

REVIEW

Open Access

Survey of hyperspectral image denoising methods based on tensor decompositions

Tao Lin and Salah Bourennane*

Abstract

A hyperspectral image (HSI) is always modeled as a three-dimensional tensor, with the first two dimensions indicating the spatial domain and the third dimension indicating the spectral domain. The classical matrix-based denoising methods require to rearrange the tensor into a matrix, then filter noise in the column space, and finally rebuild the tensor. To avoid the rearranging and rebuilding steps, the tensor-based denoising methods can be used to process the HSI directly by employing multilinear algebra. This paper presents a survey on three newly proposed HSI denoising methods and shows their performances in reducing noise. The first method is the Multiway Wiener Filter (MWF), which is an extension of the Wiener filter to data tensors, based on the TUCKER3 decomposition. The second one is the PARAFAC filter, which removes noise by truncating the lower rank K of the PARAFAC decomposition. And the third one is the combination of multidimensional wavelet packet transform (MWPT) and MWF (MWPT-MWF), which models each coefficient set as a tensor and then filters each tensor by applying MWF. MWPT-MWF has been proposed to preserve rare signals in the denoising process, which cannot be preserved well by using the MWF or PARAFAC filters. A real-world HYDICE HSI data is used in the experiments to assess these three tensor-based denoising methods, and the performances of each method are analyzed in two aspects: signal-to-noise ratio and improvement of subsequent target detection results.

1 Review

1.1 Introduction

Hyperspectral images (HSI) attract more and more interest in recent years in different domains, such as geography, agriculture, and military [1-3]. They use the HSI to do the target detection [4] or classification [5] to find objects or materials of interest on the ground. Unfortunately, in the capturing procedure, the HSI is usually impaired by several types of noise, such as thermal noise [6], photonic noise [7], and strip noise [8]. Therefore, denoising methods [9-13] have become a critical step for improving the subsequent target detection and classification in remote sensing imaging applications [14].

In HSI processing, images are modeled as a three-dimensional tensor, i.e., two spatial dimensions and one spectral dimension. The classical denoising methods [15-18] rearrange the HSI into a matrix whose columns contain the spectral signatures of all the pixels, then estimate the signal subspace by methods based on the analysis

of second-order statistics, and finally rebuild the original HSI structure after processing.

Since matrix-based techniques cannot take advantage of spectra in hyperspectral images, therefore, in order to treat the HSI as a whole entity, some new techniques were developed. For example, an HSI was treated as a hypercube in order to take into account the correlation among different bands [19,20], tensor-algebra was brought to jointly analyze the 3D HSI, etc. In this paper, we mainly focus on the problem of applying tensor algebra in reducing noise in HSIs. Unlike the matrix-based denoising methods which are based on matrix algebra, the newly proposed tensor-based denoising methods utilize multilinear algebra to analyze the HSI tensor directly. It is well known that SVD (singular value decomposition) is important for matrix analysis. Similarly, there are two important tensor decompositions: TUCKER3 and PARAFAC. These two decompositions play significant roles in analyzing tensors. Therefore, in this paper, we focus on comparative methods based on multilinear algebra for sake of coherence with the recently developed method which combines multidimensional wavelet packet transform and

*Correspondence: salah.bourennane@fresnel.fr
Institut Fresnel/CNRS-UMR 7249 Ecole Centrale Marseille, Aix-Marseille
Université, Marseille 13013, France

TUCKER3 decomposition: The three methods involve a tensor decomposition either TUCKER3 or PARAFAC.

TUCKER3 decomposition, also known as lower rank- (K_1, \dots, K_N) tensor approximation (LRTA- (K_1, \dots, K_N)), has been firstly used as multimode PCA, which uses the first K_n PCA components in mode n , $n = 1, \dots, N$, to restore the multidimensional signal. The LRTA- (K_1, \dots, K_N) has been employed for seismic wave separation [21], face recognition [22], and color image denoising [23]. Although the LRTA- (K_1, \dots, K_N) can obtain good results in denoising, it is not an optimal solution for filtering noise in the aspect of the mean squared error (MSE). The multidimensional Wiener filter (MWF) has been proposed to overcome this drawback of LRTA- (K_1, \dots, K_N) [24]. MWF calculates the filter in each mode under the criterion of minimizing the MSE between the desired signal and the estimated signal, therefore it can be understood as an optimal LRTA- (K_1, \dots, K_N) . Moreover, MWF can also be understood as an extension of the classical matrix-based Wiener filter to the tensor model by using multilinear algebra tools. MWF has been used in seismic wave denoising [24] and HSI denoising [12,25], and obtained good results. Recently, a statistical criteria has been adapted to estimate the rank of signal subspace in each mode [13], which makes MWF an automatic method to reduce noise in the data.

Apart from TUCKER3, the PARAFAC [26] decomposition, also known as CANDECOMP [27], is another way to decompose a tensor into lower rank factors. Distinguishing from TUCKER3, PARAFAC decomposes a tensor into a sum of rank-one tensors and only one rank K needs to be estimated for the tensor. Moreover, the PARAFAC decomposition is unique when the rank K is greater than one, whereas TUCKER3 cannot be. PARAFAC decomposition has recently been applied to chemical sciences [28], array processing [29], telecommunications [30], and HSI denoising [14]. As a comparison of MWF, reference [31] shows the potential of PARAFAC in the HSI denoising. However, there is not an efficient way to estimate the rank of PARAFAC, which constrains it in automatic denoising.

In a HSI, a rare signal is the one that is represented by only a few number of pixels, while the abundant signal is the one that contains a large number of pixels compared to a rare signal [17]. MWF and PARAFAC treat a HSI as a whole entity in the denoising operation; therefore, the abundant signals and the rare signals are processed together, which inhibits a drawback: the rare signals may be unintentionally removed. In fact, the energy of the rare signal is so weak compared to that of the abundant signal that the estimated signal subspace cannot include the rare signal, and as a result, the rare signal is removed. MWPT-MWF (multidimensional wavelet packet transform (MWPT) with multiway

Wiener filter) has been proposed to overcome this drawback of MWF and PARAFAC [32]. Instead of treating the HSI as a whole entity, MWPT-MWF firstly decomposes the HSI into several coefficient sets, also called components, by employing MWPT, therefore the abundant signal and the rare signal can be separated. After this step, each component is filtered by MWF automatically. Because the rare signal and the abundant signal are separated into different components, the signal subspace in each component can be estimated more exactly.

The goal of this paper is to present a survey of the tensor-based denoising methods applied in filtering the HSI. Some recent simulations and comparative results on a real-world HYDICE HSI are also presented. The remainder of this paper is organized as follows: Section 1.2 briefly introduces some basic knowledge about multilinear algebra. Section 1.3 introduces the signal model used in this paper. Sections 1.4, 1.5, and 1.6 present the recently proposed denoising methods MWF, PARAFAC, and MWPT-MWF, respectively. Section 1.7 supplies some comparative denoising and detection results. And finally, Section 2 concludes this paper.

1.2 Basics on tensor tools and multilinear algebra

1.2.1 Tensor model

A multiway signal is also called tensor. A tensor is a multidimensional array, $\mathcal{X} \in \mathbb{R}^{I_1 \times I_2 \times \dots \times I_N}$, in which \mathbb{R} indicates the real manifold, and N is the number of dimensions. The elements in this tensor can be expressed as $x_{i_1 i_2 \dots i_N}$, with $i_1 = 1, \dots, I_1$; $i_2 = 1, \dots, I_2$; \dots ; $i_N = 1, \dots, I_N$. The n -th dimension of this tensor is called n -mode. In particular, tensor \mathcal{X} is called a rank-one tensor when it can be written as the outer product of N vectors [33]:

$$\mathcal{X} = \mathbf{a}_1 \circ \dots \circ \mathbf{a}_N, \quad (1)$$

where \circ indicates the outer product [34].

1.2.2 Multilinear algebra tools

n -mode unfolding $\mathbf{X}_n \in \mathbb{R}^{I_n \times M_n}$ denotes the n -mode unfolding matrix of a tensor $\mathcal{X} \in \mathbb{R}^{I_1 \times I_2 \times \dots \times I_N}$, where $M_n = I_{n+1} \dots I_1 I_N \dots I_{n-1}$. The columns of \mathbf{X}_n are the I_n -dimensional vectors obtained from \mathcal{X} by varying index i_n while keeping the other indices fixed. Here, we define the n -mode rank K_n as the n -mode unfolding matrix rank, i.e., $K_n = \text{rank}(\mathbf{X}_n)$.

n -mode product The n -mode product is defined as the product between a data tensor $\mathcal{X} \in \mathbb{R}^{I_1 \times I_2 \times \dots \times I_N}$ and a matrix $\mathbf{B} \in \mathbb{R}^{I_n \times J_n}$ in mode n . This n -mode product is denoted by

$$\mathcal{C} = \mathcal{X} \times_n \mathbf{B}, \quad (2)$$

whose entries are given by

$$c_{i_1 \dots i_{n-1} j_{n+1} \dots i_N} \triangleq \sum_{i_n=1}^{I_n} x_{i_1 \dots i_{n-1} i_n i_{n+1} \dots i_N} b_{j_{i_n}} \quad (3)$$

where $\mathcal{C} \in \mathbb{R}^{I_1 \times I_2 \times \dots \times I_{n-1} \times J \times I_{n+1} \times \dots \times I_N}$.

1.3 Problem formulation and signal modeling

A noisy HSI is modeled as a tensor $\mathcal{R} \in \mathbb{R}^{I_1 \times I_2 \times I_3}$ resulting from a pure HSI $\mathcal{X} \in \mathbb{R}^{I_1 \times I_2 \times I_3}$ impaired by an additive noise $\mathcal{N} \in \mathbb{R}^{I_1 \times I_2 \times I_3}$. The tensor \mathcal{R} can be expressed as:

$$\mathcal{R} = \mathcal{X} + \mathcal{N}. \quad (4)$$

In this paper, we assume that the noise \mathcal{N} is zero-mean white Gaussian noise and independent from the signal \mathcal{X} . The aim of this paper was to estimate the desired signal \mathcal{X} from the noisy HSI \mathcal{R} .

1.4 Multiway Wiener filtering

1.4.1 Denoising model

MWF provides an estimate $\hat{\mathcal{X}}$ of the desired signal \mathcal{X} from data tensor \mathcal{R} by using a three-dimensional filtering, which can be expressed as follows [35]:

$$\hat{\mathcal{X}} = \mathcal{R} \times_1 \mathbf{H}_1 \times_2 \mathbf{H}_2 \times_3 \mathbf{H}_3. \quad (5)$$

From the signal processing point of view, the n -mode product is a n -mode filtering of \mathcal{R} ; therefore, \mathbf{H}_n is named as n -mode filter.

In order to obtain the optimal n -mode filters $\{\mathbf{H}_n, n = 1, 2, 3\}$, the usually used criterion is the mean squared error (MSE) between the estimated signal $\hat{\mathcal{X}}$ and the desired signal \mathcal{X} :

$$\begin{aligned} e(\mathbf{H}_1, \mathbf{H}_2, \mathbf{H}_3) &= E \left[\|\mathcal{X} - \hat{\mathcal{X}}\|^2 \right] \\ &= E \left[\|\mathcal{X} - \mathcal{R} \times_1 \mathbf{H}_1 \times_2 \mathbf{H}_2 \times_3 \mathbf{H}_3\|^2 \right]. \end{aligned} \quad (6)$$

Then, the optimal n -mode filters are the ones which can minimize the MSE given in (6).

1.4.2 Calculation of \mathbf{H}_n

To minimize the MSE given in (6) with respect to n -mode filters $\{\mathbf{H}_n, n = 1, 2, 3\}$, the derivation is employed and the calculation details are presented in [24]. By setting the derivation of the MSE to zero, the expression of the optimal n -mode filter \mathbf{H}_n is [24]:

$$\mathbf{H}_n = \mathbf{V}_s^{(n)} \Lambda^{(n)} \mathbf{V}_s^{(n)T}, \quad (7)$$

where $\mathbf{V}_s^{(n)}$ is a matrix containing the K_n orthonormal basis vectors of the signal subspace in the column space of the n -mode unfolding matrix \mathbf{R}_n , and

$$\Lambda^{(n)} = \text{diag} \left(\frac{\lambda_1^\gamma - \sigma_\gamma^{(n)2}}{\lambda_1^\Gamma}, \dots, \frac{\lambda_{K_n}^\gamma - \sigma_\gamma^{(n)2}}{\lambda_{K_n}^\Gamma} \right), \quad (8)$$

in which $\{\lambda_i^\gamma, i = 1, \dots, K_n\}$ and $\{\lambda_i^\Gamma, i = 1, \dots, K_n\}$ are the K_n largest eigenvalues of matrices $\gamma_{RR}^{(n)}$ and $\Gamma_{RR}^{(n)}$ respectively, where

$$\gamma_{RR}^{(n)} = E \left[\mathbf{R}_n \mathbf{Q}^{(n)} \mathbf{R}_n^T \right] \quad (9)$$

$$\Gamma_{RR}^{(n)} = E \left[\mathbf{R}_n \mathbf{Q}^{(n)} \mathbf{R}_n^T \right] \quad (10)$$

with

$$\mathbf{Q}^{(n)} = \mathbf{H}_{p_1} \otimes \mathbf{H}_{p_2}, \quad (11)$$

$$\mathbf{Q}^{(n)} = \mathbf{H}_{p_1}^T \mathbf{H}_{p_1} \otimes \mathbf{H}_{p_2}^T \mathbf{H}_{p_2} \quad (12)$$

where $p_1 \neq n, p_2 \neq n, p_1, p_2 = 1, 2, 3$ and \otimes defines the Kronecker product. Moreover, $\sigma_\gamma^{(n)2}$ is equal to the $I_n - K_n$ smallest eigenvalues $\{\lambda_i^\gamma, i = K_n + 1, \dots, I_n\}$ of $\gamma_{RR}^{(n)}$. However, in the practice, the $I_n - K_n$ smallest eigenvalues are generally different. Hence, $\sigma_\gamma^{(n)2}$ can be estimated by:

$$\hat{\sigma}_\gamma^{(n)2} = \frac{1}{I_n - K_n} \sum_{i=K_n+1}^{I_n} \lambda_i^\gamma. \quad (13)$$

1.4.3 Estimation of K_n

Being used in the computation of the n -mode filter \mathbf{H}_n , expression (7) requires the unknown K_n value, i.e., the number of largest eigenvalues of the covariance matrix of $\gamma_{RR}^{(n)}$, for $n=1, 2, 3$. Choosing a small K_n makes that some signals are lost whereas choosing a large K_n makes that noise is included after restoration. For this case, the optimal K_n should be estimated to yield an optimum restoration. Akaike information criterion (AIC) is a criterion used to measure the information lost; therefore, it is employed in MWF to determine the optimal rank K_n [13]. For mode n , the AIC can be expressed as:

$$\begin{aligned} \text{AIC}(k_n) &= -2M_n \sum_{i=k_n+1}^{I_n} \log \lambda_i^\gamma \\ &+ 2M_n(I_n - k_n) \log \left(\frac{1}{I_n - k_n} \sum_{i=k_n+1}^{I_n} \lambda_i^\gamma \right) \\ &+ 2k_n(2I_n - k_n), \end{aligned} \quad (14)$$

where $\{\lambda_i^\gamma, i = 1, \dots, I_n\}$ are the eigenvalues of $\gamma_{RR}^{(n)}$, M_n is the column number of $\gamma_{RR}^{(n)}$ and k_n changes in the range of $\{1, \dots, I_n - 1\}$. The estimated n -mode rank K_n is the value of k_n which minimizes AIC criterion.

1.4.4 ALS algorithm

To jointly find n -mode filters $\{\mathbf{H}_n, n = 1, 2, 3\}$ that minimize (6), an Alternating Least Square (ALS) algorithm [13] is necessary. Owing to this procedure, any filter along a given mode depends on the filters along all other

modes. The steps of this algorithm can be summarized as presented here.

1. **Input:** Data tensor \mathcal{R}
2. **Initialization** $k = 0$:
 $\mathcal{X}^0 = \mathcal{R} \iff \mathbf{H}_n = \mathbf{I}_{I_n} \forall n = 1, 2, 3$. Where \mathbf{I}_{I_n} is the $I_n \times I_n$ identity matrix.
3. **ALS loop:** Repeat until convergence, that is, for example, while $\|\mathcal{X}^{k+1} - \mathcal{X}^k\| > \varepsilon$
 - (a) Estimation of $K_n, n = 1, 2, 3$,
 $K_n = \arg \min_{k_n} [\text{AIC}(k_n)], k_n = 1, \dots, I_n - 1$.
 - (b) Estimation of \mathbf{H}_n^{k+1} for $n = 1, 2, 3$.
 - (i) $\mathcal{X}_n^k = \mathcal{R} \times_p \mathbf{H}_p^{k+1} \times_q \mathbf{H}_q^k$,
 $p, q = 1, 2, 3, p, q \neq n$ and $p < q$
 - (ii) $\mathbf{H}_n^{k+1} = \arg \min_{\mathbf{Z}_n} \|\mathcal{X} - \mathcal{X}_n^k \times_n \mathbf{Z}_n\|^2$
 subject to $\mathbf{Z}_n \in \mathbb{R}^{I_n \times I_n}$.
 - (c) Multidimensional Wiener filtering
 $\mathcal{X}^{k+1} = \mathcal{R} \times_1 \mathbf{H}_1^{k+1} \times_2 \mathbf{H}_2^{k+1} \times_3 \mathbf{H}_3^{k+1}$.
 - (d) $k \leftarrow k + 1$.
4. **Output:** Estimated signal tensor
 $\hat{\mathcal{X}} = \mathcal{R} \times_1 \mathbf{H}_1^{k_c} \times_2 \mathbf{H}_2^{k_c} \times_3 \mathbf{H}_3^{k_c}$, where k_c is the convergence iteration index.

As the calculation of n -mode filter \mathbf{H}_n in step 33b utilizes the filters in other modes $\{\mathbf{H}_i, 1 \leq i \leq 3 \text{ and } i \neq n\}$, it shows that the MWF considers the relationships between elements in all modes of the data set.

1.5 PARAFAC filtering

1.5.1 Denoising model

Since the decomposition by TUCKER3 model is not unique and needs to estimate the rank K_n in each mode, another tensor decomposition model PARAFAC was recently introduced to reduce noise in \mathcal{R} [14]. Different from TUCKER3 model, PARAFAC model can decompose a tensor \mathcal{R} uniquely into a sum of rank-one tensors:

$$\hat{\mathcal{R}} = \sum_{k=1}^K \mathbf{a}_1^k \circ \mathbf{a}_2^k \circ \mathbf{a}_3^k + \hat{\mathcal{N}}, \quad (15)$$

where $\hat{\mathcal{N}}$ is the decomposition error. Under the assumption that signal can be expressed by finite rank PARAFAC factorization, the estimate $\hat{\mathcal{X}}$ of the desired signal \mathcal{X} can be expressed by the PARAFAC model:

$$\hat{\mathcal{X}} = \sum_{k=1}^K \mathbf{a}_1^k \circ \mathbf{a}_2^k \circ \mathbf{a}_3^k = \mathcal{I} \times_1 \mathbf{A}_1 \times_2 \mathbf{A}_2 \times_3 \mathbf{A}_3, \quad (16)$$

where \mathcal{I} is a identity tensor, and $\mathbf{A}_n = [\mathbf{a}_n^1, \dots, \mathbf{a}_n^K]$, $n = 1, 2, 3$. In order to obtain the optimal \mathbf{A}_n , the error between $\hat{\mathcal{X}}$ and \mathcal{R} should be minimized:

$$e(\mathbf{A}_1, \mathbf{A}_2, \mathbf{A}_3) = \|\mathcal{R} - \mathcal{I} \times_1 \mathbf{A}_1 \times_2 \mathbf{A}_2 \times_3 \mathbf{A}_3\|^2. \quad (17)$$

Nonetheless, it is worth noting that the criterion of PARAFAC is the squared error between the estimate $\hat{\mathcal{X}}$ and the noisy HSI \mathcal{R} , while that of MWF is the mean squared error between the estimate $\hat{\mathcal{X}}$ and the desired signal \mathcal{X} (see (6)). For a given rank K , minimizing (17) means removing as little signal as possible in the denoising process.

1.5.2 Calculation of \mathbf{A}_n

To obtain \mathbf{A}_n in each mode, the error given in (17) should be minimized:

$$\mathbf{A}_n = \arg \min(e) = \arg \min(\|\hat{\mathbf{X}}_n - \mathbf{R}_n\|^2), \quad (18)$$

where $\hat{\mathbf{X}}_n$ is the n -mode unfolding matrix of $\hat{\mathcal{X}}$ in (16):

$$\hat{\mathbf{X}}_n = \mathbf{A}_n (\mathbf{A}_p \odot \mathbf{A}_q)^T, \quad (19)$$

where $p, q = 1, 2, 3$ and $p \neq q \neq n$. By substituting (19) into (18), we can obtain

$$\mathbf{A}_n = \arg \min(\|\mathbf{A}_n (\mathbf{A}_p \odot \mathbf{A}_q)^T - \mathbf{R}_n\|^2). \quad (20)$$

Obviously, the estimation of \mathbf{A}_n needs information of \mathbf{A}_p and \mathbf{A}_q , which are not known. In this situation, an ALS algorithm should be employed to calculate the optimal \mathbf{A}_n .

1.5.3 PARAFAC ALS algorithm

To jointly estimate \mathbf{A}_n , a 'PARAFAC ALS' algorithm is introduced and its steps are listed as follows:

1. **Input:**
Data tensor \mathcal{R} .
2. **Initialization:**
Set $k = 0$ and $e_k = 0$. Randomly initialize $\mathbf{A}_n^0 \in \mathbb{R}^{I_n \times K}, n = 1, 2, 3$.
3. **Loop:**
 - (a) Estimate \mathbf{A}_n^{k+1}
 - (i) $\mathbf{U}_1^{k+1} = \mathbf{A}_3^k \odot \mathbf{A}_2^k$
 $\mathbf{A}_1^{k+1} = \mathbf{X}_1 \mathbf{U}_1^{k+1} (\mathbf{U}_1^{k+1})^T \mathbf{U}_1^{k+1}$
 - (ii) $\mathbf{U}_2^{k+1} = \mathbf{A}_3^k \odot \mathbf{A}_1^k$
 $\mathbf{A}_2^{k+1} = \mathbf{X}_2 \mathbf{U}_2^{k+1} (\mathbf{U}_2^{k+1})^T \mathbf{U}_2^{k+1}$
 - (iii) $\mathbf{U}_3^{k+1} = \mathbf{A}_2^k \odot \mathbf{A}_1^k$
 $\mathbf{A}_3^{k+1} = \mathbf{X}_3 \mathbf{U}_3^{k+1} (\mathbf{U}_3^{k+1})^T \mathbf{U}_3^{k+1}$
 - (b) Compute $\hat{\mathbf{X}}_3^{k+1} = \mathbf{A}_3^{k+1} \mathbf{U}_3^{k+1}{}^T$
 - (c) $e_{k+1} = \|\mathbf{R}_3 - \hat{\mathbf{X}}_3^{k+1}\|^2$, if $|e_{k+1} - e_k| > \epsilon$ and k is less than the maximum number of iteration, $k \leftarrow k + 1$ and then go back to step 33a. Otherwise, break the loop.
4. **Output:**
Return $\mathbf{A}_n = \mathbf{A}_n^{k+1}, n = 1, 2, 3$.

1.5.4 Rank estimation

As described in Section 1.5.1, PARAFAC filtering is an algorithm which minimizes $e(\mathbf{A}_1, \mathbf{A}_2, \mathbf{A}_3)$ in (17) under a given rank K . In other words, it is assumed that the rank K is known in PARAFAC filtering. Unfortunately, the rank K is generally unknown in practice; therefore, an algorithm used to estimate K is presented in this section. The details are as follows:

1. **Input:**
Data tensor \mathcal{R}
2. **Initialization:**
Set $i = 1$. Set rank-searching-set K-SCOPE.
3. **Loop:**
 - (a) Set $K = \text{K-SCOPE}[i]$.
 - (b) Do PARAFAC decomposition:
 $\mathcal{R} = \sum_{k=1}^K \mathbf{a}_1^k \circ \mathbf{a}_2^k \circ \mathbf{a}_3^k + \hat{\mathcal{N}}$.
 - (c) At $n=1,2,3$, calculate the covariance matrix \mathbf{C}_n of $\hat{\mathbf{N}}_n$, the n -mode unfolding matrix of $\hat{\mathcal{N}}$.
 - (d) If
 - (i) $\sigma_{\text{diag}}^2 = 1/I_n \sum_{i=1}^{I_n} (c_{i,i} - 1/I_n \sum_{i=1}^{I_n} c_{i,i})^2 < \delta_1$, where $c_{i,i}$ is the diagonal elements of \mathbf{C}_n .
 - (ii) $|\|\mathbf{C}_n\|^2 - \sum_{i=1}^{I_n} c_{i,i}^2| < \delta_2$
 these two conditions are satisfied for all $n = 1, 2, 3$ at the same time, break the loop. Otherwise, $i \leftarrow i + 1$.
4. **Output:**
Return the rank K .

1.6 MWPT-MWF

1.6.1 Denoising model

MWF and PARAFAC treat \mathcal{R} as an entire entity in the denoising process. This works well when there are only abundant signals or the rare signals can be neglected. However, in the situation where the rare signals cannot be neglected, such as the target detection, MWF and PARAFAC might remove rare signals in the denoising process.

MWPT-MWF has been proposed to preserve rare signals in the denoising process and hence improve the denoising performance. In MWPT-MWF, \mathcal{X} is estimated by minimizing MSE between the desired signal \mathcal{X} and its estimate $\hat{\mathcal{X}}$:

$$\text{MSE} = \text{E} \left[\|\hat{\mathcal{X}} - \mathcal{X}\|^2 \right]. \quad (21)$$

Nevertheless, unlike MWF or PARAFAC, MWPT-MWF reduces noise by jointly filtering the wavelet packet

coefficient set. The details of MWPT-MWF will be described in the following subsections.

1.6.2 Multidimensional wavelet packet transform

The multidimensional wavelet packet transform (MWPT) can be written in tensor form as:

$$\mathcal{C}^R = \mathcal{R} \times_1 \mathbf{W}_1 \times_2 \mathbf{W}_2 \times_3 \mathbf{W}_3 \quad (22)$$

and the reconstruction can be written as:

$$\mathcal{R} = \mathcal{C}^R \times_1 \mathbf{W}_1^T \times_2 \mathbf{W}_2^T \times_3 \mathbf{W}_3^T, \quad (23)$$

where $\mathbf{W}_n \in \mathbb{R}^{I_n \times I_n}$, $n = 1, 2, 3$ indicate the wavelet packet transform matrices. When the transform level vector is $\mathbf{l} = [l_1, l_2, l_3]^T$, where $l_n \geq 0$ denotes the wavelet packet transform level in mode n , the coefficient tensor $\mathcal{C}_{\mathbf{l}, \mathbf{m}}^R$, which is also called a component in this paper, of scale $\mathbf{m} = [m_1, m_2, m_3]$, where $0 \leq m_n \leq 2^{l_n} - 1$, can be extracted by

$$\mathcal{C}_{\mathbf{l}, \mathbf{m}}^R = \mathcal{C}^R \times_1 \mathbf{E}_{m_1} \times_2 \mathbf{E}_{m_2} \times_3 \mathbf{E}_{m_3} \quad (24)$$

and the corresponding inverse process is

$$\mathcal{C}^R = \sum_{m_1} \sum_{m_2} \sum_{m_3} \mathcal{C}_{\mathbf{l}, \mathbf{m}}^R \times_1 \mathbf{E}_{m_1}^T \times_2 \mathbf{E}_{m_2}^T \times_3 \mathbf{E}_{m_3}^T, \quad (25)$$

where the extraction operator \mathbf{E}_{m_n} is defined as

$$\mathbf{E}_{m_n} = [\mathbf{0}_1, \mathbf{I}_{\frac{I_n}{2^{l_n}} \times \frac{I_n}{2^{l_n}}}, \mathbf{0}_2] \in \mathbb{R}^{I_n/2^{l_n} \times I_n}, \quad (26)$$

where $\mathbf{0}_1$ is a zero matrix with size $\frac{I_n}{2^{l_n}} \times \frac{m_n I_n}{2^{l_n}}$ and $\mathbf{0}_2$ is a zero matrix with size $\frac{I_n}{2^{l_n}} \times \frac{(2^{l_n} - 1 - m_n) I_n}{2^{l_n}}$.

1.6.3 Multiway Wiener filter in multidimensional wavelet packet domain

After the MWPT, abundant and rare signals can be separated into different components; therefore, the signal subspace of each component can be estimated more accurately than that of the entire dataset. Furthermore, a better estimation of the signal subspace can improve the performance of MWF in each component. However, the denoising criterion of MWPT-MWF is the minimization of the MSE between \mathcal{X} and $\hat{\mathcal{X}}$ but not the component and its estimate; therefore, this subsection proves the ability of MWPT-MWF in minimizing the MSE between \mathcal{X} and $\hat{\mathcal{X}}$, which is defined in (21). By performing MWPT to tensor \mathcal{R} , \mathcal{X} and \mathcal{N} in expression (4), we obtain

$$\begin{aligned} & \mathcal{R} \times_1 \mathbf{W}_1 \times_2 \mathbf{W}_2 \times_3 \mathbf{W}_3 \\ &= (\mathcal{X} + \mathcal{N}) \times_1 \mathbf{W}_1 \times_2 \mathbf{W}_2 \times_3 \mathbf{W}_3 \\ &= \mathcal{X} \times_1 \mathbf{W}_1 \times_2 \mathbf{W}_2 \times_3 \mathbf{W}_3 + \mathcal{N} \times_1 \mathbf{W}_1 \times_2 \mathbf{W}_2 \times_3 \mathbf{W}_3. \end{aligned} \quad (27)$$

The coefficient tensor of each part:

$$C_1^{\mathcal{R}} = \mathcal{R} \times_1 \mathbf{W}_1 \times_2 \mathbf{W}_2 \times_3 \mathbf{W}_3 \quad (28)$$

$$C_1^{\mathcal{X}} = \mathcal{X} \times_1 \mathbf{W}_1 \times_2 \mathbf{W}_2 \times_3 \mathbf{W}_3 \quad (29)$$

$$C_1^{\mathcal{N}} = \mathcal{N} \times_1 \mathbf{W}_1 \times_2 \mathbf{W}_2 \times_3 \mathbf{W}_3 \quad (30)$$

and the coefficient tensor of the estimate $\hat{\mathcal{X}}$:

$$\hat{C}_1^{\mathcal{X}} = \hat{\mathcal{X}} \times_1 \mathbf{W}_1 \times_2 \mathbf{W}_2 \times_3 \mathbf{W}_3. \quad (31)$$

Extracting the components of each frequency $C_{1,\mathbf{m}}^{\mathcal{R}}$, $C_{1,\mathbf{m}}^{\mathcal{X}}$ and $C_{1,\mathbf{m}}^{\mathcal{N}}$ from $C_1^{\mathcal{R}}$, $C_1^{\mathcal{X}}$ and $C_1^{\mathcal{N}}$ respectively by using (24), we obtain

$$C_{1,\mathbf{m}}^{\mathcal{R}} = C_{1,\mathbf{m}}^{\mathcal{X}} + C_{1,\mathbf{m}}^{\mathcal{N}}. \quad (32)$$

From Parseval's theorem, the following expression can be obtained:

$$\|\mathcal{X} - \hat{\mathcal{X}}\|^2 = \|C_1^{\mathcal{X}} - \hat{C}_1^{\mathcal{X}}\|^2 = \sum_{\mathbf{m}} \|C_{1,\mathbf{m}}^{\mathcal{X}} - \hat{C}_{1,\mathbf{m}}^{\mathcal{X}}\|^2 \quad (33)$$

which means that minimizing the MSE between \mathcal{X} and its estimate $\hat{\mathcal{X}}$ is equivalent to minimizing the MSE between $C_{1,\mathbf{m}}^{\mathcal{X}}$ and $\hat{C}_{1,\mathbf{m}}^{\mathcal{X}}$ for each \mathbf{m} . If $\hat{C}_{1,\mathbf{m}}^{\mathcal{X}}$ is estimated by Tucker3 decomposition of $C_{1,\mathbf{m}}^{\mathcal{R}}$:

$$\hat{C}_{1,\mathbf{m}}^{\mathcal{X}} = C_{1,\mathbf{m}}^{\mathcal{R}} \times_1 \mathbf{H}_{1,\mathbf{m}} \times_2 \mathbf{H}_{2,\mathbf{m}} \times_3 \mathbf{H}_{3,\mathbf{m}} \quad (34)$$

then $\mathbf{H}_{1,\mathbf{m}}$, $\mathbf{H}_{2,\mathbf{m}}$, $\mathbf{H}_{3,\mathbf{m}}$ are the n -mode filters of the multiway Wiener filter aforementioned in Section 1.4. After estimating $\hat{C}_{1,\mathbf{m}}^{\mathcal{X}}$ for each \mathbf{m} , we obtain $\hat{\mathcal{X}}$ by concatenating $\hat{C}_{1,\mathbf{m}}^{\mathcal{X}}$. Furthermore, the estimate $\hat{\mathcal{X}}$ can be obtained by inverse MWPT:

$$\hat{\mathcal{X}} = \hat{C}_1^{\mathcal{X}} \times_1 \mathbf{W}_1^T \times_2 \mathbf{W}_2^T \times_3 \mathbf{W}_3^T \quad (35)$$

1.6.4 Best transform level and basis selection

In MWPT-MWF, several parameters should be determined, as presented here.

1. Level of transform: the performance of the algorithm is affected by the level of transform, which depends on the size of tensor \mathcal{R} . The maximum level can be calculated by

$$N_{L_k} = \lceil \log_2 I_k \rceil - 5, \quad k = 1, 2, 3, \quad (36)$$

where $\lceil \cdot \rceil$ rounds a number upward to its nearest integer, and the constant 5 is subtracted from $\lceil \log_2 I_k \rceil$ to make sure there are enough elements in each mode so that the transform is meaningful. Then, the set of possible transform levels can be expressed as:

$$L_k = \{0, 1, \dots, N_{L_k}\}, \quad k = 1, 2, 3, \quad (37)$$

where $\{\cdot\}$ denotes a set.

2. Basis of transform: there are many wavelet bases designed for different cases. For the simplicity of expression, we define

$$W = \{w_1, w_2, \dots, w_{N_w}\} \quad (38)$$

to denote the set of possible wavelet bases, where N_w is the number of wavelets in this set.

The best transform level and basis should minimize the MSE or risk $R_c(\mathcal{X}, \hat{\mathcal{X}}) = E[\|\mathcal{X} - \hat{\mathcal{X}}\|^2]$ [36], whose equivalent form for each component can be expressed as:

$$R_c(\mathcal{X}, \hat{\mathcal{X}}) = \sum_{\mathbf{m}} E[\|C_{1,\mathbf{m}}^{\mathcal{X}} - \hat{C}_{1,\mathbf{m}}^{\mathcal{X}}\|^2] \quad (39)$$

Then, the best transform level and basis can be selected by

$$\mathbf{l}, w = \arg \min_{l_k \in L_k, w \in W} \sum_{\mathbf{m}} E[\|C_{1,\mathbf{m}}^{\mathcal{X}} - \hat{C}_{1,\mathbf{m}}^{\mathcal{X}}\|^2], \quad k = 1, 2, 3. \quad (40)$$

While selecting the optimal \mathbf{l}, w depends on \mathcal{X} which is generally unknown. To overcome this drawback, an alternative solution should be found. Denoting by $\hat{C}_{1,\mathbf{m}}^{\mathcal{X}}[d]$ the estimate of $C_{1,\mathbf{m}}^{\mathcal{X}}$ at the d -th ALS loop aforementioned in Section 1.4.4 and noticing that, when $\|\hat{C}_{1,\mathbf{m}}^{\mathcal{X}}[d] - \hat{C}_{1,\mathbf{m}}^{\mathcal{X}}[d-1]\|^2$ is minimized, $\hat{C}_{1,\mathbf{m}}^{\mathcal{X}} \triangleq \hat{C}_{1,\mathbf{m}}^{\mathcal{X}}[d]$ is the optimal estimate of $C_{1,\mathbf{m}}^{\mathcal{X}}$ obtained by MWF, and at the same time, $E[\|C_{1,\mathbf{m}}^{\mathcal{X}} - \hat{C}_{1,\mathbf{m}}^{\mathcal{X}}\|^2]$ is minimized according to Section 1.4.2. Therefore, (40) can be replaced by

$$\mathbf{l}, w = \arg \min_{l_k \in L_k, w \in W} \hat{R}_c, \quad k = 1, 2, 3, \quad (41)$$

where

$$\hat{R}_c = \sum_{\mathbf{m}} \|\hat{C}_{1,\mathbf{m}}^{\mathcal{X}}[d] - \hat{C}_{1,\mathbf{m}}^{\mathcal{X}}[d-1]\|^2. \quad (42)$$

1.6.5 Summary of the MWPT-MWF

The proposed algorithm, which is denoted by MWPT-MWF, can be summarized as presented here.

1. **Input:**
Data tensor \mathcal{R} .
2. **Initialization:**
Set $L = \{1, \dots, N_{L_k}\}$, $W = \{w_1, \dots, w_{N_w}\}$ and the risk threshold ϵ .
3. **Loop:**
For each $l_1, l_2, l_3 \in L$ and $w \in W$. Loop l_1, l_2, l_3 and w :
 - (a) Decompose the whitened data \mathcal{R} by MWPT:
 $C_1^{\mathcal{R}} = \mathcal{R} \times_1 \mathbf{W}_1 \times_2 \mathbf{W}_2 \times_3 \mathbf{W}_3$.
 - (b) Extract component $C_{1,\mathbf{m}}^{\mathcal{R}}$ from $C_1^{\mathcal{R}}$ by (24), for $\mathbf{m} = [m_1, m_2, m_3]^T$, where $0 \leq m_k \leq 2^{l_k} - 1$, $k = 1, 2, 3$.
 - (c) Filter component $C_{1,\mathbf{m}}^{\mathcal{R}}$ by MWF:
 $\hat{C}_{1,\mathbf{m}}^{\mathcal{X}} = C_{1,\mathbf{m}}^{\mathcal{R}} \times_1 \mathbf{H}_{1,\mathbf{m}} \times_2 \mathbf{H}_{2,\mathbf{m}} \times_3 \mathbf{H}_{3,\mathbf{m}}$.

- (d) Calculate the risk $\hat{R}_c = \sum_m \|\hat{C}_{1,m}^{\mathcal{X}}[d] - \hat{C}_{1,m}^{\mathcal{X}}[d-1]\|^2$. If \hat{R}_c reaches a fixed threshold ϵ , return the optimal l_1, l_2, l_3, w and $\hat{C}_{1,m}^{\mathcal{X}}$.

4. **Output:** Concatenate $\hat{C}_{1,m}^{\mathcal{X}}$ to obtain $C_1^{\mathcal{X}}$ and perform inverse MWPT: $\hat{\mathcal{X}} = \hat{C}_1^{\mathcal{X}} \times_1 \mathbf{W}1^T \times_2 \mathbf{W}2^T \times_3 \mathbf{W}3^T$.

1.7 Experimental results

In this section, we use a real-world high spatial resolution image acquired by HYperspectral Digital Imagery Collection Experiment (HYDICE). The HYDICE image contains 65 rows, 100 columns, and 160 spectral bands, and is modeled as a $65 \times 100 \times 160$ tensor in this paper. Six targets of interest are selected in the image as shown in the ground-truth map in Figure 1 and the corresponding mask is shown in Figure 2. The spectral signatures of these six targets are presented in Figure 3. These six targets are chosen because they have different spectral signatures and sizes, so that the denoising and target detection performance on different target sizes can be evaluated.

White Gaussian noise is added into the HSI with signal-to-noise ratio (SNR) ranged from 15 to 30 dB (with a step of 5 dB) to reproduce different simulation scenarios. MWF, PARAFAC, and MWPT-MWF are used to reduce noise in the HSI. The rank-searching-set of PARAFAC is set as [51,101,151,201], and wavelet db3 is selected to do MWPT with transform levels $[l_1, l_2, l_3] = [1, 1, 0]$.

1.7.1 Denoising performance evaluation and comparison

To present the denoising results intuitively, Figure 4 shows the target spectral signatures of the noisy HSI and the HSI denoised by MWF, PARAFAC, and MWPT-MWF, respectively. By comparing the four sub-figures in Figure 4, it is evident that denoising is a necessary procedure to restore the target spectral signatures. Moreover, we can see that there still exists more noise in Figure 4b than Figure 4c and Figure 4d. Especially, the spectral signatures of targets 1,

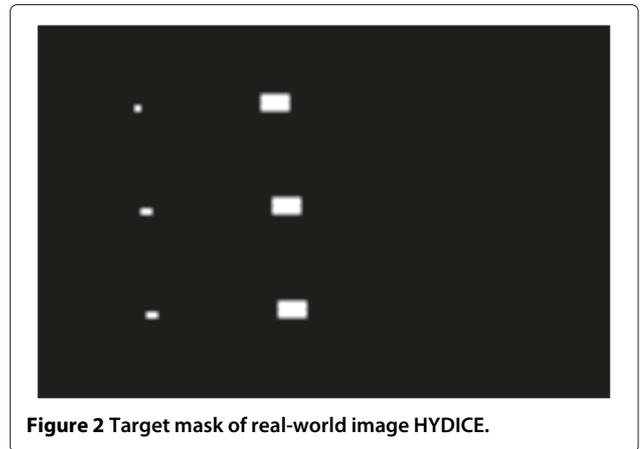


Figure 2 Target mask of real-world image HYDICE.

3, and 5 are almost mixed together in Figure 4b. Figure 4c and Figure 4d are much better, at least the residual noise is small after denoising. However, the spectral signatures are changed more greatly after denoising by PARAFAC than by MWPT-MWF, which can be seen obviously from the signatures of targets 5 and 6. In Figure 4c, the signatures of targets 5 and 1 are almost overlapped, while in Figure 4d, these two signatures can be distinguished easily.

To compare the performances of MWF, PARAFAC, and MWPT-MWF, the SNR of the image after denoising, also named as SNR output, is defined as below [12]:

$$\text{SNR}_{\text{OUTPUT}} = 10 \log\left(\frac{\|\mathcal{X}\|^2}{\|\hat{\mathcal{X}} - \mathcal{X}\|^2}\right). \quad (43)$$

If $\text{SNR}_{\text{OUTPUT}}$ is greater than $\text{SNR}_{\text{INPUT}}$, we can conclude that the algorithm improves the SNR of the image.

The $\text{SNR}_{\text{OUTPUT}}$ values obtained when $\text{SNR}_{\text{INPUT}}$ is varying from 15 to 30 dB by different denoising methods are shown in Table 1. It is obvious that MWPT-MWF outperforms the other two denoising methods significantly. When the $\text{SNR}_{\text{INPUT}}$ is low, from 15 to 25 dB in Table 1, the denoising result of PARAFAC is better than that of

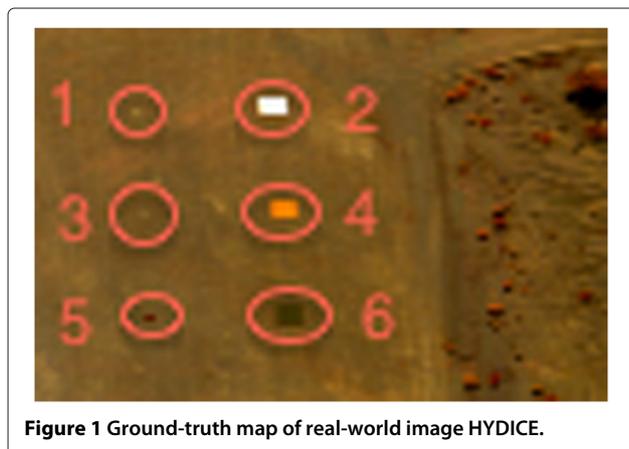


Figure 1 Ground-truth map of real-world image HYDICE.

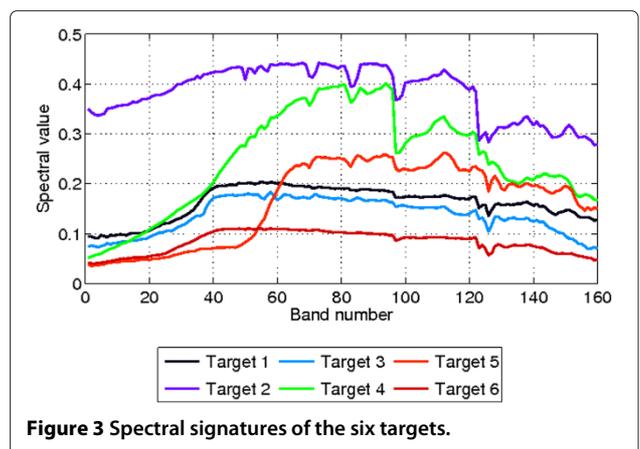


Figure 3 Spectral signatures of the six targets.

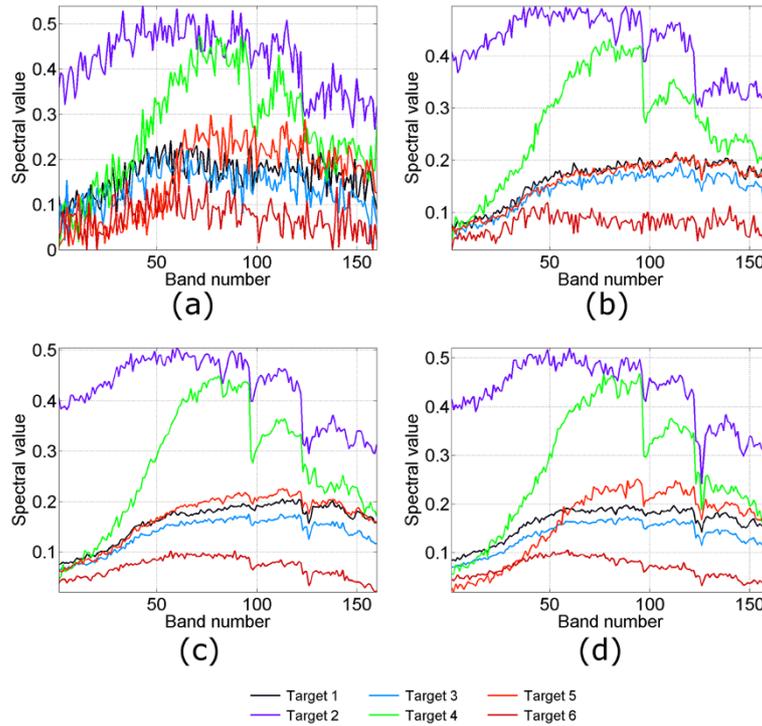


Figure 4 Comparison of target spectra in (a) the noisy HSI and HSI denoised by (b) MWF, (c) PARAFAC, and (d) MWPT-MWF, $SNR_{INPUT} = 15dB$.

MWF. But when the SNR_{INPUT} is high, from 25 to 30 dB, the performance of MWF is slightly better than that of PARAFAC. Moreover, it is worth noting that all of these three methods can improve the SNR significantly. When the SNR_{INPUT} is 15 dB, the SNR_{OUTPUT} after denoising is 30 dB maximum by MWPT-MWF and 24 dB minimum by MWF. The denoising results shown in Table 1 give the experimental evidence of the benefits derived from the denoising procedure.

1.7.2 Target detection performance evaluation and comparison

In the last subsection, we have compared the denoising performances of different methods in the aspect of SNR_{OUTPUT} . However, sometimes SNR_{OUTPUT} cannot reflect the denoising performance we want, especially

when we consider preserving small targets in the HSI while removing noise. Hence, in this subsection, we compare the target detection performance after denoising by MWF, PARAFAC, and MWPT-MWF.

Spectral Angle Mapper (SAM) detector [37] is used in the experiment to detect targets in the HSI. As SAM does not require the characterization of the background, it can avoid the inaccuracy of the comparison result caused by the noise covariance matrix estimation error. The SAM detector can be expressed as

$$T_{SAM}(\mathbf{x}) = \frac{\mathbf{s}^T \mathbf{x}}{(\mathbf{s}^T \mathbf{s})^{1/2} (\mathbf{x}^T \mathbf{x})^{1/2}}, \quad (44)$$

where \mathbf{s} is the reference spectrum and \mathbf{x} is the pixel spectrum.

To assess the performances of detection, the probability of detection (P_d) is defined as

$$P_d = \frac{\sum_i^{n_s} N_i^{rd}}{\sum_i^{n_s} N_i} \quad (45)$$

and the probability of false alarm (P_{fa}) is defined as

$$P_{fa} = \frac{\sum_i^{n_s} N_i^{fd}}{\sum_i^{n_s} (I_1 \times I_2 - N_i)}, \quad (46)$$

Table 1 SNR_{OUTPUT} vs. SNR_{INPUT} obtained after denoising by methods MWF, PARAFAC, and MWPT-MWF

$SNR_{INPUT}(dB)$	$SNR_{OUTPUT}(dB)$		
	MWF	PARAFAC	MWPT-MWF
15	23.55	28.80	30.27
20	29.68	32.04	33.58
25	35.54	35.19	36.60
30	38.35	38.07	39.19

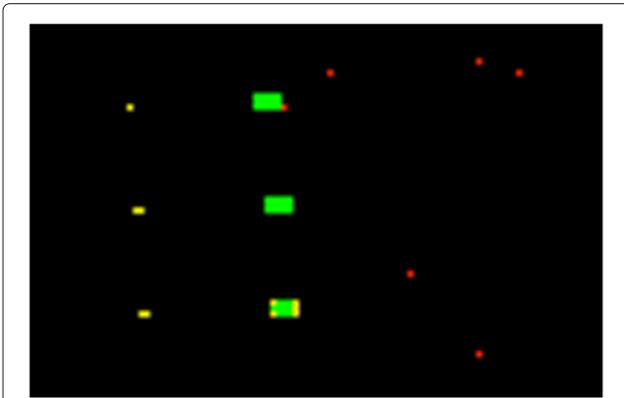


Figure 5 Detection result obtained after denoising by MWF with parameters $Pfa=10^{-4}$ and $SNR_{INPUT} = 15$ dB.

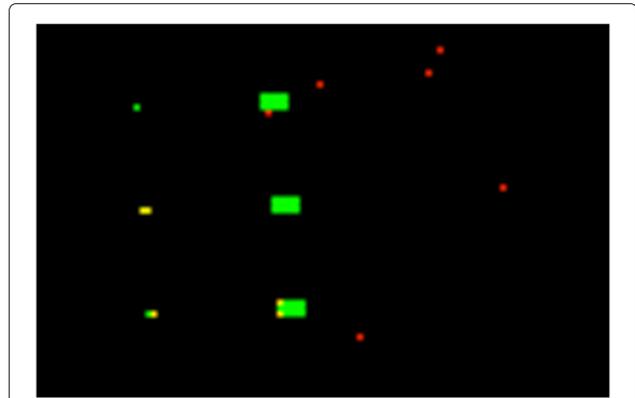


Figure 7 Detection result obtained after denoising by MWPT-MWF with parameters $Pfa=10^{-4}$ and $SNR_{INPUT} = 15$ dB.

where n_s is the number of spectral signatures, N_i the number of pixels with spectral signature i , N_i^{fd} the number of correctly detected pixels, and N_i^{fd} the number of false-alarm pixels.

Figures 5, 6, and 7 are the target detection results after denoising by MWF, PARAFAC, and MWPT-MWF, respectively, in the noise environment $SNR_{INPUT} = 15$ dB. In the images, the black pixel indicates no-target, the green the correct-detection, the red the false alarm, and the blue the missed target. From Figure 5, which shows the detection result after denoising by MWF, we can see that all of the three small targets (targets 1, 3, and 5) are missed in the detection. Moreover, most of the pixels of target 5 are also missed. The detection result after denoising by PARAFAC, in Figure 6, is slightly better than that by MWF, but all of the small targets are also lost in the detection. MWPT-MWF shows its capability of preserving small targets in Figure 7, in which two of the three small targets are detected correctly. Apart from preserving small targets, MWPT-MWF can also improve

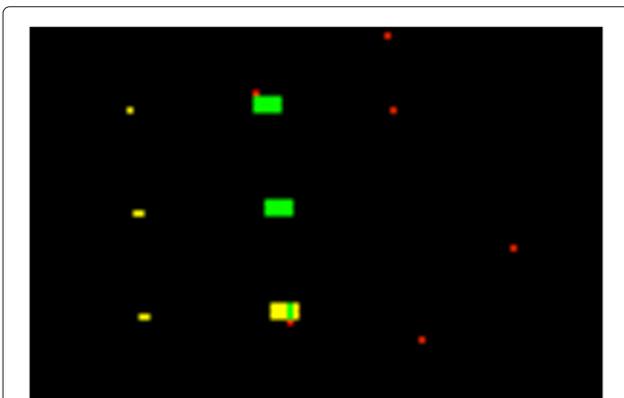


Figure 6 Detection result obtained after denoising by PARAFAC with parameters $Pfa=10^{-4}$ and $SNR_{INPUT} = 15$ dB.

the detection performance of the large-size-small-energy target 6, which is obvious by comparing Figures 5, 6, and 7.

To evaluate the detection performance in different noise environments, Table 2 shows the Pd values versus SNR_{INPUT} of different denoising methods with SNR_{INPUT} ranged from 15 to 30 dB.

It is obvious that the detection result after denoising by MWPT-MWF outperforms the two other methods. By comparing Table 2 with Table 1, we can understand that the denoising process can improve the target detection performance.

2 Conclusion

In this paper, a survey has been presented on three recently proposed tensor filtering methods: MWF, PARAFAC, and MWPT-MWF. They utilize multilinear algebra in analyzing a multidimensional data cube to jointly filter it in each mode.

The MWF extends the classical Wiener filter to the multidimensional case by using the TUCKER3 decomposition while minimizing the MSE between the desired signal tensor and the estimated signal tensor. As the filter in one mode relies on the filters in the other modes, the ALS algorithm is used to jointly calculate the MWF filters. In the filtering process, the signal subspace rank in mode n needs to be known to remove the noise in the orthogonal

Table 2 SNR_{INPUT} vs. Pd obtained after denoising by methods MWF, PARAFAC, and MWPT-MWF

SNR_{INPUT} (dB)	Pd		
	MWF	PARAFAC	MWPT-MWF
15	0.724	0.878	0.922
20	0.972	0.998	0.998
25	1	1	1
30	1	1	1

complement subspace of the signal subspace. For this reason, the AIC algorithm is taken to estimate the rank in mode n , which implies that the MWF can reduce noise automatically.

The PARAFAC filtering method was proposed to reduce the number of rank values to be estimated. As aforementioned, the rank in each mode must be estimated in MWF, while only one rank must be estimated in PARAFAC filtering. Moreover, the low-rank PARAFAC decomposition is unique for rank values higher than one, whereas the TUCKER3 decomposition is not. However, there is not an efficient way to estimate the PARAFAC rank automatically. Though we have shown a rank estimation method in this paper, it is a time-consuming brute force searching way.

The MWF and PARAFAC were proposed to process the HSI as a whole entity, but this may remove the small targets in an HSI in the denoising process. Distinguishing from MWF and PARAFAC, MWPT-MWF firstly transforms the HSI into different wavelet packet sets, also called components in this paper, and then filters each component as a whole entity. As the small targets are separated from the large ones, the former can be well preserved in the denoising process.

A real-world HYDICE HSI is used in the comparative study. Quantitative and visual evaluation of the three methods is shown. From the experimental results, we can conclude that MWPT-MWF is a suitable tool for denoising especially when there exist small targets in the HSI.

Competing interests

The authors declare that they have no competing interests.

Acknowledgements

The authors would like to thank the reviewers for their careful reading and helpful comments which improve the quality of this paper.

Received: 16 August 2013 Accepted: 2 December 2013

Published: 17 December 2013

References

1. K Kotwal, S Chaudhuri, Visualization of hyperspectral images using bilateral filtering. *IEEE Trans. Geosci. Remote Sens.* **48**(5), 2308–2316 (2010)
2. S Lewis, A Hudak, R Ottmar, P Robichaud, L Lentile, S Hood, J Cronan, P Morgan, Using hyperspectral imagery to estimate forest floor consumption from wildfire in boreal forests of Alaska, USA. *Int. J. Wildland Fire* **20**(2), 255–271 (2011)
3. K Tiwari, M Arora, D Singh, An assessment of independent component analysis for detection of military targets from hyperspectral images. *Int. J. Appl. Earth Obs. Geoinf.* **13**(5), 730–740 (2011)
4. T Veracini, S Matteoli, M Diani, G Corsini, Nonparametric framework for detecting spectral anomalies in hyperspectral images. *IEEE Geosci. Remote Sens. Lett.* **8**(4), 666–670 (2011)
5. S Prasad, W Li, JE Fowler, LM Bruce, Information fusion in the redundant-wavelet-transform domain for noise-robust hyperspectral classification. *IEEE Trans. Geosci. Remote Sens.* **50**(9), 3474–3486 (2012)
6. J Kerekes, J Baum, Full-spectrum spectral imaging system analytical model. *IEEE Trans. Geosci. Remote Sens.* **43**(3), 571–580 (2005)
7. ML Uss, B Vozel, W Lukin, K Chehdi, Local signal-dependent noise variance estimation from hyperspectral textural images. *IEEE J. Sel. Topics Signal Process.* **5**(3), 469–486 (2011)
8. N Acito, M Diani, G Corsini, Subspace-based striping noise reduction in hyperspectral images. *IEEE Trans. Geosci. Remote Sens.* **49**(4), 1325–1342 (2011)
9. L Shao, R Yan, X Li, Y Liu, From heuristic optimization to dictionary learning: a review and comprehensive comparison of image denoising algorithms. *IEEE Trans. Cybernet.* (2013) in press.
10. R Yan, L Shao, Y Liu, Nonlocal hierarchical dictionary learning using wavelets for image denoising. *IEEE Trans. Image Process.* **22**(12), 4689–4698 (2013)
11. R Yan, L Shao, S Cvetković, J Klijin, Improved nonlocal means based on pre-classification and invariant block matching. *J. Display Technol.* **8**(4), 212–218 (2012)
12. D Letexier, S Bourennane, Noise removal from hyperspectral images by multidimensional filtering. *IEEE Trans. Geosci. Remote Sens.* **46**(7), 2061–2069 (2008)
13. N Renard, S Bourennane, Improvement of target detection methods by multiway filtering. *IEEE Trans. Geosci. Remote Sens.* **46**(8), 2407–2417 (2008)
14. X Liu, S Bourennane, C Fossati, Denoising of hyperspectral images using the PARAFAC model and statistical performance analysis. *IEEE Trans. Geosci. Remote Sens.* **50**(10), 3717–3724 (2012)
15. JA Richards, *Remote sensing digital image analysis: an introduction* (Springer, Berlin Heidelberg, 2012)
16. IC Chein, D Qian, Estimation of number of spectrally distinct signal sources in hyperspectral imagery. *IEEE Trans. Geosci. Remote Sens.* **42**(3), 608–619 (2004)
17. O Kuybeda, D Malah, M Barzohar, Rank estimation and redundancy reduction of high-dimensional noisy signals with preservation of rare vectors. *IEEE Trans. Signal Process.* **55**(12), 5579–5592 (2007)
18. N Acito, M Diani, G Corsini, A new algorithm for robust estimation of the signal subspace in hyperspectral images in the presence of rare signal components. *IEEE Trans. Geosci. Remote Sens.* **47**(11), 3844–3856 (2009)
19. J Martin-Herrero, Anisotropic diffusion in the hypercube. *IEEE Trans. Geosci. Remote Sens.* **45**(5), 1386–1398 (2007)
20. R Mendez-Rial, M Calvino-Cancela, J Martin-Herrero, Accurate implementation of anisotropic diffusion in the hypercube. *IEEE Geosci. Remote Sens. Lett.* **7**(4), 870–874 (2010)
21. N Le Bihan, G Ginolhac, Three-mode data set analysis using higher order subspace method: application to sonar and seismo-acoustic signal processing. *Signal Process.* **84**(5), 919–942 (2004)
22. MAO Vasilescu, D Terzopoulos, Multilinear image analysis for facial recognition, in *International Association of Pattern Recognition (IAPR)*, vol. 2 (Quebec City, August 2002), pp. 511–514
23. D Muti, S Bourennane, Multidimensional signal processing using lower-rank tensor approximation, in *IEEE ICASSP*, vol. 3 (Hongkong, 6–10 April 2003), pp. 457–60
24. D Muti, S Bourennane, Multidimensional filtering based on a tensor approach. *Signal Process.* **85**(12), 2338–2353 (2005)
25. D Letexier, S Bourennane, J Talon, Nonorthogonal tensor matricization for hyperspectral image filtering. *IEEE Geosci. Remote Sens. Lett.* **5**, 3–7 (2008)
26. RA Harshman, ME Lundy, The PARAFAC model for three-way factor analysis and multidimensional scaling, in *Research methods for multimode data analysis* (Praeger, New York, 1984), pp. 122–215
27. JD Carroll, JJ Chang, Analysis of individual differences in multidimensional scaling via an N-way generalization of Eckart-Young decomposition. *Psychometrika* **35**(3), 283–319 (1970)
28. A Smilde, R Bro, P Geladi, *Multi-way analysis: applications in the chemical sciences* (Wiley, Hoboken, 2005)
29. X Guo, S Miron, D Brie, S Zhu, X Liao, A CANDECOMP/PARAFAC perspective on uniqueness of DOA estimation using a vector sensor array. *IEEE Trans. Signal Process.* **59**(7), 3475–3481 (2011)
30. AL De Almeida, G Favier, JCM Mota, PARAFAC-based unified tensor modeling for wireless communication systems with application to blind multiuser equalization. *Signal Process.* **87**(2), 337–351 (2007)
31. X Liu, S Bourennane, C Fossati, Nonwhite noise reduction in hyperspectral images. *IEEE Geosci. Remote Sens. Lett.* **9**(3), 368–372 (2012)
32. T Lin, S Bourennane, Hyperspectral image processing by jointly filtering wavelet component tensor. *IEEE Trans. Geosci. Remote Sens.* **51**(6), 3529–3541 (2013)

33. TG Kolda, BW Bader, Tensor decompositions and applications. *SIAM Rev.* **51**(3), 455–500 (2009)
34. A Cichocki, R Zdunek, A Phan, S Amari, *Nonnegative matrix and tensor factorizations: applications to exploratory multi-way data analysis and blind source separation* (Wiley, Hoboken, 2009)
35. D Muti, S Bourennane, J Marot, Lower-rank tensor approximation and multiway, filtering. *SIAM J. Matrix Anal. Appl.* **30**(3), 1172–1204 (2008)
36. D Donoho, I Johnstone, Ideal denoising in an orthonormal basis chosen from a library of bases. *Comptes Rendus de l'Academie des Sciences-Serie I-Mathematique* **319**(12), 1317–1322 (1994)
37. X Jin, S Paswaters, H Cline, A comparative study of target detection algorithms for hyperspectral imagery, in *SPIE Defense, Security, and Sensing* (Orlando, FL, 13–17 April 2009)

doi:10.1186/1687-6180-2013-186

Cite this article as: Lin and Bourennane: Survey of hyperspectral image denoising methods based on tensor decompositions. *EURASIP Journal on Advances in Signal Processing* 2013 **2013**:186.

Submit your manuscript to a SpringerOpen[®] journal and benefit from:

- ▶ Convenient online submission
- ▶ Rigorous peer review
- ▶ Immediate publication on acceptance
- ▶ Open access: articles freely available online
- ▶ High visibility within the field
- ▶ Retaining the copyright to your article

Submit your next manuscript at ▶ springeropen.com
