applied optics

Graph-regularized tensor robust principal component analysis for hyperspectral image denoising

Yongming Nie,[1,2] Linsen Chen,[1] Hao Zhu,[1] Sidan Du,[1,3] Tao Yue,[1,4] and Xun Cao[1]

[1]School of Electronic Science and Engineering, Nanjing University, Nanjing 210023, China
[2]e-mail: nju323nym@gmail.com
[3]e-mail: coff128@nju.edu.cn
[4]e-mail: yuetao@nju.edu.cn

Received 26 April 2017; revised 29 June 2017; accepted 29 June 2017; posted 30 June 2017 (Doc. ID 294661); published 24 July 2017

In this paper, we have developed a novel model that is named graph-regularized tensor robust principal component analysis (GTRPCA) for denoising hyperspectral images (HSIs). Incorporating spectral graph regularization into TRPCA makes the model more accurate by preserving local geometric structures embedded in a high-dimensional space. Based on tensor singular value decomposition (t-SVD), we introduce a general tensor-based altering direction method of multipliers (ADMM) algorithm which can solve the proposed model for denoising HSIs. Experiments on both the synthetic and real captured datasets have demonstrated the effectiveness of the proposed method. © 2017 Optical Society of America

OCIS codes: (100.4145) Motion, hyperspectral image processing; (150.1135) Algorithms; (110.4280) Noise in imaging systems.

https://doi.org/10.1364/AO.56.006094

1. INTRODUCTION

Hyperspectral images (HSIs) play an important role in many fields, such as precision agriculture, remote sensing, surveillance, etc. However, the captured hyperspectral images may be corrupted by different noise, which severely degrades the quality of the HSI and brings difficulties in data classification and extraction of useful information. HSI denoising has drawn increasing attention in recent years [1–6].

There are mainly two kinds of state-of-the-art methods for HSI denoising: the *2D extended approach* [7–9] and the *tensor-based approach* [3,10,11]. The 2D extended approach regards each band of the HSI as a 2D image. The existing 2D image denoising methods can be applied band by band directly, e.g., block-matching and 3-D filtering (BM3D) method [7]. This extension, however, neglects the intrinsic properties of HSIs and generally cannot attain good performance in practical applications. Another reasonable extension is the patch-based image denoising methods, which take the small local patches of the image into consideration. By building 3D cubes of the HSI instead of 2D patches of a traditional image, the corresponding 3D-cube-based HSI denoising algorithm can then be constructed, e.g., BM4D [8]. The deficiency of these methods is that they neglect the useful global correlation along spectrum (GCS) [11] underlying the HSI. To overcome this problem, a tensor-based approach implements the HSI denoising by applying tensor factorization techniques to the HSI third-order tensor (i.e., two spatial and one spectral dimensions). Liu *et al.* [3] designed the parallel factor analysis (PARAFAC) method by utilizing parallel factor analysis. Renard *et al.* [10] presented a low-rank tensor approximation method by employing the Tucker decomposition. Tensor dictionary learning (TDL) [11] took both GCS and the nonlocal self-similarity across space (NSS) [11] into account for denoising. Existing tensor-based approaches unfold tensors into matrices and then solve the problem by manipulations on these unfolded matrices [12,13]. However, such matricization fails to exploit the intrinsic HSI data structure (it means that the HSI data is global correlation along spectrum and nonlocal self-similarity across space) and often leads to suboptimal procedure.

Motivated by the success of low-rank presentation and manifold learning in image processing [14], we develop a novel tensor robust principal component analysis (TRPCA) model for the task of HSI denoising by incorporating graph regularization to alleviate this issue. Both GCS and NSS of HSI data are considered by incorporating the graph regularization into TRPCA [15]. On the one hand, owing to the high spectral correlation among the ground-truth subimages in different bands [14,16], the useful information can be assumed to lie in several low-dimensional linear subspaces, i.e., the tensor of HSI data is low rank. On the other hand, a nearest neighbor graph [17–19] is built to model and enforce to preserve the local structural information of the HSI data during the denoising process. In contrast, most existing denoising methods seldom consider

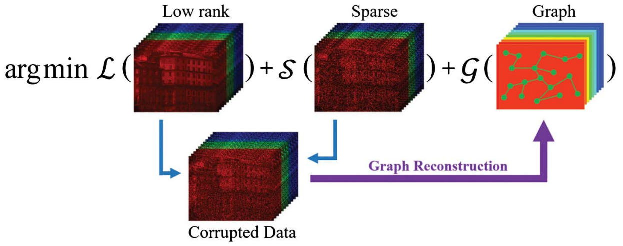

Low rank
Sparse
Graph
argmin $\mathcal{L}$( ) + $\mathcal{S}$( ) + $\mathcal{G}$( )
Graph Reconstruction
Corrupted Data

Fig. 1. Illustration of the proposed GTRPCA for hyperspectral images denoising.

this local geometrical structure, which is also very important to preserve the useful information. Graph regularization methods are proposed to model the high-dimensional data samples in a more accurate way and preserve local geometric structures embedded in a high-dimensional space. So, we propose a novel framework called graph-based tensor robust principal analysis (GTRPCA) for HSI denoising. Figure 1 is an illustration of GTRPCA. Our proposed model can be formulated as follows:

$$\min_{\mathcal{L},\mathcal{E}} \|\mathcal{L}\|_* + \lambda\|\mathcal{S}\|_1 + \gamma\,\mathrm{Re}(\mathcal{G}) \qquad \text{s.t.}\,\mathcal{X} = \mathcal{L} + \mathcal{S}, \tag{1}$$

where the sparse outliers in the data are modeled by $\mathcal{S}$, and $\mathcal{L}$ is the low-rank tensor approximation of $\mathcal{X}$ (i.e., HSI data). Parameters λ and γ control the amount of sparsity of $\mathcal{S}$ and smoothness of $\mathcal{L}$ on the graph $\mathcal{G}$, respectively. We will specifically introduce our model in Section 3.

The main contributions of this paper are summarized as follows:

- We developed a novel graph-based tensor RPCA model, namely GTRPCA, to effectively recover the noisy HSI. The model not only exploits the high spectral correlation between the observations at different spectral bands, but also preserves the intrinsic local structure of the original hyperspectral data.
- An altering direction method of multipliers (ADMM)-based algorithm is designed to effectively solve the HSI denoising optimization problem under tensor singular value decomposition (t-SVD)-based low-rank tensor constraint.
- We conduct our method on the extensive evaluation of several challenge datasets, where an obvious improvement over the state-of-the-art approaches is achieved.

The rest of this paper is organized as follows. Section 2 provides an overview of t-SVD. GTRPCA and a corresponding convex optimization algorithm are given in Section 3. In Section 4 we demonstrate the experiments of denoising hyperspectral images based on our model. Finally, the conclusion is provided in Section 5.

2. PRELIMINARIES

In this section, we introduce the tensor notations used throughout the paper and the preliminaries as proposed in [20–22].

First, we introduce the notations and give the basic definitions used throughout the paper. We denote tensors by bold upper italic letters, e.g., $\mathcal{X}$. Matrices are denoted by bold upper case letters, e.g., $\mathbf{X}$. Vectors are denoted by lower case letters, e.g., x. For a third-order tensor $\mathcal{X} \in \mathbb{R}^{n_1\times n_2\times n_3}$, we denote its (i,j,k)-th element as $\mathcal{X}_{ijk}$ or x_{ijk} and use the notation $\mathcal{X}(i,:,:)$, $\mathcal{X}(:,i,:)$, $\mathcal{X}(:,:,i)$ to denote the i-th horizontal, lateral, and frontal slice, respectively. Generally, the frontal slice $\mathcal{X}(:,:,i)$ is denoted compactly as $\mathcal{X}^{(i)}$. $\mathcal{X}(:,i,j)$, $\mathcal{X}(i,:,j)$, and $\mathcal{X}(i,j,:)$ denote the model-1, model-2, and model-3 **tubal** (tubal is a 1D section defined by fixing all indices but one [23]). $\mathcal{X}_f = fft(\mathcal{X},[],3)$ denotes the Fourier transform along the third dimension; in the same fashion, we can also compute $\mathcal{X}$ from $\mathcal{X}_f$ via $ifft(\mathcal{X}_f,[],3)$ using the inverse fast Fourier transformation (FFT).

Before introducing the t-SVD and its derived tensor unclear norm (as defined by Definition 4), the new tensor-tensor product is defined based on an important concept, *block circular matrices* [22] ($bcirc(\mathcal{X})$), which can be regarded as a new matricization of a tensor. Reference [22] also defines the block-based operators, the block vectorizing $bvec(\mathcal{X})$ and its opposite operation $bvfold(bvec(\mathcal{X})) = \mathcal{X}$ and the block diagonal matrix and its opposite operation $bdfold(bdiag(\mathcal{X})) = \mathcal{X}$:

$$bcirc(\mathcal{X}) := \begin{pmatrix} \mathcal{X}^{(1)} & \mathcal{X}^{(n_3)} & \cdots & \mathcal{X}^{(2)} \\ \mathcal{X}^{(2)} & \mathcal{X} & \cdots & \mathcal{X}^{(3)} \\ \vdots & \vdots & \vdots & \vdots \\ \mathcal{X}^{(n_3)} & \mathcal{X}^{(n_3-1)} & \cdots & \mathcal{X}^{(1)} \end{pmatrix}. \tag{2}$$

We also define the following block-based operators, the block vectorizing and its opposite operation:

$$bvec(\mathcal{X}) := \begin{pmatrix} \mathcal{X}^{(1)} \\ \mathcal{X}^{(2)} \\ \vdots \\ \mathcal{X}^{(n_3)} \end{pmatrix}, bvfold(bvec(\mathcal{X})) = \mathcal{X}, \tag{3}$$

and the block diagonal matrix and its opposite operation:

$$\mathrm{bdiag}(\mathcal{X}) := \begin{pmatrix} \mathcal{X}^{(1)} & & \\ & \ddots & \\ & & \mathcal{X}^{(n_3)} \end{pmatrix}, bdfold(bdiag(\mathcal{X})) = \mathcal{X}. \tag{4}$$

A. Tensor Singular Value Decomposition t-SVD

The t-product between two third-order tensors can be defined as follows [22]:

Definition 1 (t-product): *Let $\mathcal{A} \in \mathbb{R}^{n_1\times n_2\times n_3}$ and $\mathcal{B} \in \mathbb{R}^{n_1\times n_4\times n_3}$; the t-product $\mathcal{C} = \mathcal{A} * \mathcal{B}$ is a tensor of size $n_1 \times n_4 \times n_3$, where the $(i,j)_{\mathrm{th}}$ tubal is denoted by $\mathcal{C}(i,j,:)$ for $i = 1,2,\ldots,n_1$ and $j = 1,2,\ldots,n_4$ of the tensor $\mathcal{C}$ is given by $\sum_{k=1}^{n_2}\mathcal{A}(i,k,:) * \mathcal{B}(k,j,:)$.*

Note that the t-product reduces to the standard matrix by matrix product when $n_3 = 1$, and is analogous to the matrix by matrix product except that the circular convolution replaces the product operation between the elements, which are now tubal elements. The t-product is given as follows:

$$\mathcal{C} = \mathcal{A} * \mathcal{B} := bvfold(bcirc(\mathcal{A})bvec(\mathcal{B})). \tag{5}$$

The t-SVD of $\mathcal{X}$ is given by Theorem 2.

Theorem 2 (t-SVD): *For $\mathcal{X} \in \mathbb{R}^{n_1\times n_2\times n_3}$, the t-SVD of $\mathcal{X}$ is given by*

$$\mathcal{X} = \mathcal{U} * \mathcal{S} * \mathcal{V}^T,$$

where $\mathcal{U}$ and $\mathcal{V}$ are orthogonal tensor of size $n_1 \times n_1 \times n_3$ and $n_2 \times n_2 \times n_3$, respectively. $\mathcal{S}$ is a rectangular f-diagonal tensor of size $n_1 \times n_2 \times n_3$, and $*$ denotes the t-product.

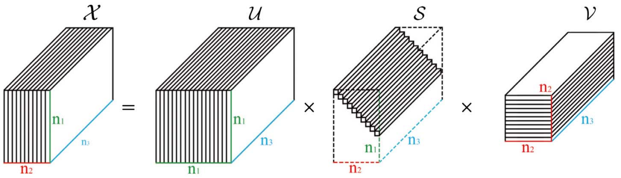


Fig. 2. t-SVD of an $n_1 \times n_2 \times n_3$ tensor.

We can obtain the t-SVD by computing matrix SVDs in the Fourier domain, see Algorithm 1. Figure 2 illustrates the t-SVD factorization. The t-SVD allows the tensor $\mathcal{X}$ to be written as a finite sum of outer product of matrices [21]:

$$\mathcal{X} = \sum_{i=1}^{\min(n1,n2)} \mathcal{U}(:,i,:) * \mathcal{S}(i,i,:) * \mathcal{V}(:,i,:)^T. \tag{6}$$

Note that the t-SVD can be efficiently computed based on the matrix SVD in the Fourier domain. This is based on a key property that the block circular matrix can be mapped to a block diagonal matrix in the Fourier domain [21], i.e.,

$$\begin{pmatrix} \mathcal{X}_f^{(1)} & & \\ & \ddots & \\ & & \mathcal{X}_f^{(n_3)} \end{pmatrix} = \begin{pmatrix} \mathcal{U}_f^{(1)} & & \\ & \ddots & \\ & & \mathcal{U}_f^{(n_3)} \end{pmatrix} \cdot \begin{pmatrix} \mathcal{S}_f^{(1)} & & \\ & \ddots & \\ & & \mathcal{S}_f^{(n_3)} \end{pmatrix} \cdot \begin{pmatrix} \mathcal{V}_f^{(1)} & & \\ & \ddots & \\ & & \mathcal{V}_f^{(n_3)} \end{pmatrix},$$

where $\cdot$ denotes common matrix product, and we have n_3 blocks matrix SVD: $\mathcal{X}_f^{(i)} = \mathcal{U}_f^{(i)} \mathcal{S}_f^{(i)} (\mathcal{V}_f^{(i)})^T$, $i = 1, \ldots, n_3$.

Algorithm 1: *t-SVD*

```
Procedure T-SVD (Data tensor $\mathcal{X} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$)
    $\mathcal{X}_f = fft(\mathcal{X}, [], 3)$
    for $k = 1{:}n_3$ do
        $[U, \Sigma, V] = SVD(\mathcal{X}_f^{(k)})$;
        $\mathcal{U}_f^{(k)} = U$, $\mathcal{S}_f^{(k)} = \Sigma$, $\mathcal{V}_f^{(k)} = V$;
        $\mathcal{U} = ifft(\mathcal{U}_f, [], 3)$;
        $\mathcal{S} = ifft(\mathcal{S}_f, [], 3)$;
        $\mathcal{V} = ifft(\mathcal{V}_f, [], 3)$.
    return $\mathcal{U}, \mathcal{S}, \mathcal{V}$.
```

B. Tensor Nuclear Norm via t-SVD

Now, we introduce the tensor tubal rank and nuclear norm [15,22,24,25].

Definition 3 (tensor tubal rank) [15]: *The tensor tubal rank, denoted as $rank_t(\mathcal{X})$, is defined as the number of nonzero tubal of $\mathcal{S}$, where $\mathcal{S}$ is from the $t-SVD$ of $\mathcal{X} = \mathcal{U} * \mathcal{S} * \mathcal{V}^T$. That is,*

$$rank_t(\mathcal{X}) = \sharp\{i{:}\mathcal{S}(i,i,:) \neq 0\}.$$

Definition 4 (tensor nuclear norm) [25]: *The tensor nuclear norm of a tensor $\mathcal{X} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$ (denoted as $\|\mathcal{X}\|_*$) is defined as the average of the nuclear norm of all the frontal slices of $\mathcal{X}_f$, i.e., $\|\mathcal{X}\|_* := \frac{1}{n_3}\sum_{i=1}^{n_3} \|\mathcal{X}_f^{(i)}\|_*$.*

The tensor nuclear norm, which is defined in the Fourier domain, is the tightest convex relaxation of the tensor tubal rank. As following [25], $\|\mathcal{X}\|_* := \frac{1}{n_3}\|bcirc(\mathcal{X})\|_*$, the tensor nuclear norm equals the nuclear norm of the block circular matrix in the original domain. The tensor nuclear norm is important for our model and solving the optimization equation. Unlike the current approaches that lost optimality in the representation to handle multi-linear data extend the 2D SVD-based vector space approach to the high-order ($N > 2$) case, our approach is based upon recent results on decomposition of tensor [22]. In the next section we will undertake a novel study for HSI denoising (tensor data) using a graph regularization tensor robust principal component analysis model based on the tensor nuclear norm.

3. GRAPH-REGULARIZED TENSOR ROBUST PRINCIPAL COMPONENT ANALYSIS

In this section, the graph-regularized tensor robust principal component analysis method is presented. The method takes the intrinsic geometric structure of the subimages in different bands into consideration. It can be achieved by incorporating manifold learning into the TRPCA [15,25] framework, which is a natural extension of RPCA [26].

A. Model

For the case of noisy HSI data $\mathcal{X}$, we seek a robust $\mathcal{L}$ on graph $\mathcal{G}$. On the one hand, to take full advantage of the high spectral correlation between the observation images in distinct bands [24], the data tensor $\mathcal{X}$ is enforced to be low rank; on the other hand, to preserve the intrinsic local structure of the original HSI data, the graph regularizer is incorporated in the objective function. At first, a nearest neighbor graph $\mathcal{G}^{(i)}$ can be constructed by viewing each x_i as its vertex. The weight matrix of $\mathcal{G}^{(k)}$ is denoted by W. If x_i is one of the k-nearest neighbors of x_j, the weight is assigned as $W_{ij} = \exp(-\|x_i - x_j\|_2^2/\sigma^2)$, which is introduced in [17]. To preserve the local geometrical structure in coefficient space, one may naturally hope that, if two data points x_i and x_j are close in the graph, their corresponding representations l_i and l_j are also close. To achieve so, a reasonable way is to minimize the following function:

$$\begin{aligned} \frac{1}{2}\sum_{i,j=1}^{N} \|l_i - l_j\|_2^2 W_{ij} &= \sum_{i=1}^{N} l_i^T l_i D_{ii} - \sum_{i,j=1}^{N} l_i^T l_j W_{ij} \\ &= \mathrm{Tr}(\mathcal{L}^{(i)} D \mathcal{L}^{(i)T}) - \mathrm{Tr}(\mathcal{L}^{(i)} W \mathcal{L}^{(k)T}) \\ &= \mathrm{Tr}(\mathcal{L}^{(i)} \mathcal{G}^{(i)} \mathcal{L}^{(i)T}), \end{aligned}$$

where l_i, l_j represents the point in span space of $\mathcal{L}^{(i)}$, $\mathrm{Tr}(\cdot)$ stands for the trace of a matrix, D is a diagonal matrix in which $D_{ii} = \sum_j W_{ij}$, and $\mathcal{G}^{(i)} = D - W$ is the graph Laplacian matrix [27–30]. Incorporating this graph regularization term into Eq. (1), we propose graph-regularized tensor robust principal component analysis, which is formulated as follows:

$$\begin{aligned} &\min_{\mathcal{L},\mathcal{S}} \|\mathcal{L}\|_* + \lambda\|\mathcal{S}\|_1 + \gamma \sum_{i=1}^{n_3} \mathrm{Tr}(\mathcal{L}^{(i)} \mathcal{G}^{(i)} \mathcal{L}^{(i)T}) \\ &\text{s.t. } \mathcal{X} = \mathcal{L} + \mathcal{S}, \end{aligned} \tag{7}$$

where the sparse outliers in the data are modeled by $\mathcal{S}$ and $\mathcal{L}$ is the low tensor multi-rank approximation of $\mathcal{X}$. Parameters λ

and γ control the amount of sparsity of $\mathcal{S}$ and smoothness of $\mathcal{L}$ on the graph $\mathcal{G}$, respectively.

We name our proposed model as graph-regularized tensor robust principal component analysis. An ADMM-based [31–34] algorithm is proposed to solve this optimization problem in Subsection 3.C.

B. Graph Generation

We build the undirected graph $\mathcal{G}$ from the input HSI by using a standard and fast K-nearest neighbor strategy. The first step is to search the closest neighbors for all the subimages of HSI using Euclidean distances. We connect each x_i to its nearest neighbors x_j, resulting in the number of connections. The K-nearest neighbors are nonsymmetric. The second step is to compute the graph weight matrix W as

$$W_{ij} = \begin{cases} \exp\left(-\frac{\|(x_i - x_j)\|_2^2}{\sigma^2}\right) & \text{if } x_j \text{ is connected to } x_i, \\ 0 & \text{otherwise.} \end{cases}$$

The parameter σ can be set empirically. Finally, in the third step, the normalized graph Laplacian $\mathcal{G}^{(i)} = D - W$ is calculated, where D is the diagonal degree matrix $D_{ii} = \sum_j W_{ij}$.

For all the experiments in this paper, we use the following parameters for graph $\mathcal{G}^{(i)}$. K-nearest neighbors $= 12$ and $\sigma^2 = 0.02$. It is important to point out here that different types of data might call for slightly different parameters for graphs. However, for a given dataset, the use of the same graph parameters for all the graph-regularized models ensures a fair comparison.

C. Algorithm

By following the ADMM [31], Eq. (7) is first converted into the equivalent problem

$$\begin{aligned} \min_{\mathcal{L},\mathcal{S},\mathcal{W}} & \|\mathcal{L}\|_* + \lambda\|\mathcal{S}\|_1 + \gamma \sum_{i=1}^{n_3} \mathrm{Tr}(\mathcal{W}^{(i)}\mathcal{G}^{(i)}\mathcal{W}^{(i)T}) \\ \text{s.t. } & \mathcal{X} = \mathcal{L} + \mathcal{S} \\ & \mathcal{L} = \mathcal{W}, \end{aligned} \tag{8}$$

where $\mathcal{W}$ is an auxiliary variable.

In order to efficiently solve the optimization Eq. (8), the augmented Lagrangian multiplier [35] method is utilized. Then the augmented Lagrange multipliers are added to the above objective function making it an unconstraint optimization problem. Following the method in [35], Eq. (8) can be transformed as the following augmented Lagrange function:

$$\begin{aligned} (\mathcal{L}_k, \mathcal{S}_k, \mathcal{W}_k) = \min_{\mathcal{L},\mathcal{S},\mathcal{W}} & \|\mathcal{L}\|_* + \lambda\|\mathcal{S}\|_1 \\ & + \gamma \sum_{i=1}^{n_3} \mathrm{Tr}(\mathcal{W}^{(i)}\mathcal{G}^{(i)}\mathcal{W}^{(i)T}) \\ & + \langle \mathcal{Z}_1^k, \mathcal{X} - \mathcal{L} - \mathcal{S}\rangle + \frac{\mu_1}{2}\|\mathcal{X} - \mathcal{L} - \mathcal{S}\|_F^2 \\ & + \langle \mathcal{Z}_2^k, \mathcal{W} - \mathcal{L}\rangle + \frac{\mu_2}{2}\|\mathcal{W} - \mathcal{L}\|_F^2; \\ \mathcal{Z}_1^k = \mathcal{Z}_1^k + \mu_1(\mathcal{X} - \mathcal{L}_k - \mathcal{S}_k); & \\ \mathcal{Z}_2^k = \mathcal{Z}_2^k + \mu_2(\mathcal{W}_k - \mathcal{L}_k). & \end{aligned} \tag{9}$$

where $\mathcal{Z}_1$ and $\mathcal{Z}_2$ are the Lagrange multipliers. The above equation can be solved by fixing the other variables and optimizing it with only one variable.

Fixing variables $\mathcal{S}$ and $\mathcal{W}$, update $\mathcal{L}_{k+1}$ by

$$\begin{aligned} \mathcal{L}_{k+1} &= \arg\min_{\mathcal{L}} \|\mathcal{L}\|_* + \frac{\mu_1}{2}\left\|\mathcal{L} - \left(\mathcal{X} - \mathcal{S}_k + \frac{\mathcal{Z}_1^k}{\mu_1}\right)\right\|_F^2 \\ &\quad + \frac{\mu_2}{2}\left\|\mathcal{L} - \left(\mathcal{W}_k + \frac{\mathcal{Z}_2^k}{\mu_2}\right)\right\|_F^2 \\ &= \arg\min_{\mathcal{L}} \|\mathcal{L}\|_* + \frac{\mu_1 + \mu_2}{2}\left\|\mathcal{L} - \frac{\mu_1\mathcal{H}_1^k + \mu_2\mathcal{H}_2^k}{\mu_1 + \mu_2}\right\|_F^2, \end{aligned} \tag{10}$$

where $\mathcal{H}_1^k = \mathcal{X} - \mathcal{S}_k + \mathcal{Z}_1^k/\mu_1$ and $\mathcal{H}_2^k = \mathcal{W}_k + \mathcal{Z}_2^k/\mu_2$. The solution of Eq. (10) is given by the iterative singular-tubal shrinkage using convolution operation between the singular tubes and a tube of threshold vectors [15].

Following a similar procedure, we can write the update for $\mathcal{S}_{k+1}$:

$$\mathcal{S}_{k+1} = \arg\min_{\mathcal{S}} \|\mathcal{S}\|_1 + \frac{\mu_1}{2}\left\|\mathcal{S} - \left(\mathcal{X} - \mathcal{L}_{k+1} + \frac{\mathcal{Z}_1^k}{\mu_1}\right)\right\|_F^2. \tag{11}$$

By keeping only the terms with $\mathcal{W}$ in Eq. (9), we get

$$\begin{aligned} \mathcal{W}_{k+1} = \arg\min_{\mathcal{W}} & \gamma \sum_{i=1}^{n_3} \mathrm{Tr}(\mathcal{W}^{(i)}\mathcal{G}^{(i)}\mathcal{W}^{(i)T}) \\ & + \frac{\mu_2}{2}\left\|\mathcal{W} - \left(\mathcal{L}_k - \frac{\mathcal{Z}_2^k}{2}\right)\right\|_F^2. \end{aligned} \tag{12}$$

See Algorithm 2 for the optimization details. The shrinkage operation in the Fourier domain on the singular values of the tensor is computed by Algorithm 1.

Algorithm 2: ***Algorithm for GTRPCA***

```
Procedure GTRPCA (Data $\mathcal{X} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$, graph $\mathcal{G}$, $\lambda, \gamma, \varepsilon$).
    $\mu_1 = 10^{-6}, \mu_2 = 10^{-6}, \mu_{\max} = 10^6, \rho = 1.1$.
    while $\|\mathcal{L}_{k+1} - \mathcal{L}_k\| \geq \varepsilon, \|\mathcal{S}_{k+1} - \mathcal{S}_k\| \geq \varepsilon$ do
        Update $\mathcal{L}_{k+1}$ as Eq. (10);
        Update $\mathcal{S}_{k+1}$ as Eq. (11);
        Update $\mathcal{W}_{k+1}$ as Eq. (12);
        Update the multipliers $\mathcal{Z}_1^k, \mathcal{Z}_2^k$ as Eq. (9);
        Update the parameter:
        $\mu_1 = \min(\rho\mu_1, \mu_{\max})$;
        $\mu_2 = \min(\rho\mu_2, \mu_{\max})$;
    return $\mathcal{L}_{k+1}, \mathcal{S}_{k+1}$.
```

4. EXPERIMENTS

The experiments are conducted on both synthetic and real captured datasets. Both the qualitative and quantitative comparison with the state-of-the-art algorithms demonstrate the effectiveness of the proposed methods. The basic parameters are set as follows: iteration number $k_{\max} = 100$ and regularization parameter $\lambda = \frac{1}{\sqrt{\max(n_1, n_2) n_3}}$ [26], $\varepsilon = 1e - 8$, $\gamma = 0.1$.

A. Synthetic Hyperspectral Images Denoising

A synthetic HSI dataset [36] consisting of a variety of different scenes and objects is utilized to test the performance of the proposed method. Each HSI has 512 × 512 spatial pixels and

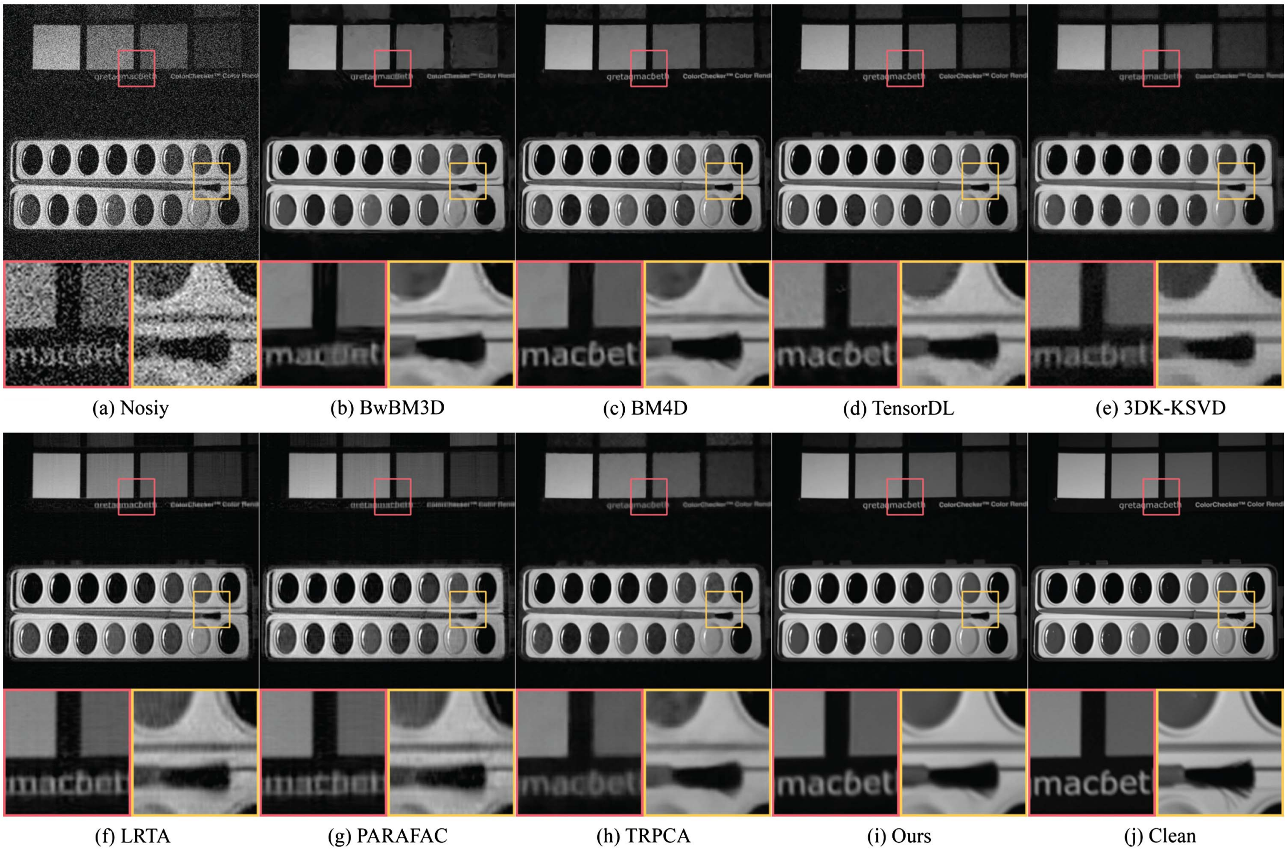


Fig. 3. Images at 530 nm of the *Paints* dataset. (a) The corresponding image corrupted by Gaussian noise with variance $v = 0.15$. (b)–(g) The restored image obtained by the seven utilized HSI denoising methods. (h) The restored image obtained by the no graph (TRPCA) HSI denoising method. (i) The restored image obtained by the graph-regularized (GTRPCA) HSI denoising method. (j) The image of clean data (ground truth). The red and yellow rectangular areas in each image are zoomed up and the same degree of contrast for easy observation of details.

31 spectral bands (ranging from 400 to 700 nm at an interval of 10 nm). The 16-bit image intensity is scaled to [0, 1]. Additive Gaussian noise with different standard derivations (which is denoted as v) is added to the HSIs.

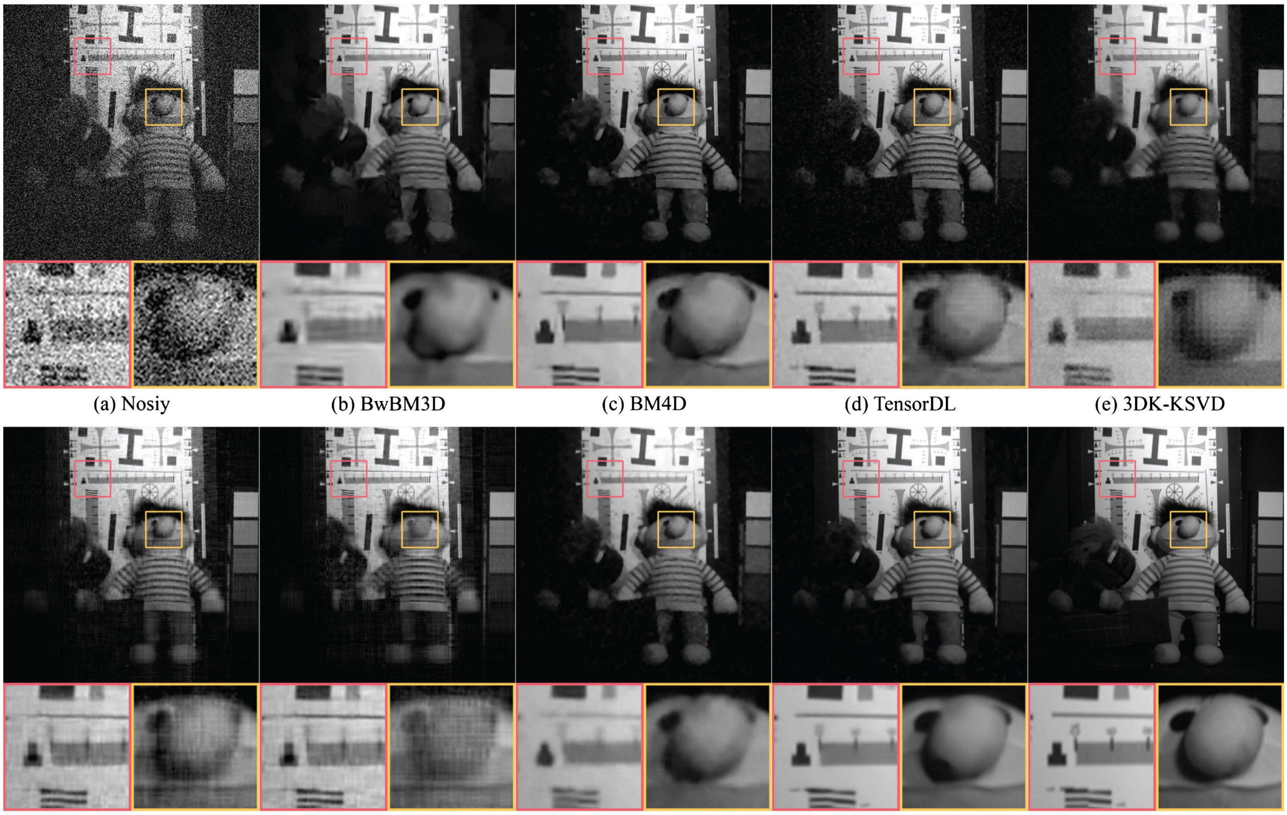
We have compared the proposed method with several recently developed multi-spectral image denoising methods, including band-wise BM3D [7], block-matching and 4-D filtering (BM4D) [8], tensor dictionary learning (TensorDL) [11], 3 dimension k-SVD (3DK-SVD) [9], low-rank tensor approximation (LRTA) [10], PARAFAC [5], and TRPCA [15]. To further verify the effectiveness of the proposed graph, the TRPCA method is also included.

To quantitatively evaluate the performance, the peak signal-to-noise ratio (PSNR) [37] and structure similarity (SSIM) [38] indices are employed. Both PSNR and SSIM are calculated on each 2D spectral band image and averaged over the spectral dimension.

Real-world HSIs are usually contaminated by several different types of noises, including the most common Gaussian noise, mixture noise, and strips [39]. In order to simulate these

Table 1. Quantitative Comparison for Different Noise Levels on the Synthetic Dataset

Noisy Type	Gaussian Noisy						Striping Noisy		Mixture Noisy	
	$v = 0.15$		$v = 0.20$		$v = 0.25$				$v = 0.15 - 0.25$	
Methods	PSNR (dB)	SSIM	PSNR (dB)	SSIM	PSNR (dB)	SSIM	PSNR (dB)	SSIM	PSNR (dB)	SSIM
Noisy	16.474	0.141	13.975	0.098	12.037	0.072	23.776	0.372	21.215	0.325
BwBM3D [7]	33.326	0.918	31.847	0.900	30.479	0.873	33.920	0.864	30.898	0.813
BM4D [8]	36.213	0.947	35.45	0.906	**34.421**	**0.887**	35.998	0.896	33.147	0.891
TensorDL [11]	35.329	0.941	33.659	0.913	32.284	0.883	28.615	0.617	24.051	0.479
3DK-SVD [9]	29.648	0.770	28.191	0.735	26.899	0.701	28.190	0.764	24.841	0.710
LRTA [10]	33.373	0.892	31.920	0.865	30.867	0.836	30.269	0.693	22.345	0.408
PARAFAC [5]	31.045	0.875	30.455	0.841	29.526	0.793	27.261	0.733	26.414	0.730
TRPCA [15]	32.965	0.935	31.385	0.910	30.238	0.883	35.145	0.909	32.508	0.924
Ours	**36.831**	**0.973**	**35.6233**	**0.9125**	34.417	0.886	**36.296**	**0.899**	**34.086**	**0.919**



(a) Nosiy (b) BwBM3D (c) BM4D (d) TensorDL (e) 3DK-KSVD

(f) LRTA (g) PARAFAC (h) TRPCA (i) Ours (j) Clean

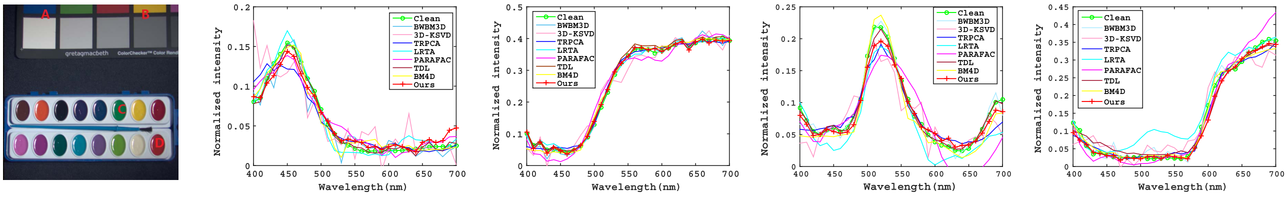
Fig. 4. Image at 700 nm of the *chart and staffed toy* dataset. (a) The corresponding images corrupted by Gaussian noise with variance $v = 0.20$. (b)–(g) The restored image obtained by the seven utilized HSI denoising methods. (h) The restored image obtained by the no-graph (TRPCA) HSI denoising method. (i) The restored image obtained by the graph-regularized (GTRPCA) HSI denoising method. (j) The image of clean data (ground truth). The red and yellow rectangular areas in each image are zoomed up and the same degree of contrast for easy observation of details.

real HSI noise scenarios, we added three kinds of noises to the original HSI data.

- Gaussian noise: Entries in all bands were corrupted by zero-mean i.i.d. Gaussian noise $\mathcal{N}(0, v^2)$.
- Striping noise: 20 bands in synthetic data were randomly chosen to add stripe noise. The number of strips in each band is ranging from 20 to 40.
- Mixture noise: Each band was randomly corrupted by at least one kind of Gaussian noise or stripe noise.

The average assessments are shown in Table 1. It is obvious that our method outperforms the other methods by at least 0.62 dB and 0.17 dB PSNR improvement, and 0.026 and 0.02 SSIM improvement at $v = 0.15$ and $v = 0.20$, respectively, which has demonstrated the effectiveness of our proposed method. In addition, when the noise level continues to grow ($v = 0.25$), our method also achieves comparable performance with the best one. On the more practical complicated non-i.i.d. Gaussian noise case, the advantage of the proposed method is evident.

For qualitative comparison, we also show the reconstructed 2D image at 530 nm of the Paints dataset in Fig. 3 and the results at the 700 nm band of the Chart and Staffed toy dataset in Fig. 4. The close-up of the region marked by the red box is given to show details. It should be noted that our method performs well both for large structures and fine details, while



(a) (b) (c) (d) (e)

Fig. 5. Graph method (GTRPCA) and no-graph method (TRPCA) compared to verify the effectiveness of GTRPCA in synthetic dataset (*Paints*) corrupted by Gaussian noise with variance $v = 0.15$. (a) Four selected spatial points in the original hyperspectral image (synthesized to the RGB format). (b)–(e) The restored spectra of points A, B, C, and D by the different utilized HSI denoising methods.

Table 2. Pearsonian Correlation Coefficient Comparison for Different Denoising Method on the Synthetic Dataset

Method	BWKSVD	BWBM3D	KSVD	TRPCA	LRTA	PARAFAC	TDL	BM4D	Ours
PCC [40]	0.2044	0.8286	0.5518	0.8010	0.9106	0.8231	0.8959	0.8075	**0.9524**

BwB3D [7] and BM4D [8] over smooth the fine details and TRPCA [15] and TensorDL [11] induce severe artifacts.

For an intuitive comparison in the spectral domain, Fig. 5 shows the reconstructed spectra at four distinct pixels. It can be seen that the spectral signature reconstructed by our method is closest to the ground truth.

We attempt to illustrate that the proposed GTRPCA method can still preserve the useful spectral information of the original hyperspectral image while removing the noise. To this end, the reflectances of several points are presented with no graph and graph denoising, respectively. The reflectance spectrum at each pixel was computed from the measured multi-spectrum image using calibrated estimates of the illumination spectrum and camera spectral response (http://www.cs.columbia.edu/CAVE/databases/multispectral/). Figure 5 shows the reflectance of different points (A and B) from bands 1–31 of the synthetic HSI dataset. It can be seen form Fig. 5 that most of the spectral information is well preserved when using the graph TRPCA method. The reason for the difference of the reflectance near band 31 may be attributed to the impact of noising in the sub-image in band 31, which also demonstrates the ability of GTRPCA in HSI denoising to some extent. Here, we observe that, in all cases, we obtain higher denoising accuracy in GTRPCA representation.

To further verify the effectiveness of the proposed approach, we have used the Pearsonian correlation coefficient (PCC) [40,41] and S-CIELAB [42] to evaluate the efficiency of the proposed method. PCC comparison for different noising methods on the synthetic dataset is presented in Table 2. We calculate the S-CIELAB error image as [42], and the results are shown in Fig. 6. It obviously shows that our proposed method is more effective than the others.

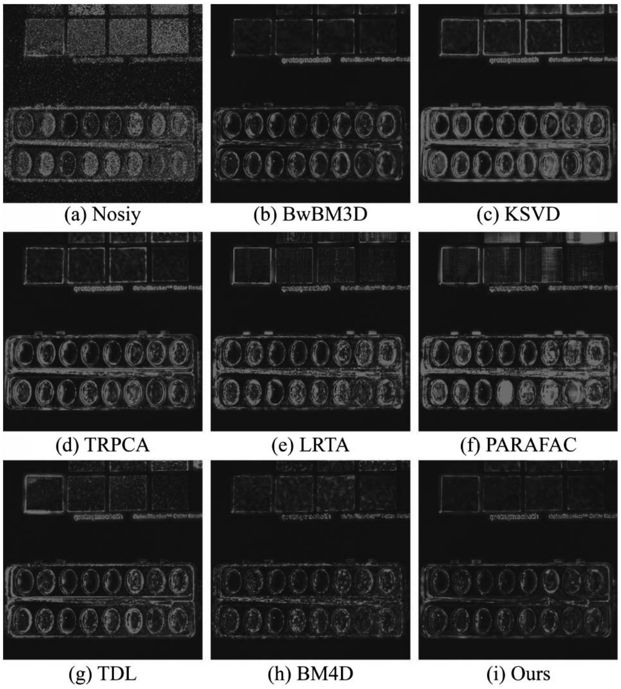


Fig. 6. S-CIELAB error images by the different utilized HSI denoising methods.

B. Real Hyperspectral Images Denoising

1. Indian Pines Datasets

The AVIRIS Indian Pines test datasets, which were provided by professor David Landgrebe and can be downloaded from https://purr.purdue.edu/publications/1947/1, are used in the first real data experiment. The data size is 145 $*$ 145 pixels, with 220 bands. Before the denoising processing, the atmospheric and water absorption bands from bands 150–163 were

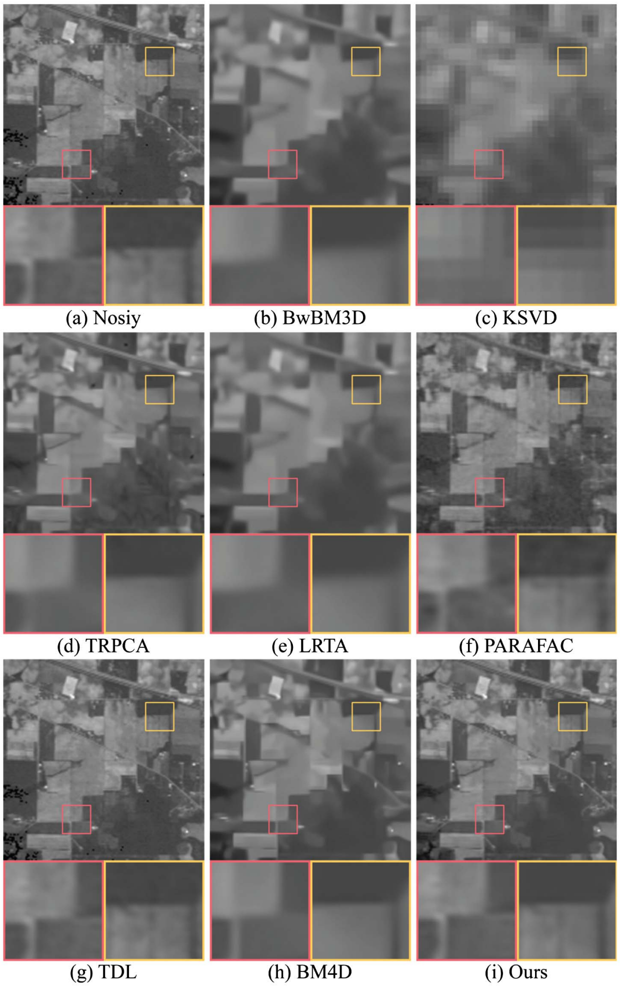


Fig. 7. Qualitative comparison on the real-world AVIRIS Indian Pines datasets (images at third band). The restored image obtained by the different HSI denoising methods.

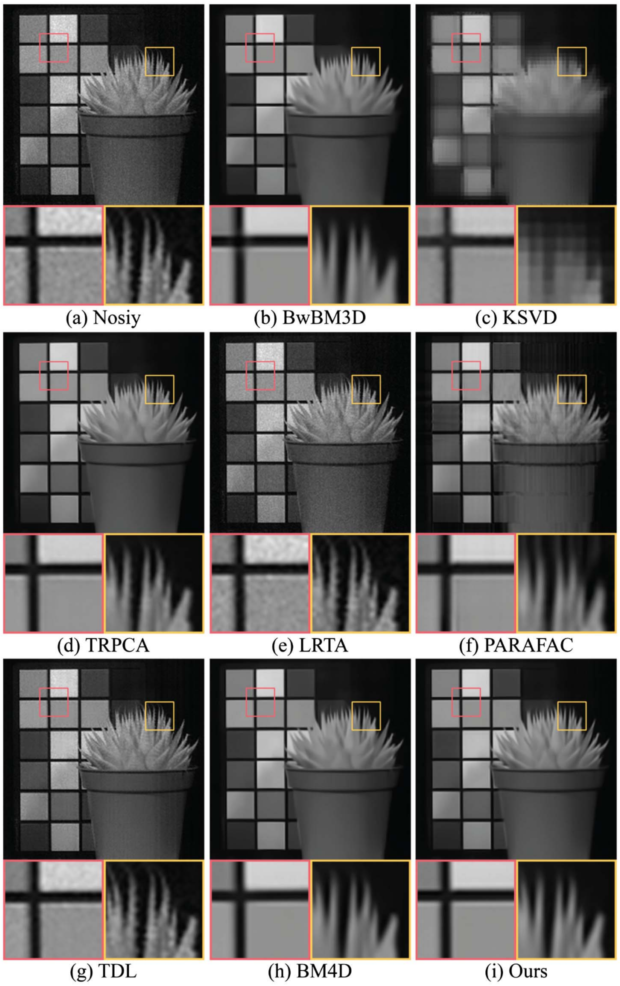


Fig. 8. Qualitative comparison on the real-world captured images (band images at 740 nm). The recovered image obtained by the different HSI denoising methods.

removed from the original hyperspectral image. Therefore, there were only 206 bands used in the experiment.

The denoised images at the third band are shown in Fig 7. It can be clearly seen that the proposed GTRPCA gives better denoised results than the other seven methods.

The reference noise-free image is not available in the Indian Pines dataset. Thus, the quantitative evaluations cannot be implemented on each denoised image to evaluate the denoising algorithms.

2. *Real Captured Hyperspectral Images Denoising*

We also test the proposed method on the real captured hyperspectral image. The image consists of 214 ∗ 407 spatial pixels and 25 spectral bands (ranging from 600 to 1000 nm), which was captured by the BaySpec Compact OCI(TM)-2000 Snapshot Spectral Imager (http://www.bayspec.com/spectroscopy/snapshot-hyperspectral-imager/). As the scene is illuminated by the poor reflected sunlight into the room, the captured image is severely polluted by noise.

We show the denoising results at the 740 nm band of the test HSI in Fig. 8. It can be easily observed that the image restored by our method is capable of properly removing the noise while finely preserving the structure underlying the HSI.

5. CONCLUSION

In this paper, we provided a novel HSI denoising model that incorporates spectral graph regularization into the tensor robust PCA framework. We also designed an efficient ADMM algorithm to solve the model. The experiments on both synthetic and real-captured data have demonstrated the superiority of the proposed method beyond the existing methods.

Funding. Science and Technology Support Program of Jiangsu Province (BE2015152); National Natural Science Foundation of China (NSFC) (61422107, 61627804, 61671236).

Acknowledgment. We would like to acknowledge funding from the National Science Foundation of China; National Science Foundation for Young Scholar of Jiangsu Province, China; and Key Research Projects of Jiangsu Province of China. The authors also would like to thank F. Yasuma, T. Mitsunaga, D. Iso, and S.K. Nayar for their hyperspectral datasets.

REFERENCES

1. A. A. Green, M. Berman, P. Switzer, and M. D. Craig, "A transformation for ordering multispectral data in terms of image quality with implications for noise removal," IEEE Trans. Geosci. Remote Sens. **26**, 65–74 (1988).
2. D. Goldfarb and Z. Qin, "Robust low-rank tensor recovery: models and algorithms," SIAM J. Matrix Anal. Appl. **35**, 225–253 (2013).
3. J. Liu, P. Musialski, P. Wonka, and J. Ye, "Tensor completion for estimating missing values in visual data," in *IEEE International Conference on Computer Vision* (2013), pp. 2114–2121.
4. D. Kressner, M. Steinlechner, and B. Vandereycken, "Low-rank tensor completion by Riemannian optimization," Bit Numer. Math. **54**, 447–468 (2014).
5. X. Liu, S. Bourennane, and C. Fossati, "Denoising of hyperspectral images using the PARAFAC model and statistical performance analysis," IEEE Trans. Geosci. Remote Sens. **50**, 3717–3724 (2012).
6. X. Cao, T. Yue, X. Lin, S. Lin, X. Yuan, Q. Dai, L. Carin, and D. J. Brady, "Computational snapshot multispectral cameras: toward dynamic capture of the spectral world," IEEE Signal Process. Mag. **33**(5), 95–108 (2016).
7. K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "Image denoising by sparse 3-d transform-domain collaborative filtering," IEEE Trans. Image Process. **16**, 2080–2095 (2007).
8. M. Maggioni, V. Katkovnik, K. Egiazarian, and A. Foi, "Nonlocal transform-domain filter for volumetric data denoising and reconstruction," IEEE Trans. Image Process. **22**, 119–133 (2013).
9. M. Elad and M. Aharon, "Image denoising via sparse and redundant representations over learned dictionaries," IEEE Trans. Image Process. **15**, 3736–3745 (2006).
10. N. Renard, S. Bourennane, and J. Blanc-Talon, "Denoising and dimensionality reduction using multilinear tools for hyperspectral images," IEEE Geosci. Remote Sens. Lett. **5**, 138–142 (2008).
11. Y. Peng, D. Meng, Z. Xu, and C. Gao, "Decomposable nonlocal tensor dictionary learning for multispectral image denoising," in *IEEE Conference on Computer Vision and Pattern Recognition* (IEEE, 2014), pp. 2949–2956.
12. G. Zhong and M. Cheriet, "Large margin low rank tensor analysis," Neural Comput. **26**, 761–780 (2014).

13. R. Tomioka, K. Hayashi, and H. Kashima, “Estimation of low-rank tensors via convex optimization,” arXiv preprint arXiv: 1010.0789 (2010).
14. M. Zheng, J. Bu, C. Chen, and C. Wang, “Graph regularized sparse coding for image representation,” IEEE Trans. Image Process. **20**, 1327–1336 (2011).
15. Z. Zhang, G. Ely, S. Aeron, and N. Hao, “Novel methods for multilinear data completion and de-noising based on tensor-SVD,” in *IEEE Conference on Computer Vision and Pattern Recognition* (IEEE, 2014), pp. 3842–3849.
16. N. Keshava and J. F. Mustard, “Spectral unmixing,” IEEE Signal Process. Mag. **19**(1), 44–57 (2002).
17. N. Shahid, N. Perraudin, V. Kalofolias, and G. Puy, “Fast robust PCA on graphs,” IEEE J. Sel. Top. Signal Process. **10**, 740–756 (2016).
18. D. Cai, X. He, J. Han, and T. S. Huang, “Graph regularized nonnegative matrix factorization for data representation,” IEEE Trans. Pattern Anal. Mach. Intell. **33**, 1548–1560 (2010).
19. S. Gao, W. H. Tsang, and L. T. Chia, “Laplacian sparse coding, hypergraph Laplacian sparse coding, and applications,” IEEE Trans. Pattern Anal. Mach. Intell. **35**, 92–104 (2013).
20. K. Braman, “Third-order tensors as linear operators on a space of matrices,” Linear Algebra Appl. **433**, 1241–1253 (2010).
21. M. E. Kilmer and C. D. Martin, “Factorization strategies for third-order tensors,” Linear Algebra Appl. **435**, 641–658 (2011).
22. M. E. Kilmer, K. Braman, N. Hao, and R. C. Hoover, “Third-order tensors as operators on matrices: a theoretical and computational framework with applications in imaging,” SIAM J. Matrix Anal. Appl. **34**, 148–172 (2013).
23. T. G. Kolda and B. W. Bader, “Tensor decompositions and applications,” SIAM Rev. **51**, 455–500 (2009).
24. O. Semerci, N. Hao, M. E. Kilmer, and E. L. Miller, “Tensor-based formulation and nuclear norm regularization for multienergy computed tomography,” IEEE Trans. Image Process. **23**, 1678–1693 (2014).
25. C. Lu, J. Feng, Y. Chen, W. Liu, Z. Lin, and S. Yan, “Tensor robust principal component analysis: exact recovery of corrupted low-rank tensors via convex optimization,” in *IEEE Conference on Computer Vision and Pattern Recognition* (IEEE, 2016), pp. 5249–5257.
26. E. J. Candès, X. Li, Y. Ma, and J. Wright, “Robust principal component analysis?” J. ACM **58**, 1–37 (2011).
27. B. Jiang, C. Ding, B. Luo, and J. Tang, “Graph-Laplacian PCA: closed-form solution and robustness,” in *IEEE Conference on Computer Vision and Pattern Recognition* (IEEE, 2013), pp. 3492–3498.
28. Z. Zhang and K. Zhao, “Low-rank matrix approximation with manifold regularization,” IEEE Trans. Pattern Anal. Mach. Intell. **35**, 1717–1729 (2013).
29. T. Jin, J. Yu, J. You, K. Zeng, C. Li, and Z. Yu, “Low-rank matrix factorization with multiple hypergraph regularizer,” Pattern Recogn. **48**, 1011–1022 (2015).
30. L. Tao, H. H. Ip, Y. Wang, and X. Shu, “Low rank approximation with sparse integration of multiple manifolds for data representation,” Appl. Intell. **42**, 430–446 (2015).
31. S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, “Distributed optimization and statistical learning via the alternating direction method of multipliers,” Found. Trends Mach. Learn. **3**, 1–122 (2011).
32. B. Romera-paredes and M. Pontil, “A new convex relaxation for tensor completion,” in *Advances in Neural Information Processing Systems 26* (2013), pp. 2967–2975.
33. S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, “Distributed optimization and statistical learning via the alternating direction method of multipliers,” Found. Trends Mach. Learn. **3**, 1–122 (2010).
34. Z. Zhou, X. Li, J. Wright, and E. Candes, “Stable principal component pursuit,” in *IEEE International Symposium on Information Theory Proceedings* (2010), pp. 1518–1522.
35. Z. Lin, M. Chen, and Y. Ma, “The augmented Lagrange multiplier method for exact recovery of corrupted low-rank matrices,” arXiv: 1009.5055 (2009).
36. F. Yasuma, T. Mitsunaga, D. Iso, and S. K. Nayar, “Generalized assorted pixel camera: postcapture control of resolution, dynamic range, and spectrum,” IEEE Trans. Image Process. **19**, 2241–2253 (2010).
37. L. Zhang, D. Zhang, X. Mou, and D. Zhang, “FSIM: a feature similarity index for image quality assessment,” IEEE Trans. Image Process. **20**, 2378–2386 (2011).
38. Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, “Image quality assessment: from error visibility to structural similarity,” IEEE Trans. Image Process. **13**, 600–612 (2004).
39. H. Zhang, W. He, L. Zhang, H. Shen, and Q. Yuan, “Hyperspectral image restoration using low-rank matrix recovery,” IEEE Trans. Geosci. Remote Sens. **52**, 4729–4743 (2014).
40. O. A. de Carvalho, Jr. and P. R. Meneses, “Spectral correlation mapper (SCM): an improvement on the spectral angle mapper (SAM),” in *Summaries of the 9th JPL Airborne Earth Science Workshop* (JPL, 2000), Vol. **9**.
41. O. De Carvalho, R. F. Guimarães, R. A. T. Gomes, N. C. D. Silva, and E. Martins, “Spectral multiple correlation mapper,” in *IEEE International Conference on Geoscience and Remote Sensing Symposium, IGARSS* (IEEE, 2006), pp. 2773–2776.
42. X. Zhang and B. A. Wandell, “A spatial extension of CIELAB for digital color image reproduction,” J. Soc. Inf. Display **5**, 61–63 (1997).