikolova, M. Ng, S.Q. Zhang, W.K. Ching, Efficient reconstruction of piecewise constant images using nonsmooth nonconvex minimization, *SIAM J. Imaging Sci.* 1 (2008) 2–25.
- [31] M.B.W. E. J. Candès, S. Boyd, Enhancing sparsity by reweighted ℓ_1 minimization.
- [32] S.Q. Zhang, J. Li, H.C. Li, C.D. Deng, A. Plaza, Spectral-spatial weighted sparse regression for hyperspectral image unmixing, *IEEE Trans. Geosci. Remote Sens.* 56 (6) (2018) 3265–3276.
- [33] D. Gabay, B. Mercier, A dual algorithm for the solution of nonlinear variational problems via finite element approximation, *Comput. Math. Appl.* 2 (1) (1976) 17–40.
- [34] R.H. Chan, M. Tao, X.M. Yuan, Constrained total variational deblurring models and fast algorithms based on alternating direction method of multipliers, *SIAM J. Imaging Sci.* (Sci. 6) (2013) 680–697.
- [35] J.F. Yang, Y. Zhang, Alternating direction algorithms for ℓ_1 -problems in compressive sensing, *SIAM J. Scient. Comput.* 33 (1) (2011) 250–278.
- [36] J. Yang, Y. Zhang, W. Yin, A fast alternating direction method for tv $\ell_1 - \ell_2$ signal reconstruction from partial fourier data, *IEEE J. Sel. Top. Signal Process.* 4 (2) (2010) 288–297.
- [37] T.X. Jiang, T.Z. Huang, X.L. Zhao, L.J. Deng, Y. Wang, Fastderain: a novel video rain streak removal method using directional gradient priors, *IEEE Trans. Image Process.* 28 (4) (2018) 2089–2102.
- [38] Y. Chen, T.Z. Huang, X.L. Zhao, Destriping of multispectral remote sensing image using low-rank tensor decomposition, *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* 11 (12) (2018) 4950–4967.
- [39] A. Beck, M. Teboulle, A fast iterative shrinkage-thresholding algorithm for linear inverse problems, *SIAM J. Imaging Sci.* 2 (1) (2009) 183–202.
- [40] J.M. Bioucas-Dias, M.A.T. Figueiredo, A new twist: two-step iterative shrinkage/thresholding algorithms for image restoration, *IEEE Trans. Image Process.* 16 (12) (2007) 2992–3004.
- [41] J.M. Bioucas-Dias, M.A.T. Figueiredo, Two-step algorithms for linear inverse problems with non-quadratic regularization, in: *Proceedings of the IEEE International Conference on Image Processing*, 1, 2007.
- [42] P.L. Combettes, Signal recovery by proximal forward-backward splitting, *Multiscale Model. Simul.* 4 (4) (2006) 1168–1200.
- [43] S.J. Wright, R.D. Nowak, M.A.T. Figueiredo, Sparse reconstruction by separable approximation, *IEEE Trans. Signal Process.* 57 (7) (2008) 2479–2493.
- [44] J.F. Cai, S. Osher, Z. Shen, Convergence of the linearized bregman iteration for ℓ_p -norm minimization, *Math. Comput.* 78 (268) (2009) 2127–2136.
- [45] W. Yin, D.G.S. Osher, J. Darbon, Bregman iterative algorithms for ℓ_1 -minimization with applications to compressed sensing, *SIAM J. Imaging Sci.* 1 (1) (2008) 143–168.
- [46] K. Abercromby, Communication of the NASA JSC spacecraft materials spectral database, 2006.