Geoscience and Remote Sensing (2019) Vol. 2: 25-40 **DOI: 10.23977/geors.2019.21002**
Clausius Scientific Press, Canada **ISSN 2523-2592**

Spectral-Spatial Method for Hyperspectral Image classification in Noisy Environment

Pierre Delmas, Caroline Fossati and Salah Bourennane

Aix Marseille Univ, CNRS, Centrale Marseille, Institut Fresnel, Marseille, France

Keywords: Detection, multi-linear algebra, reduction, wavelet, hyperspectral image.

Abstract: Target detection is an important issue in the HyperSpectral Image (HSI) processing field. However, current spectral-identification-based target detection algorithms are sensitive to the noise and most denoising algorithms cannot preserve small targets, therefore it is necessary to design a robust detection algorithm that can preserve small targets. Because the signal-dependent photonic noise has become as dominant as the signal-independent noise generated by the electronic circuitry in HSI data collected by new-generation hyperspectral sensors, the reduction of the additive signal-dependent photonic noise becomes the focus of the current research in this field. To reduce the opto-elctronic noise from HSIs, a new method is developed in this paper. Firstly, a pre-whitening procedure is proposed to whiten noise in HSIs. Secondly, a three-dimensional wavelet packet transform (3-WPT) in tensor form is presented to find different component tensors of HSI. Then, to jointly filter a component tensor in each mode, multiway Wiener filter (MWF) is introduced. Moreover, to determine the best transform level and basis of 3-WPT a risk function is proposed. The effectiveness of our method in classification is experimentally demonstrated on a real-world HSI acquired by airborne sensor.

1. Introduction

Hyperspectral image has been considered with an increasing interest since the 90s in many fields such as medicine [1], forest monitoring [2], etc. In hyperspectral image processing, images are modeled as *three*-dimensional data: two spatial dimensions and one spectral dimension [3, 4, 5, 6]. Most of hyperspectral images, such as images acquired by Hyperspectral Digital Imagery Collection Experiment(HYDICE) and Airborne Visible/Infrared Imaging Spectrometer (AVIRIS) sensors are impaired by noise from solar radiation, atmospheric scattering, interactions between solar radiation and the Earth's surface, and responsivity of the sensing instrument in the acquisition process as well as calibration and quantization error [7, 8, 9, 10].

Noise in a hyperspectral image can be broadly grouped into two main classes: random noise and fixed pattern noise [11]. The focus of this paper is about random noise reduction. This random noise includes signal-dependent photonic noise and signal-independent thermal noise [12]. A widely used model for random noise in the hyperspectral image is the additive white noise along both spectral and spatial dimensions [13, 4, 14], which is reasonable when the thermal noise is dominant [11]. However, as the performance of the electronic component is improved in the new-generation hyperspectral sensors [15, 16], the signal-dependent photonic noise is as dominant as the signal-independent thermal noise, therefore the aforementioned white noise model is not appropriate.

Many applications such as classification and target detection require high signal-to-noise ratio hyperspectral image to achieve good performance. Therefore, noise reduction is required as preliminary step. Generally, in the classical hyperspectral image denoising methods, the matrix-algebra methods are applied on vectorized data tensor. But the incautiously splitting of tensor leads to a loss of information as the tensor is no longer considered as a whole: the relationships between channels are lost. To overcome these disadvantages, a hyperspectral image is considered as an entity, so as to keep jointly the spatial and spectral relationships of different channels [17, 18, 19, 20]. They proposed a multiway Wiener filter (MWF) to remove noise in the hyperspectral images and

the results are better than those of classical methods. The multiway Wiener filter uses multilinear algebra tools, such as TUCKER3 [21, 22], to extend the known Wiener filter to the tensor model. As the signal subspace dimension or rank is estimated for each mode, a rank(K_1, K_2, K_3) multiway Wiener filter was proposed to remove the noise [23, 24].

Distinguishing from MWF which treats a dataset as an entity [4], wavelet packet transform [25] was proposed to decompose a dataset into different sets of coefficients. To remove noise from the wavelet coefficients, Donoho and Johnstone proposed a strategy named *soft threshold* [26]. In this strategy, coefficients below the threshold are removed whereas those above the threshold are shrunk. Three algorithms *VisuShrink* [27], *SureShrink* [28] and *BayesShrink* [29] which is based on Bayes principle, were proposed to determine the threshold aforementioned. The noise variance required in these algorithms is estimated from the detail coefficients of highest level by the robust median estimator [27, 28]. To extend wavelet transform to the hyperspectral image case, a hyperspectral image is vectorized and processed as a *one*-dimensional signal, which obviously neglected the relationships among different elements of the tensor [30]. In [31] 2-D wavelet transform is performed to each band and averaged the criteria of all the bands to obtain the criterion for the whole hyperspectral image. Moreover, in [32] the first-order derivative for each spectral band is calculated and then performed 2-D wavelet to each band. These algorithms treat spatial and spectral information separately without considering the relationships between them.

In this paper, the main idea is to decompose a hyperspectral image into different coefficient tensors and jointly filter each of these tensors in three modes. To apply wavelet packet transform to a tensor conveniently, a *three*-dimensional wavelet packet transform is deduced. Through the multidimensional wavelet packet transform, a tensor $\mathcal{R} \in \mathbb{R}^{I_1 \times I_2 \times I_3}$ is transformed into a series of wavelet packet coefficient tensors. Each of these coefficient tensors can be treated as a component with different "frequencies", where the low frequency component contains more signal while the high frequency component contains more noise. The space of these coefficients is named as the multidimensional wavelet packet domain (3-WPD) in this paper. To jointly filter each component tensor in three modes, MWF is introduced into the 3-WPD in our proposed denoising algorithm 3-WPD-MWF. Moreover, the signal-dependent noise is also taken into account in 3-WPD-MWF. As MWF performs well when the additive noise is white [23, 24], to adapt 3-WPD-MWF for the signal-dependent noise situation, a noise-whitening method is proposed as well. Besides, the performance of 3-WPD-MWF is affected by the transform level and basis, therefore a risk function is also proposed to determine the best transform level and basis.

The remainder of the paper is organized as follows. Section briefly introduces the multiway data model and some basic knowledge of multilinear algebra. Section gives the signal model used in this paper. Section overviews the multiway Wiener filtering algorithm. In section , the 1-D wavelet packet transform is outlined. Section presents the proposed denoising method. Section gives some comparative classification results. Finally, section details the conclusion of the research undertaken.

2. Signal model and multilinear algebra tools

2.1. Signal model

A multiway signal is also called tensor. A tensor is a multidimensional array, $\mathcal{X} \in \mathbb{R}^{I_1 \times I_2 \times \ldots \times I_N}$, in which $\mathbb{R}$ indicates the real mainfold, and N is the number of dimensions. The elements in this tensor can be expressed as $x_{i_1 i_2 \ldots i_N}$, with $i_1 = 1, \ldots, I_1;\ i_2 = 1, \ldots, I_2; \cdots;\ i_N = 1, \ldots, I_N$. The n-th dimension of this tensor is called n-mode.

A hyperspectral image can be represented as a three-way array $\mathcal{X} \in \mathbb{R}^{I_1 \times I_2 \times I_3}$, where I_1 indicates the number of rows, I_2 the number of columns and I_3 the number of spectral channels. The three entries are related to pixel localization and spectral band.

2.2. Multilinear algebra tools

2.2.1. n-mode unfolding

$\mathbf{X}_n \in \mathbb{R}^{I_n \times M_n}$ denotes the n-mode unfolding matrix of a tensor $\mathcal{X} \in \mathbb{R}^{I_1 \times I_2 \times \ldots \times I_N}$, where $M_n = I_{n+1} \ldots I_1 I_N \ldots I_{n-1}$. 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.*, $K_n = \text{rank}\ (\mathbf{X}_n)$.

2.2.2. n-mode product

The n-mode product is defined as the product between a data tensor $\mathcal{X} \in \mathbb{R}^{I_1 \times I_2 \times \ldots \times I_N}$ and a matrix $\mathbf{B} \in \mathbb{R}^{J \times I_n}$ in mode n. This n-mode product is denoted by $\mathcal{C} = \mathcal{X} \times_n \mathbf{B}$, whose entries are given by

$$c_{i_1 \ldots i_{n-1} j i_{n+1} \ldots i_N} \triangleq \sum_{i_n=1}^{I_n} x_{i_1 \ldots i_{n-1} i_n i_{n+1} \ldots i_N} b_{j i_n} \tag{1}$$

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

2.2.3. Element extraction

For a given tensor $\mathcal{X} \in \mathbb{R}^{I_1 \times I_2 \times \ldots \times I_N}$, we define the element extraction operation:

$$\mathcal{X}(i_1, i_2, \ldots, i_N) \triangleq x_{i_1 i_2 \ldots i_N} \tag{2}$$

3. Signal modelling

In this paper we model a noisy hyperspectral image as a tensor $\mathcal{R} \in \mathbb{R}^{I_1 \times I_2 \times I_3}$ resulting from a multidimensional signal $\mathcal{X} \in \mathbb{R}^{I_1 \times I_2 \times I_3}$ 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:

$$\mathcal{R} = \mathcal{X} + \mathcal{N} \tag{3}$$

in which the noise $\mathcal{N}$ accounts for signal-dependent photonic noise $\mathcal{X}_{sr} \circledast \mathcal{P}$ and signal-independent thermal noise $\mathcal{T}$ [11, 16]:

$$\mathcal{N} = \mathcal{X}_{sr} \circledast \mathcal{P} + \mathcal{T} \tag{4}$$

where $\mathcal{X}_{sr}$ denotes a tensor obtained by calculating the square root of each element of $\mathcal{X}$ and $\circledast$ indicates the Hadamard product [33]. By using (3) and (4) each element of tensor $\mathcal{R}$ can be written as [12]:

$$r_{i_1 i_2 i_3} = x_{i_1 i_2 i_3} + n_{i_1 i_2 i_3} = x_{i_1 i_2 i_3} + \sqrt{x_{i_1 i_2 i_3}} p_{i_1 i_2 i_3} + t_{i_1 i_2 i_3} \tag{5}$$

where $p_{i_1 i_2 i_3}$ and $t_{i_1 i_2 i_3}$ are zero-mean Gaussian-distributed spectrally and spatially uncorrelated random values with variances σ^2_{p,i_3} and σ^2_{t,i_3} for band i_3, respectively. It is well known that the summation of two Gaussian-distributed random variables is also Gaussian-distributed, therefore $n_{i_1 i_2 i_3} = \sqrt{x_{i_1 i_2 i_3}} p_{i_1 i_2 i_3} + t_{i_1 i_2 i_3}$ is a conditional Gaussian-distributed random variable with its mean being zero and its variance being $\sigma^2_{n_{i_1 i_2 i_3}}$, which can be expressed as [12, 11]:

$$\sigma^2_{n_{i_1 i_2 i_3}} = \mathrm{E}\left[(\sqrt{x_{i_1 i_2 i_3}} p_{i_1 i_2 i_3} + t_{i_1 i_2 i_3})^2 | x_{i_1 i_2 i_3} \right] = x_{i_1 i_2 i_3} \sigma^2_{p,i_3} + \sigma^2_{t,i_3} \tag{6}$$

Signal-dependent noise becomes as dominant as the signal-independent noise only in the recent new-generation hyperspectral sensors [11, 12], therefore there are few denoising algorithms based on the signal-dependent noise model [12]. In the following section and section , we will overview multiway Wiener filter and wavelet packet transform which are usually used to remove white noise from the image. Then in section a new algorithm will be proposed to remove the signal-dependent and signal-independent noise from the hyperspectral image.

4. Multiway Wiener filtering

4.1. Denoising model

Multiway Wiener filter provides an estimate $\hat{\mathcal{X}}$ of the desired signal $\mathcal{X}$ from data tensor $\mathcal{R}$ using a *three*-dimensional filtering, which can be expressed as follows [19]:

$$\hat{\mathcal{X}} = \mathcal{R} \times_1 \mathbf{H}_1 \times_2 \mathbf{H}_2 \times_3 \mathbf{H}_3 \tag{7}$$

From the signal processing point of view, the n-mode product is a n-mode filtering of $\mathcal{R}$, and we call $\mathbf{H}_n$ the n-mode filter.

In order to obtain the optimal n-mode filters $\{\mathbf{H}_n,\ n = 1, 2, 3\}$, the mean squared error (MSE) between the estimated signal $\hat{\mathcal{X}}$ and the initial signal $\mathcal{X}$ should be minimized [19]:

$$e(\mathbf{H}_1, \mathbf{H}_2, \mathbf{H}_3) = E\left(\| \mathcal{X} - \mathcal{R} \times_1 \mathbf{H}_1 \times_2 \mathbf{H}_2 \times_3 \mathbf{H}_3 \|^2 \right) \tag{8}$$

In the classical multidimensional and multi-mode signal processing assumption, the n-mode vector space $E^{(n)}$ of dimension I_n associated with the n-mode of tensor $\mathcal{R}$ is the direct sum of signal subspace $E_1^{(n)}$ of dimension K_n and the noise subspace $E_2^{(n)}$ of dimension $I_n - K_n$, *i.e.*, $E^{(n)} = E_1^{(n)} \oplus E_2^{(n)}$ [19].

4.2. $\mathbf{H}_n$ estimation

The n-mode filters $\{\mathbf{H}_n,\ n = 1, 2, 3\}$ can be obtained by the minimization of the optimization criterion MSE in (8). By setting the derivation of MSE to zero, the expression of the optimal n-mode filter $\mathbf{H}_n$ is [19]:

$$\mathbf{H}_n = \mathbf{V}_s^{(n)} \mathbf{\Lambda}^{(n)} {\mathbf{V}_s^{(n)}}^T \tag{9}$$

where $\mathbf{V}_s^{(n)}$ is a matrix containing the K_n orthonormal basis vectors of n-mode signal subspace and

$$\mathbf{\Lambda}^{(n)} = diag\left(\frac{\lambda_1^{\gamma} - {\sigma_{\gamma}^{(n)}}^2}{\lambda_1^{\mathbf{\Gamma}}}, \ldots, \frac{\lambda_{K_n}^{\gamma} - {\sigma_{\gamma}^{(n)}}^2}{\lambda_{K_n}^{\mathbf{\Gamma}}} \right) \tag{10}$$

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

$$\gamma_{RR}^{(n)} = E\left[\mathbf{R}_n \mathbf{q}^{(n)} \mathbf{R}_n^T\right] \tag{11}$$

$$\Gamma_{RR}^{(n)} = E\left[\mathbf{R}_n \mathbf{Q}^{(n)} \mathbf{R}_n^T\right] \tag{12}$$

with

$$\mathbf{q}^{(n)} = \mathbf{H}_{p_1} \otimes \mathbf{H}_{p_2}, \tag{13}$$

$$\mathbf{Q}^{(n)} = \mathbf{H}_{p_1}^T \mathbf{H}_{p_1} \otimes \mathbf{H}_{p_2}^T \mathbf{H}_{p_2} \tag{14}$$

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

$$\hat{\sigma}_{\gamma}^{(n)^2} = \frac{1}{I_n - K_n} \sum_{i=K_n+1}^{I_n} \lambda_i^{\gamma} \tag{15}$$

4.3. ALS algorithm

To jointly find n-mode filters $\{\mathbf{H}_n,\ n = 1, 2, 3\}$ that minimize (8), an Alternating Least Square (ALS) algorithm [23] 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 follows:

1. **Input:** Data tensor $\mathcal{R}$
2. **Initialization** $k = 0$:
   $\mathcal{X}^0 = \mathcal{R} \Longleftrightarrow \mathbf{H}_n = \mathbf{I}_{I_n}\ \forall n = 1, 2, 3$. 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, $n = 1, 2, 3$,
   $K_n = \arg\min_{k_n} \left[\mathrm{AIC}(k_n)\right], k_n = 1, \ldots, I_n - 1$.

(b) Estimation of $\mathbf{H}_n^{k+1}$ for $n = 1, 2, 3$.

i. $\mathcal{X}_n^k = \mathcal{R} \times_p \mathbf{H}_p^{k+1} \times_q \mathbf{H}_q^k$. $p, q = 1, 2, 3, \quad p, q \neq n$ and $p < q$

ii. $\mathbf{H}_n^{k+1} = \underset{\mathbf{Z}_n}{\arg\min} \|\mathcal{X} - \mathcal{X}_n^k \times_n \mathbf{Z}_n\|^2$ 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 3b utilizes the filters in other modes $\{\mathbf{H}_i,\ 1 \leq i \leq 3 \text{ and } i \neq n\}$, it shows that MWF considers the relationships between elements in all modes of the dataset.

5. Wavelet packet transform

5.1. 1-D wavelet packet transform

Wavelet packet transform [34] can decompose data $\mathbf{r} \in \mathbb{R}^I$ into different components (coefficient vectors) $\mathbf{c}_{l,m} \in \mathbb{R}^{I/2^l}$, $m = 0, \ldots, 2^l - 1$ in transform level l. Especially, $\mathbf{c}_{0,0} = \mathbf{r}$ indicates the coefficients of transform level $l = 0$. The decomposition from level l to $l+1$ can be expressed as follows:

$$\begin{bmatrix} \mathbf{c}_{l+1,2m} \\ \mathbf{c}_{l+1,2m+1} \end{bmatrix} = \mathbf{W}_m^{l+1} \mathbf{c}_{l,m} \tag{16}$$

and the reconstruction:

$$\mathbf{c}_{l,m} = (\mathbf{W}_m^{l+1})^T \begin{bmatrix} \mathbf{c}_{l+1,2m} \\ \mathbf{c}_{l+1,2m+1} \end{bmatrix} \tag{17}$$

where $\mathbf{W}_m^{l+1}$ denotes wavelet transform matrix of coefficients $\mathbf{c}_{l,m}$. $\mathbf{W}_m^{l+1}$ is an orthogonal matrix. In this paper, only an orthogonal basis is used in the transform. By defining: $\bar{\mathbf{W}}^l = \text{diag}\left(\mathbf{W}_0^l, \mathbf{W}_1^l, \cdots, \mathbf{W}_{2^{l-1}-1}^l\right)$ and $\mathbf{c}_l = [\mathbf{c}_{l,0}, \mathbf{c}_{l,1}, \cdots, \mathbf{c}_{l,2^l-1}]^T$, $\mathbf{c}_l$ can be obtained directly from $\mathbf{r}$ and we only need to perform:

$$\mathbf{c}_l = \prod_{i=0}^{l} \bar{\mathbf{W}}^i \mathbf{c}_0 = \prod_{i=0}^{l} \bar{\mathbf{W}}^i \mathbf{r} \tag{18}$$

Then the decomposition and the reconstruction can be expressed as $\mathbf{c}_l = \tilde{\mathbf{W}}^l \mathbf{r}$ and $\mathbf{r} = \left(\tilde{\mathbf{W}}^l\right)^T \mathbf{c}_l$ respectively, where $\tilde{\mathbf{W}}^l = \prod_{i=0}^{l} \bar{\mathbf{W}}^i$.

5.2. Overview of SureShrink filter

Stein's unbiased risk estimator (SURE) Shrink minimizes the stein's unbiased risk for estimating the threshold. The formula to obtain *SureShrink* threshold is [28]:

$$\tau = \underset{0 \le \tau \le \sqrt{2 \log N_{l,m}}}{\arg\min} R_{SURE}(\tau, \mathbf{c}_{l,m}) \tag{19}$$

where $N_{l,m}$ is the number of wavelet coefficients in $\mathbf{c}_{l,m}$ and R_{SURE} is the SURE risk for a threshold τ, which is defined as:

$$R_{SURE}(\tau, \mathbf{c}_{l,m}) = N_{l,m} - 2n_{l,m} + \sum_{k=1}^{N_{l,m}} [\min(\mid \mathbf{c}_{l,m}(k) \mid, \tau)]^2 \tag{20}$$

in which $n_{l,m}$ is the number of elements in set $\{k :\mid \mathbf{c}_{l,m}(k) \mid \le \tau\}$ and $\mathbf{c}_{l,m}(k)$ indicates the k-th element of vector $\mathbf{c}_{l,m}$. After obtaining the threshold τ, we can use soft threshold [26] to filter the coefficients:

$$\eta_T(\mathbf{c}_{l,m}) = \text{sgn}(\mathbf{c}_{l,m}) \circledast \max(|\mathbf{c}_{l,m}| - \tau \mathbf{1}, \mathbf{0}) \tag{21}$$

Where $\mathbf{1}$ and $\mathbf{0}$ denote all-one vector all-zero vector, respectively.

In this paper, a hyperspectral image is modeled as a *third*-order tensor to conserve the element position information. In the following section, the multidimensional wavelet packet transform (MWPT) in tensor form will be derived to transform a hyperspectral image into its component tensors, which build the multidimensional wavelet packet domain (3-WPD). *SureShrink* will be applied to filter the component tensor, and for convenience, this algorithm is named 3-WPD-SURE.

To apply *SureShrink* to the component tensor, we firstly convert the tensor into its vectorization form [33], then use (19) to obtain the threshold τ, and finally perform soft threshold (21) to the elements of the vector. After SureShrink, the vector is reshaped into its tensor form. For each tensor, the threshold calculated from (19) is different. In other words, the threshold changes adaptively with the tensor.

From (19) and (21) we know that *SureShrink* processes a component tensor without considering the position of each its element, which contains the signal information. Therefore, an algorithm will be proposed in the next section to apply MWF in the 3-WPD.

6. Proposed method

6.1. Noise pre-whitening

The main idea of the proposed method is to filter the wavelet packet coefficients by MWF better than *SureShrink*. As MWF performs well when the noise is white [35], it is necessary to whiten the noise in the hyperspectral image. As aforementioned in section , $n_{i_1i_2i_3}$ obeys the zero-mean Gaussian distribution with variance $\sigma^2_{n_{i_1i_2i_3}}$, therefore the element $r_{i_1i_2i_3}$ in the hyperspectral image can be whitened as:

$$\underline{r}_{i_1i_2i_3} = \frac{r_{i_1i_2i_3}}{\sigma_{n_{i_1i_2i_3}}} \tag{22}$$

However, in (22), $\sigma_{n_{i_1i_2i_3}}$ is unknown in the realistic case, therefore it should be estimated from the hyperspectral image. In [11], a HYNPE algorithm is proposed to estimate σ^2_{p,i_3} and σ^2_{t,i_3}. In the estimation procedure of HYNPE, the estimate $\tilde{x}_{i_1i_2i_3}$ of the signal $x_{i_1i_2i_3}$ is obtained by the MLR-theory-based method [36, 37], which calculates the estimate of the signal in band i_3 [33] by the linear prediction of the other I_3-1 bands. Furthermore, $\hat{\sigma}^2_{p,i_3}$ and $\hat{\sigma}^2_{t,i_3}$ are obtained by maximizing the likelihood function [11]:

$$\{\hat{\sigma}^2_{p,i_3}, \hat{\sigma}^2_{t,i_3}\} = \underset{\substack{\sigma_{p,i_3}>0\\ \sigma_{t,i_3}>0}}{\arg\max} \ln\left(\sigma_{p,i_3}, \sigma_{t,i_3}\right) \tag{23}$$

with

$$\ln\left(\sigma_{p,i_3}, \sigma_{t,i_3}\right) = -\frac{M}{2}\ln(2\pi) - \frac{1}{2}\sum_{i_1=1}^{I_1}\sum_{i_2=1}^{I_2}\ln\left[\sigma^2_{p,i_3}\cdot x_{i_1i_2i_3} + \sigma^2_{t,i_3}\right] - \frac{1}{2}\sum_{i_1=1}^{I_1}\sum_{i_2=1}^{I_2}\frac{n^2_{i_1i_2i_3}}{\sigma^2_{p,i_3}\cdot x_{i_1i_2i_3} + \sigma^2_{t,i_3}} \tag{24}$$

in (23) $\hat{\sigma}^2_{p,i_3}$ and $\hat{\sigma}^2_{t,i_3}$ are the estimates of σ^2_{p,i_3} and σ^2_{t,i_3} respectively. After $\hat{\sigma}^2_{p,i_3}$ and $\hat{\sigma}^2_{t,i_3}$ are obtained, the estimate of the noise variance can be obtained by using (6). As $x_{i_1i_2i_3}$ in (6) is unknown, we use the estimate $\tilde{x}_{i_1i_2i_3}$ instead. Finally the estimate of the noise variance for each element can be calculated as:

$$\hat{\sigma}^2_{n_{i_1i_2i_3}} = \tilde{x}_{i_1i_2i_3}\hat{\sigma}^2_{p,i_3} + \hat{\sigma}^2_{t,i_3} \tag{25}$$

Therefore, the prewhitening process can be expressed as:

$$\underline{r}_{i_1i_2i_3} = \underline{x}_{i_1i_2i_3} + \underline{n}_{i_1i_2i_3} = \frac{x_{i_1i_2i_3}}{\hat{\sigma}_{n_{i_1i_2i_3}}} + \frac{n_{i_1i_2i_3}}{\hat{\sigma}_{n_{i_1i_2i_3}}} \tag{26}$$

and its corresponding tensor form can be written as:

$$\underline{\mathcal{R}} = \underline{\mathcal{X}} + \underline{\mathcal{N}} \tag{27}$$

To get the estimated signal $\hat{\mathcal{X}}$, an inverse process of prewhitening is necessary after we get the denoised result $\underline{\hat{\mathcal{X}}}$. The inverse prewhitening process can be obtained by multiplying each element by the noise standard deviation:

$$\hat{x}_{i_1i_2i_3} = \underline{\hat{x}}_{i_1i_2i_3}\hat{\sigma}_{n_{i_1i_2i_3}} \tag{28}$$

where $\underline{\hat{x}}_{i_1i_2i_3}$ is the element of tensor $\underline{\hat{\mathcal{X}}}$, which is the estimate of $\underline{\mathcal{X}}$ obtained by the following sections.

6.2. 3-dimensional wavelet packet transform in tensor form

The multidimensional wavelet packet transform (3-WPT) can be computed by performing 1-D wavelet packet transform in each mode [38]. Therefore, the wavelet packet coefficient tensor $\mathcal{C}^{\mathcal{R}}_{\mathbf{l}}$ can be computed as:

$$\mathcal{C}^{\mathcal{R}}_{\mathbf{l}} = \underline{\mathcal{R}} \times_1 \tilde{\mathbf{W}}^{l_1}_1 \times_2 \tilde{\mathbf{W}}^{l_2}_2 \times_3 \tilde{\mathbf{W}}^{l_3}_3 \tag{29}$$

and the reconstruction can be written as:

$$\underline{\mathcal{R}} = \mathcal{C}^{\mathcal{R}}_{\mathbf{l}} \times_1 (\tilde{\mathbf{W}}^{l_1}_1)^T \times_2 (\tilde{\mathbf{W}}^{l_2}_2)^T \times_3 (\tilde{\mathbf{W}}^{l_3}_3)^T \tag{30}$$

Where $\mathbf{l} = [l_1, l_2, l_3]^T$ and $l_1, l_2, l_3 \geq 0$. Especially, when $l_1, l_2, l_3 > 0$, 3-WPT indicates the 3-wavelet packet transform. $\tilde{\mathbf{W}}^{l_k}_k$ denotes the l_k level wavelet packet transform to k-th mode of $\underline{\mathcal{R}}$. The wavelet packet coefficient

tensors constitute the 3-dimensional wavelet packet domain (3-WPD). $\mathcal{C}^{\mathcal{R}}_{\mathbf{l},\mathbf{m}}$ is defined as the coefficient sub-tensor of $\mathcal{C}^{\mathcal{R}}_{\mathbf{l}}$, where $\mathbf{m} = [m_1, m_2, m_3]^T$ is the index vector, and $0 \leq m_k \leq 2^{l_k} - 1$, $k = 1, 2, 3$. Then for each element of $\mathcal{C}^{\mathcal{R}}_{\mathbf{l},\mathbf{m}}$ we can define:

$$\mathcal{C}^{\mathcal{R}}_{\mathbf{l},\mathbf{m}}(j_1,\ j_2,\ j_3) \triangleq \mathcal{C}^{\mathcal{R}}_{\mathbf{i}}(\mathbf{J}_1(j_1),\ \mathbf{J}_2(j_2),\ \mathbf{J}_3(j_3)) \tag{31}$$

where

$$\left\{ \mathbf{J}_n = \left[\frac{m_n I_n}{2^l}, \ldots, \frac{(m_n+1)I_n}{2^l} - 1 \right]^T, n = 1, 2, 3 \right\} \tag{32}$$

and

$$j_n \in \left\{ 1,\ \ldots,\ \frac{I_n}{2^l} \right\},\ n = 1, 2, 3 \tag{33}$$

The notation $\mathcal{C}^{\mathcal{R}}_{\mathbf{l},\mathbf{m}}(j_1,\ j_2,\ j_3)$ indicates the element of tensor $\mathcal{C}^{\mathcal{R}}_{\mathbf{l},\mathbf{m}}$ in position $(j_1,\ j_2,\ j_3)$ as defined in (2). From the properties of the wavelet packet transform, we know that m_n indicates the "frequency" of mode n. Thus, $\mathbf{m}$ is the frequency index of coefficient block $\mathcal{C}^{\mathcal{R}}_{\mathbf{l},\mathbf{m}}$. For convenience, a component tensor of $\underline{\mathcal{R}}$ is referred to as $\mathcal{C}^{\mathcal{R}}_{\mathbf{l},\mathbf{m}}$ in this paper.

6.3. Multiway Wiener filter in *three*-dimensional wavelet packet domain

In the existing MWF algorithm, the filter is applied to the whole hyperspectral image $\underline{\mathcal{R}}$. As the calculation of the filters in expression (9) needs the estimation of the signal subspace or rank in each mode for suppressing the smallest eigenvalues, some weak signal might be removed in this procedure. Therefore, the SNR is an important factor influencing the rank. When SNR is higher, the rank estimated is greater, therefore more signal is preserved in the filtering process. In the contrast condition, more signal is lost. When the noise is white, the power of noise in each component $\mathcal{C}^{\mathcal{R}}_{\mathbf{l},\mathbf{m}}$ is the same, while the signal concentrates in the lower frequency component. That is to say, in different components, the SNR is different. When MWF is applied to each component, more signal can be preserved. Performing 3-WPT to tensor $\underline{\mathcal{R}}$, $\underline{\mathcal{X}}$ and $\underline{\mathcal{N}}$ in expression (27), we obtain:

$$\begin{aligned} &\underline{\mathcal{R}} \times_1 \tilde{\mathbf{W}}_1^{l_1} \times_2 \tilde{\mathbf{W}}_2^{l_2} \times_3 \tilde{\mathbf{W}}_3^{l_3} \\ &= (\underline{\mathcal{X}} + \underline{\mathcal{N}}) \times_1 \tilde{\mathbf{W}}_1^{l_1} \times_2 \tilde{\mathbf{W}}_2^{l_2} \times_3 \tilde{\mathbf{W}}_3^{l_3} \\ &= \underline{\mathcal{X}} \times_1 \tilde{\mathbf{W}}_1^{l_1} \times_2 \tilde{\mathbf{W}}_2^{l_2} \times_3 \tilde{\mathbf{W}}_3^{l_3} + \underline{\mathcal{N}} \times_1 \tilde{\mathbf{W}}_1^{l_1} \times_2 \tilde{\mathbf{W}}_2^{l_2} \times_3 \tilde{\mathbf{W}}_3^{l_3} \end{aligned} \tag{34}$$

The coefficient tensor of each part:

$$\mathcal{C}^{\mathcal{R}}_{\mathbf{l}} = \underline{\mathcal{R}} \times_1 \tilde{\mathbf{W}}_1^{l_1} \times_2 \tilde{\mathbf{W}}_2^{l_2} \times_3 \tilde{\mathbf{W}}_3^{l_3} \tag{35}$$

$$\mathcal{C}^{\mathcal{X}}_{\mathbf{l}} = \underline{\mathcal{X}} \times_1 \tilde{\mathbf{W}}_1^{l_1} \times_2 \tilde{\mathbf{W}}_2^{l_2} \times_3 \tilde{\mathbf{W}}_3^{l_3} \tag{36}$$

$$\mathcal{C}^{\mathcal{N}}_{\mathbf{l}} = \underline{\mathcal{N}} \times_1 \tilde{\mathbf{W}}_1^{l_1} \times_2 \tilde{\mathbf{W}}_2^{l_2} \times_3 \tilde{\mathbf{W}}_3^{l_3} \tag{37}$$

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

$$\hat{\mathcal{C}}^{\mathcal{X}}_{\mathbf{l}} = \hat{\underline{\mathcal{X}}} \times_1 \tilde{\mathbf{W}}_1^{l_1} \times_2 \tilde{\mathbf{W}}_2^{l_2} \times_3 \tilde{\mathbf{W}}_3^{l_3} \tag{38}$$

Extracting the components of each frequency $\mathcal{C}^{\mathcal{R}}_{\mathbf{l},\mathbf{m}}$, $\mathcal{C}^{\mathcal{X}}_{\mathbf{l},\mathbf{m}}$ and $\mathcal{C}^{\mathcal{N}}_{\mathbf{l},\mathbf{m}}$ from $\mathcal{C}^{\mathcal{R}}_{\mathbf{l}}$, $\mathcal{C}^{\mathcal{X}}_{\mathbf{l}}$ and $\mathcal{C}^{\mathcal{N}}_{\mathbf{l}}$ respectively by using(31), we obtain:

$$\mathcal{C}^{\mathcal{R}}_{\mathbf{l},\mathbf{m}} = \mathcal{C}^{\mathcal{X}}_{\mathbf{l},\mathbf{m}} + \mathcal{C}^{\mathcal{N}}_{\mathbf{l},\mathbf{m}} \tag{39}$$

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

$$\|\underline{\mathcal{X}} - \hat{\underline{\mathcal{X}}}\|^2 = \|\mathcal{C}^{\mathcal{X}}_{\mathbf{l}} - \hat{\mathcal{C}}^{\mathcal{X}}_{\mathbf{l}}\|^2 = \sum_{\mathbf{m}} \|\mathcal{C}^{\mathcal{X}}_{\mathbf{l},\mathbf{m}} - \hat{\mathcal{C}}^{\mathcal{X}}_{\mathbf{l},\mathbf{m}}\|^2 \tag{40}$$

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

$$\hat{\mathcal{C}}^{\mathcal{X}}_{\mathbf{l},\mathbf{m}} = \mathcal{C}^{\mathcal{R}}_{\mathbf{l},\mathbf{m}} \times_1 \mathbf{H}_{1,\mathbf{m}} \times_2 \mathbf{H}_{2,\mathbf{m}} \times_3 \mathbf{H}_{3,\mathbf{m}} \tag{41}$$

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 . After estimating $\hat{\mathcal{C}}^{\mathcal{X}}_{\mathbf{l},\mathbf{m}}$ for each $\mathbf{m}$, we obtain $\hat{\mathcal{C}}^{\mathcal{X}}_{\mathbf{l}}$ by concatenating $\hat{\mathcal{C}}^{\mathcal{X}}_{\mathbf{l},\mathbf{m}}$. Furthermore, the estimate $\hat{\underline{\mathcal{X}}}$ can be obtained by inverse MWPT:

$$\hat{\underline{\mathcal{X}}} = \hat{\mathcal{C}}^{\mathcal{X}}_{\mathbf{l}} \times_1 (\tilde{\mathbf{W}}^{l_1}_1)^T \times_2 (\tilde{\mathbf{W}}^{l_2}_2)^T \times_3 (\tilde{\mathbf{W}}^{l_3}_3)^T \tag{42}$$

6.4. Best transform level and basis selection

In the proposed algorithm, several parameters should be determined:

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

$$N_{L_k} = \lceil \log_2 I_k \rceil - 5, \ k = 1, 2, 3 \tag{43}$$

where $\lceil \cdot \rceil$ rounds a number upward to its nearest integer and the constant 5 is reduced 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:

$$L_k = \{0, 1, \cdots, N_{L_k}\}, \ k = 1, 2, 3 \tag{44}$$

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:

$$W = \{\mathrm{w}_1, \ \mathrm{w}_2, \cdots, \mathrm{w}_{N_W}\} \tag{45}$$

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 $R_c(\underline{\mathcal{X}}, \hat{\underline{\mathcal{X}}}) = \mathrm{E}\left[\|\underline{\mathcal{X}} - \hat{\underline{\mathcal{X}}}\|^2\right]$ [39], whose equivalent form using the coefficients can be expressed as:

$$R_c(\underline{\mathcal{X}}, \hat{\underline{\mathcal{X}}}) = \sum_{\mathbf{m}} \mathrm{E}\left[\|\mathcal{C}^{\mathcal{X}}_{\mathbf{l},\mathbf{m}} - \hat{\mathcal{C}}^{\mathcal{X}}_{\mathbf{l},\mathbf{m}}\|^2\right] \tag{46}$$

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

$$\mathbf{l}, w = \underset{l_k \in L_k, \ w \in W}{\arg\min} \sum_{\mathbf{m}} \mathrm{E}\left[\|\mathcal{C}^{\mathcal{X}}_{\mathbf{l},\mathbf{m}} - \hat{\mathcal{C}}^{\mathcal{X}}_{\mathbf{l},\mathbf{m}}\|^2\right], \ k = 1, 2, 3 \tag{47}$$

As the selection of the optimal $\mathbf{l}, w$ depends on $\underline{\mathcal{X}}$ which is generally unknown, to overcome this drawback an alternative solution should be found. Denoting by $\hat{\mathcal{C}}^{\mathcal{X}}_{\mathbf{l},\mathbf{m}}[d]$ the estimate of $\mathcal{C}^{\mathcal{X}}_{\mathbf{l},\mathbf{m}}$ at the d-th ALS loop aforementioned in section and noticing that when $\|\hat{\mathcal{C}}^{\mathcal{X}}_{\mathbf{l},\mathbf{m}}[d] - \hat{\mathcal{C}}^{\mathcal{X}}_{\mathbf{l},\mathbf{m}}[d-1]\|^2$ is minimized, $\hat{\mathcal{C}}^{\mathcal{X}}_{\mathbf{l},\mathbf{m}} \triangleq \hat{\mathcal{C}}^{\mathcal{X}}_{\mathbf{l},\mathbf{m}}[d]$ is the optimal estimate of $\mathcal{C}^{\mathcal{X}}_{\mathbf{l},\mathbf{m}}$ obtained by MWF, and at the same time $\mathrm{E}\left[\|\mathcal{C}^{\mathcal{X}}_{\mathbf{l},\mathbf{m}} - \hat{\mathcal{C}}^{\mathcal{X}}_{\mathbf{l},\mathbf{m}}\|^2\right]$ is minimized according to section . Therefore (47) can be replaced by:

$$\mathbf{l}, w = \underset{l_k \in L_k, \ w \in W}{\arg\min} \hat{R}_c, \ k = 1, 2, 3 \tag{48}$$

where

$$\hat{R}_c = \sum_{\mathbf{m}} \|\hat{\mathcal{C}}^{\mathcal{X}}_{\mathbf{l},\mathbf{m}}[d] - \hat{\mathcal{C}}^{\mathcal{X}}_{\mathbf{l},\mathbf{m}}[d-1]\|^2 \tag{49}$$

.

6.5. Summary of the proposed method

The proposed algorithm, denoted by 3-WPD-MWF, can be summarized as follows:

1. Prewhiten noise in the original hyperspectral image:
(a) Estimate noise variance σ^2_{p,i_3} and σ^2_{t,i_3} and the signal estimate $\hat{x}_{i_1 i_2 i_3}$ by algorithm HYNPE.
(b) Whiten the original hyperspectral image using (26) and obtain the tensor form: $\underline{\mathcal{R}} = \underline{\mathcal{X}} + \underline{\mathcal{N}}$.
2. Find the optimal $l_1, l_2, l_3 \in L$ and $w \in W$. Loop l_1, l_2, l_3 and w:
(a) Decompose the whitened data $\underline{\mathcal{R}}$ by 3-WPT: $\mathcal{C}^{\mathcal{R}}_{\mathbf{l}} = \underline{\mathcal{R}} \times_1 \tilde{\mathbf{W}}^{l_1}_1 \times_2 \tilde{\mathbf{W}}^{l_2}_2 \times_3 \tilde{\mathbf{W}}^{l_3}_3$.
(b) Extract component $\mathcal{C}^{\mathcal{R}}_{\mathbf{l},\mathbf{m}}$ from $\mathcal{C}^{\mathcal{R}}_{\mathbf{l}}$ by equation (31), for $\mathbf{m} = [m_1, m_2, m_3]^T$, where $0 \le m_k \le 2^{l_k} - 1$, $k = 1, 2, 3$.
(c) Filter component $\mathcal{C}^{\mathcal{R}}_{\mathbf{l},\mathbf{m}}$ by MWF: $\hat{\mathcal{C}}^{\mathcal{X}}_{\mathbf{l},\mathbf{m}} = \mathcal{C}^{\mathcal{R}}_{\mathbf{l},\mathbf{m}} \times_1 \mathbf{H}_{1,\mathbf{m}} \times_2 \mathbf{H}_{2,\mathbf{m}} \times_3 \mathbf{H}_{3,\mathbf{m}}$.
(d) Calculate the risk $\hat{R}_c$ using (49). If $\hat{R}_c$ reaches a fixed threshold, return the optimal l_1, l_2, l_3, w and $\hat{\mathcal{C}}^{\mathcal{X}}_{\mathbf{l},\mathbf{m}}$.
3. Concatenate $\hat{\mathcal{C}}^{\mathcal{X}}_{\mathbf{l},\mathbf{m}}$ to obtain $\mathcal{C}^{\mathcal{X}}_{\mathbf{l}}$ and perform inverse MWPT: $\hat{\underline{\mathcal{X}}} = \hat{\mathcal{C}}^{\mathcal{X}}_{\mathbf{l}} \times_1 (\tilde{\mathbf{W}}^{l_1}_1)^T \times_2 (\tilde{\mathbf{W}}^{l_2}_2)^T \times_3 (\tilde{\mathbf{W}}^{l_3}_3)^T$.
4. Perform inverse prewhitening procedure to $\hat{\underline{\mathcal{X}}}$ and then the estimate of the noise-free image: $\hat{\mathcal{X}}$ can be obtained.

The corresponding intuitive flowchart of 3-WPD-MWF is illustrated in Fig. 1.

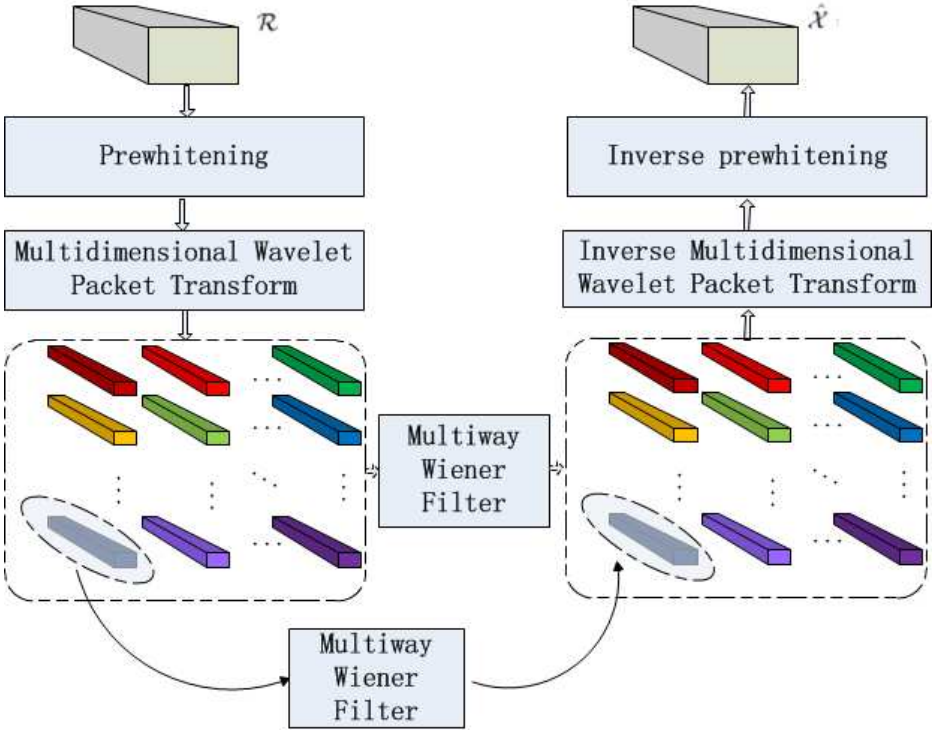


Figure 1: Flowchart of proposed algorithm

7. Experimental Results

7.1. Dataset in experiments

The dataset used in the experiments is the hyperspectral image which was captured by the airborne visible/infrared imaging spectrometer (AVIRIS) from a mixed forest/agricultural site at the Indian Pine test site in north-west Indiana. This image was taken at altitude 19812 m by the National Aeronautics and Space Administration (NASA)/ Jet Propulsion Laboratory. The original image size is $145 \times 145 \times 220$ ($I_1 = 145,\ I_2 = 145,\ I_3 = 220$), which means 145×145 pixels and 220 spectral bands [15]. To represent the signal to noise ratio in $\mathcal{R}$, an input SNR is introduced:

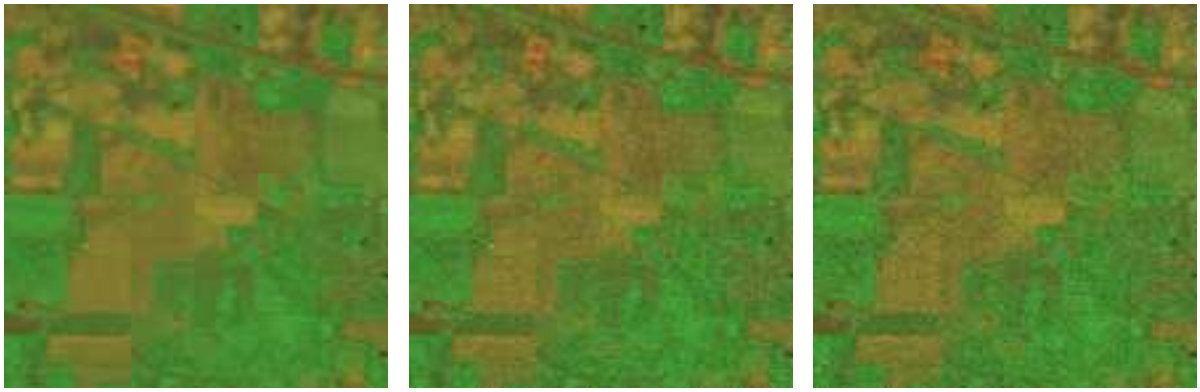
$$\mathrm{SNR}_{\mathrm{INPUT}} = 10 \log(\frac{\|\mathcal{X}\|^2}{\|\mathcal{N}\|^2}) \tag{50}$$

The $\mathrm{SNR}_{\mathrm{INPUT}}$ interval of interest in the experiments is from 10dB to 40dB including 7 $\mathrm{SNR}_{\mathrm{INPUT}}$ values: 10dB, 15dB, 20dB, 25dB, 30dB, 35dB and 40dB. Two types of noise will be added to the hyperspectral image in the experiments. The first type is signal-independent white noise, which is widely used in the classical hyperspectral image noise model. This model is a particular case of model (4) when $\|\mathcal{P}\|^2 = 0$ and $\mathcal{T}$ is white spatially and spectrally. This noise model is used to prove that the proposed algorithm performs well in the classical white noise situation. The second type is a combination of signal-dependent and signal-independent noise, whose model is as (4). As the power of signal-dependent noise and the power of signal-independent noise are of same level [11], in this paper, only the case $\mathrm{E}\left[\|\mathcal{X}_{sr} \circledast \mathcal{P}\|^2\right] = \mathrm{E}\left[\|\mathcal{T}\|^2\right]$ is taken into account. The RGB composites of $\mathcal{X}$ and $\mathcal{R}$ with two types of noise are shown in Fig. 2. The noise variance in each band of the second type noise is presented in Fig. 3.

7.2. Experiment settings

The size of the hyperspectral image is $145 \times 145 \times 220$. After padding values to each dimension to perform MWPT, the dataset size changes to $256 \times 256 \times 256$. By using (43), we can obtain $N_L = 3$. Then the transform level set can be expressed as:

$$L_k = \{0, 1, 2, 3\},\ k = 1, 2, 3 \tag{51}$$



(a) RGB composites of $\mathcal{X}$ (b) RGB composites of $\mathcal{R}$ ($\text{SNR}_{\text{INPUT}} = 20$dB, noise type 1) (c) RGB composites of $\mathcal{R}$ ($\text{SNR}_{\text{INPUT}} = 20$dB, noise type 2)

Figure 2: RGB composites of $\mathcal{X}$ and $\mathcal{R}$ (band 25, 65 and 80 for red, green and blue)

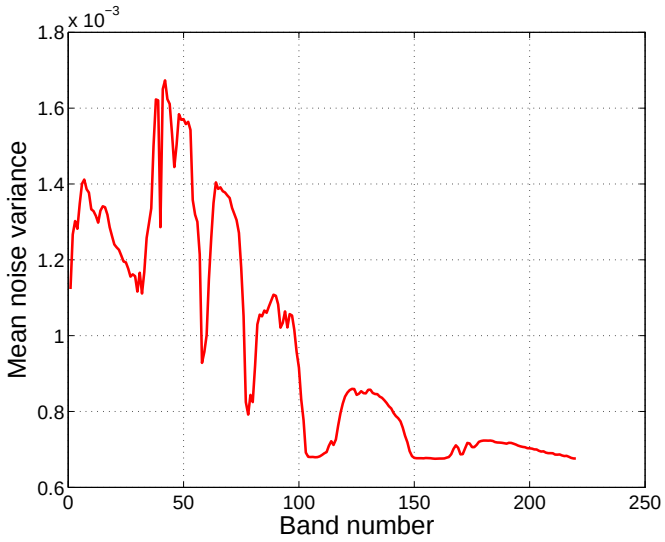


Figure 3: Noise variance in each band ($\text{SNR}_{\text{INPUT}} = 20$ dB, noise type 2)

Apart from the transform level, the wavelet type should also be taken into account. In this paper, only several usually used wavelets are chosen for the proposed MWPT method. The wavelets used in the experiments are: coif1, coif2, db1, db2, db3. Hence the number of wavelets is $N_W = 5$ and the wavelet set can be expressed as:

$$W = \{\text{coif1, coif2, db1, db2, db3}\} \tag{52}$$

The main purpose of hyperspectral image denoising is to improve the results of classification, detection, etc. In this paper, the classification is used to evaluate the performance of the considered denoising algorithms. A number of 16 classes [40] are expected in the hyperspectral image and the support vector machine (SVM) algorithm [41], which is very largely applied to HSI data, is used. The kernel function of the SVM classifier used in this experiment is radial basis function (RBF) $\exp\left(-\gamma * |u - v|^2\right)$ with $\gamma = 1$ and the regularization parameter $C = 100$. The reference data of the 16 classes are shown in the ground truth (Fig. 4), which is supplied with the original data [40]. A proportion of 10% of the reference data of each class is selected randomly as the training samples, whereas all the reference data are used as testing samples [42]. The numbers of training and testing samples are shown in TABLE 1. Note that the minimum number of training samples is set to 10 for the rare classes such as Grass/pasture-mowed.

In the experiments, MWF, 3-WPD-SURE and 3-WPD-MWF are applied to remove noise from the pre-whitened hyperspectral image. As 3-WPD-SURE is compared to 3-WPD-MWF, its optimal parameter combination is selected by minimizing the MSE, which relies on the noise-free image. In practice, the results of

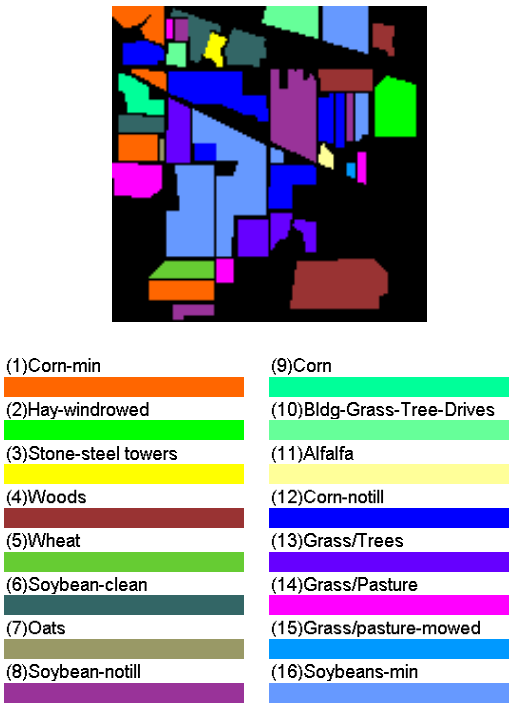


Figure 4: Classification reference data: ground truth of the area with 16 classes

Table 1: Training and testing samples of classification

ID	Class	Training Samples	Testing Samples
1	Corn-min	85	842
2	Hay-windrowed	49	489
3	Stone-steel towers	10	95
4	Woods	130	1294
5	Wheat	22	212
6	Soybean-clean	62	614
7	Oats	10	20
8	Soybean-notill	97	968
9	Corn	24	234
10	Bldg-Grass-Tree-Drives	38	380
11	Alfalfa	10	54
12	Corn-notill	144	1435
13	Grass/Trees	75	747
14	Grass/Pasture	50	500
15	Grass/pasture-mowed	10	26
16	Soybeans-min	247	2468
	Total	1063	10378

3-WPD-SURE should be worse than those obtained in these experiments. Then, after denoising by each algorithm, the denoised image is classified by SVM. The performances of these three algorithms are compared through the classification results.

7.3. Evaluation criteria: Overall Accuracy (OA)

To evaluate the experimental results and compare the proposed algorithm with others, the commonly used criteria is introduced below: To appreciate quantifiable comparisons, we determine the overall accuracy (OA) in percentage exhibited by SVM classifier. For P classes C_i, $i = 1, \ldots, P$; if a_{ij} is the number of testing samples that actually belong to class C_i and are classified into C_j for $i, j = 1, \ldots, P$, then OA is defined as follows: OA$= \frac{1}{N_{\text{total}}} \sum_{i=1}^{i=P} a_{ii}$, where N_{total} is the total number of samples, P is the number of classes C_i for $i = 1, \ldots, P$ and a_{ii} is equal to a_{ij} for $i = j$. That is, in this paper, OA is defined as:

$$\text{OA} = \frac{N_{\text{correct}}}{N_{\text{total}}} \times 100\% \tag{53}$$

where N_{correct} is the number of testing samples classified correctly into their corresponding classes. The higher the OA, the better the classification result.

7.4. Optimal parameter combination

The optimal parameter combinations of 3-WPD-MWF obtained by using (48) are given in TABLE 2 and TABLE 3 for noise type 1 and noise type 2 respectively. These two tables also present the optimal parameter combination of 3-WPD-SURE, which is obtained by minimizing the MSE. We can notice from these two tables that the parameter combination changes as a function of the $\text{SNR}_{\text{INPUT}}$ to allow 3-WPD-MWF to filter as much noise as possible. Since different wavelet types are designed for different cases, when the $\text{SNR}_{\text{INPUT}}$ changes, it is better to select a suitable wavelet automatically than to use the same for all situations. Moreover, the higher the transform level the smaller component size. This might decrease the performance of the rank estimation in MWF. Therefore a higher depth of the wavelet decomposition does not always improve the performance of 3-WPD-MWF, which can be seen from the transform levels of 3-WPD-MWF.

7.5. Classification Results

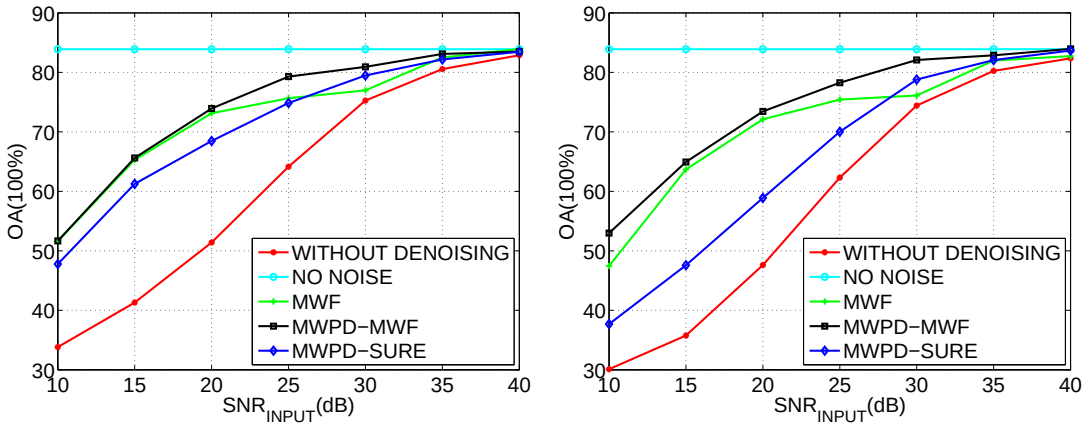
Table 2: Optimal parameter combination for each $\mathrm{SNR}_{\mathrm{INPUT}}$ (noise type 1)

Name	$\mathrm{SNR}_{\mathrm{INPUT}}$	10	15	20	25	30	35	40
3-WPD-MWF	w	db3	db3	coif1	coif2	db1	coif2	db1
	$[l_1, l_2, l_3]$	[3,3,0]	[2,2,0]	[1,1,0]	[3,3,0]	[3,3,0]	[3,3,0]	[3,3,0]
3-WPD-SURE	w	coif2	coif2	coif2	coif2	coif2	db3	coif2
	$[l_1, l_2, l_3]$	[3,3,3]	[3,3,3]	[3,3,3]	[3,3,3]	[3,3,3]	[3,3,3]	[3,3,3]

Table 3: Optimal parameter combination for each $\mathrm{SNR}_{\mathrm{INPUT}}$ (noise type 2)

Name	$\mathrm{SNR}_{\mathrm{INPUT}}$	10	15	20	25	30	35	40
3-WPD-MWF	w	db3	coif1	coif1	coif1	coif2	db3	db3
	$[l_1, l_2, l_3]$	[3,3,0]	[1,1,0]	[1,2,0]	[1,1,0]	[1,1,0]	[3,3,0]	[2,3,0]
3-WPD-SURE	w	db3	coif2	db3	coif2	coif2	db3	db3
	$[l_1, l_2, l_3]$	[3,3,3]	[3,3,3]	[3,3,3]	[3,3,3]	[3,3,3]	[3,3,3]	[3,3,3]

To evaluate the performance of the proposed algorithm in improving the classification efficiency, the classification overall accuracy is presented in Fig. 5. Fig. 5(a) and Fig. 5(b) show the classification results obtained on the denoised hyperspectral image which is perturbed by white noise and generalized noise (expression (5)) respectively. It can be seen that the classification results in Fig. 5(a) are better than the corresponding ones in Fig. 5(b). Both Fig. 5(a) and Fig. 5(b) show that the proposed algorithm 3-WPD-MWF performs better than 3-WPD-SURE and MWF in improving the classification results as well. MWF performs well when the $\mathrm{SNR}_{\mathrm{INPUT}}$ is low, while 3-WPD-SURE becomes slightly better than MWF when the $\mathrm{SNR}_{\mathrm{INPUT}}$ is high. Because 3-WPD-MWF is the combination of 3-WPD-SURE and MWF, it takes the advantages of 3-WPD-SURE and MWF, therefore it performs better than both of them. In fact, in the denoising process, 3-WPD-MWF can preserve more signal details, which is important to distinguish two classes, therefore it is reasonable for 3-WPD-MWF to exhibit a higher OA. Moreover, the classification results of hyperspectral image without noise are also given to provide a reference for the comparison of the classification results of the SVM used in this experiment.

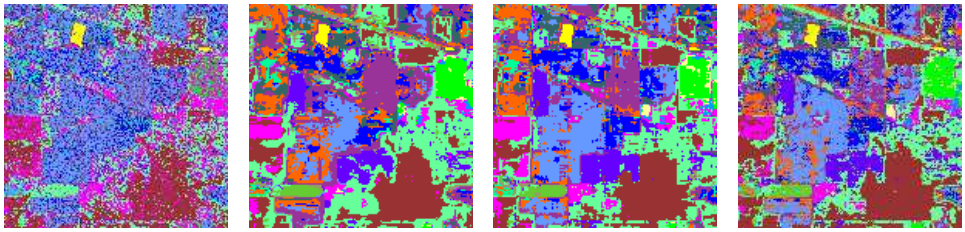


(a) Noise type 1: white noise

(b) Noise type 2: generalized noise model

Figure 5: Classification OA with respect to $\mathrm{SNR}_{\mathrm{INPUT}}$

To exemplify the performances of the considered algorithms, Fig. 6 and Fig. 7 show the results of classification at 15 dB and 30 dB. At 15 dB, it can be seen that the performances of MWF and 3-WPD-MWF are almost same and both of them are better than 3-WPD-SURE, which can be seen especially in the Corn-notill class. While at 30 dB, the performances of 3-WPD-SURE and 3-WPD-MWF are better than MWF, which is obvious also in the Corn-notill class. Moreover, at both 15 dB and 30 dB, the classification results are greatly improved after denoising, which implies that the denoising procedure is necessary for improving the classification results.

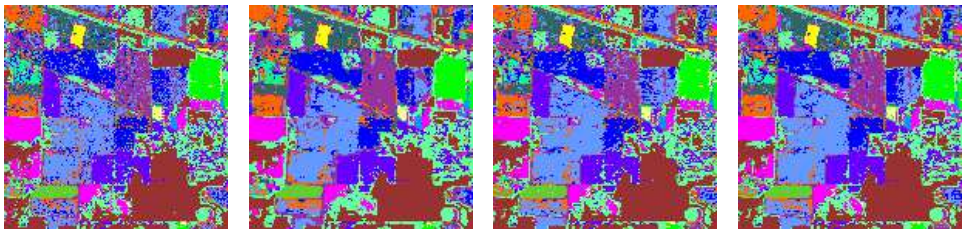


(a) Classification map of dataset without denoising ($\mathrm{SNR_{INPUT}} = 15\mathrm{dB}$)

(b) Classification map by 3-WPD-SURE ($\mathrm{SNR_{INPUT}} = 15\mathrm{dB}$)

(c) Classification map by MWF ($\mathrm{SNR_{INPUT}} = 15\mathrm{dB}$)

(d) Classification map by 3-WPD-MWF ($\mathrm{SNR_{INPUT}} = 15\mathrm{dB}$)



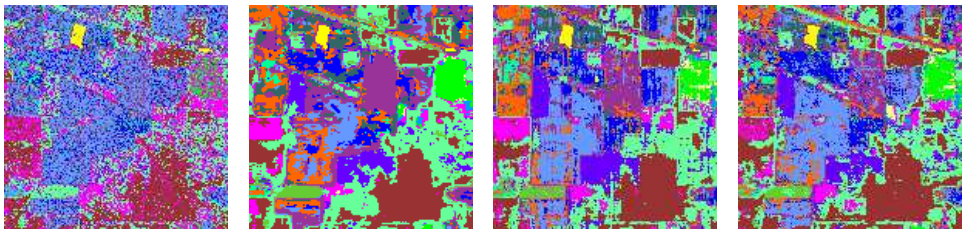
(e) Classification map of dataset without denoising ($\mathrm{SNR_{INPUT}} = 30\mathrm{dB}$)

(f) Classification map by 3-WPD-SURE ($\mathrm{SNR_{INPUT}} = 30\mathrm{dB}$)

(g) Classification map by MWF ($\mathrm{SNR_{INPUT}} = 30\mathrm{dB}$)

(h) Classification map by 3-WPD-MWF ($\mathrm{SNR_{INPUT}} = 30\mathrm{dB}$)

Figure 6: Classification results (noise type 1)

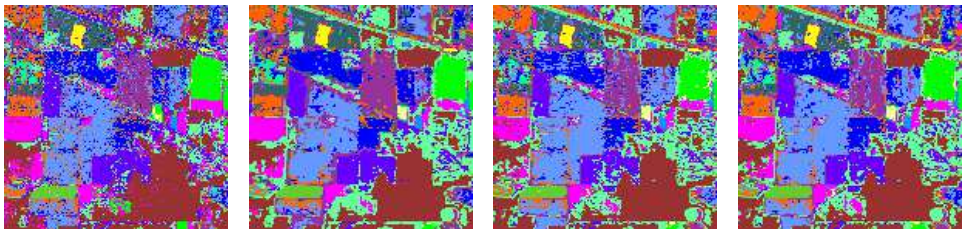


(a) Classification map of dataset without denoising ($SNR_{INPUT} = 15dB$)

(b) Classification map by 3-WPD-SURE ($SNR_{INPUT} = 15dB$)

(c) Classification map by MWF ($SNR_{INPUT} = 15dB$)

(d) Classification map by 3-WPD-MWF ($SNR_{INPUT} = 15dB$)



(e) Classification map of dataset without denoising ($SNR_{INPUT} = 30dB$)

(f) Classification map by 3-WPD-SURE ($SNR_{INPUT} = 30dB$)

(g) Classification map by MWF ($SNR_{INPUT} = 30dB$)

(h) Classification map by 3-WPD-MWF ($SNR_{INPUT} = 30dB$)

Figure 7: Classification results (noise type 2)

8. Conclusion

In this paper, an algorithm named 3-WPD-MWF is proposed to reduce the signal-dependent and signal-independent noise in a hyperspectral image. An element-by-element noise whitening method is presented to whiten the noise in the hyperspectral image. Then the multidimensional wavelet packet transform (3-WPT) in tensor form is deduced to transform a hyperspectral image into different components. Thereafter, MWF is used to jointly filter each component to remove noise. Since the SNR of each component is different, MWF can adjust adaptively as a function of the SNR to remove as much noise as possible. As the performance of 3-WPD-MWF is influenced by its parameter combination, a risk function is proposed as well to determine the optimal parameter combination for 3-WPD-MWF.

The AVIRIS dataset of Indian Pine is used to evaluate the denoising and classification results of MWF, 3-WPD-SURE and 3-WPD-MWF respectively. The experimental results show the efficiency of the proposed denoising algorithm to improve the SNR in hyperspectral image and the classification results.

Since in this paper the optimal parameter combination is found by the time consuming brute force searching, future works will be focused on the reduction of the computational load. Two possible ways might be resorted to achieve this issue: A heuristic algorithm can be used to search for the optimal (sub-optimal) parameter combination, the construction of the optimal wavelet, different from any wavelet existing in libraries. The main difference of the computational load between 3-WPD-MWF and 3-WPD-SURE is the way chosen to filter each component. 3-WPD-MWF filters each component by MWF as a tensor, while 3-WPD-SURE filters each component by *SureShrink* as a vector. Due to the joint filtering of each dimension of the component, the computation cost of 3-WPD-MWF is slightly greater than 3-WPD-SURE.

Acknowledgements

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

References

[1] Z. Liu, H. Wang, and Q. Li. Tongue tumor detection in medical hyperspectral images. Sensors 1 (2011) 162–174.

[2] S. Lewis, A. Hudak, R. Ottmar, P. Robichaud, L. Lentile, S. Hood, J. Cronan, and P. Morgan. Using hyperspectral imagery to estimate forest floor consumption from wildfire in boreal forests of alaska, usa. Int. J. Wildland Fire 2 (2011) 255–271.
[3] D. Landgrebe. Hyperspectral image data analysis. IEEE Signal Process. Mag. 1 (2002) 17–28.
[4] D. Muti and S. Bourennane. Survey on tensor signal algebraic filtering. Signal Process. 2 (2007) 237–249.
[5] J. Marot, C. Fossati, and S. Bourennane. Overview on advances in tensor data denoising methods. SIAM J. Matrix Anal. Appl. 3 (2008) 1172–1204.
[6] A. Soltani-Farani, H. R. Rabiee, and S. A. Hosseini. Spatial-aware dictionary learning for hyperspectral image classification. IEEE Trans. Geosci. Remote Sens. 1 (2015) 527–541.
[7] J. Kerekes and J. Baum. Hyperspectral imaging system modeling. Linc Lab. J. 1 (2003) 117–130.
[8] J. Kerekes and J. Baum. Full-spectrum spectral imaging system analytical model. IEEE Trans. Geosci. Remote Sens. 3 (2005) 571–580.
[9] B. Aiazzi, L. Alparone, A. Barducci, S. Baronti, P. Marcoinni, I. Pippi, and M. Selva. Noise modelling and estimation of hyperspectral data from airborne imaging spectrometers. Ann. Geophys. 1 (2006) 1–9.
[10] E. Christophe, D. Leger, and C. Mailhes. Quality criteria benchmark for hyperspectral imagery. IEEE Trans. Geosci. Remote Sens. 9 (2005) 2103–2114.
[11] N. Acito, M. Diani, and G. Corsini. Signal-dependent noise modeling and model parameter estimation in hyperspectral images. IEEE Trans. Geosci. Remote Sens. 8 (2011) 2957–2971.
[12] M. L. Uss, B. Vozel, V. V. Lukin, and K. Chehdi. Local signal-dependent noise variance estimation from hyperspectral textural images. IEEE J. Sel. Topics Signal Process. 3 (2011) 469–486.
[13] L. He, J. Li, C. Liu and S. Li. Recent Advances on Spectral-Spatial Hyperspectral Image Classification: An Overview and New Guidelines. IEEE Trans. Geosci. Remote Sens. 3 (2018) 1579–1597.
[14] N. Renard, S. Bourennane, and J. Blanc-Talon. Denoising and dimensionality reduction using multilinear tools for hyperspectral images. IEEE Geosci. Remote Sens. Lett. 2 (2008) 138–142.
[15] D. Landgrebe. Signal theory methods in multispectral remote sensing. New Jersey: Wiley, (2003).
[16] L. Alparone, M. Selva, B. Aiazzi, S. Baronti, F. Butera, and L. Chiarantini. Signal-dependent noise modelling and estimation of new-generation imaging spectrometers. Hyperspectral Image and Signal Processing: Evolution in Remote Sensing, WHISPERS 09 (2009) 1–4.
[17] D. Muti and S. Bourennane. Multidimensional filtering based on a tensor approach. Signal Process. 12 (2005) 2338–2353.
[18] D. Muti and S. Bourennane. Multiway filtering based on fourth-order cumulants. EURASIP Appl. Signal Process. 7 (2005) 1147–1158.
[19] D. Muti, S. Bourennane, and J. Marot. Lower-rank tensor approximation and multiway filtering. SIAM J. Matrix Anal. Appl. 3 (2008) 1172–1204.
[20] D. Letexier and S. Bourennane. Noise removal from hyperspectral images by multidimensional filtering. IEEE Trans. Geosci. Remote Sens. 7 (2008) 2061–2069.
[21] L. De Lathauwer, B. De Moor, and J. Vandewalle. On the best rank-1 and rank-(r1, r2,..., rn) approximation of higher-order tensors. SIAM J. Matrix Anal. Appl. 4 (2000) 1324–1342.
[22] L. De Lathauwer and J. Vandewalle. Dimensionality reduction in higher-order signal processing and rank-(r1, r2,..., rn) reduction in multilinear algebra. SIAM J. Matrix Anal. Appl. 4 (2000) 1324–1342.
[23] N. Renard and S. Bourennane. Improvement of target detection methods by multiway filtering. IEEE Trans. Geosci. Remote Sens. 8 (2008) 2407–2417.
[24] S. Bourennane, C. Fossati, and A. Cailly. Improvement of target-detection algorithms based on adaptive three-dimensional filtering. IEEE Trans. Geosci. Remote Sens. 4 (2010) 1383–1395.
[25] I. Daubechies. Orthonormal bases of compactly supported wavelets. Commun. Pure Appl. Math. 7 (1988) 909–996.
[26] D. Donoho. De-noising by soft-thresholding. IEEE Trans. Inf. Theory 3 (1995) 613–627.
[27] D. Donoho and J. Johnstone. Ideal spatial adaptation by wavelet shrinkage. Biometrika 3 (1994) 425–455.
[28] D. Donoho and I. Johnstone. Adapting to unknown smoothness via wavelet shrinkage. J. Amer. Statist. Assoc. 432 (1995) 1200–1244.
[29] S. Chang, B. Yu, and M. Vetterli. Adaptive wavelet thresholding for image denoising and compression. IEEE Trans. Image Process. 9 (2000) 1532–1546.
[30] I. Atkinson, F. Kamalabadi, S. Mohan, and D. Jones. Wavelet-based 2-d multichannel signal estimation. Proc. IEEE IGARSS 2 (2003) 743–745.
[31] A. Pizurica and W. Philips. Estimating the probability of the presence of a signal of interest in multiresolution single-and multiband image denoising. IEEE Trans. Image Process. 3 (2006) 654–665.
[32] H. Othman and S. Qian. Noise reduction of hyperspectral imagery using hybrid spatial-spectral derivative-domain wavelet shrinkage. IEEE Trans. Geosci. Remote Sens. 2 (2006) 397–408.
[33] A. Cichocki, R. Zdunek, A. Phan, and S. Amari. Nonnegative matrix and tensor factorizations: applications to exploratory multi-way data analysis and blind source separation. New Jersey: Wiley, (2009).
[34] R. Coifman and M. Wickerhauser. Entropy-based algorithms for best basis selection. IEEE Trans. Inf. Theory 2 (1992) 713–718.
[35] X. Liu, S. Bourennane, and C. Fossati. Nonwhite noise reduction in hyperspectral images. IEEE Geosci. Remote Sens. Lett. 3 (2012) 368–372.
[36] R. Roger and J. Arnold. Reliably estimating the noise in aviris hyperspectral images. Int. J. Remote Sens. 10 (1996) 1951–1962.
[37] I. C. Chein and D. Qian. Estimation of number of spectrally distinct signal sources in hyperspectral imagery. IEEE Trans. Geosci. Remote Sens. 3 (2004) 608–619.
[38] S. Mallat. A theory for multiresolution signal decomposition: The wavelet representation. IEEE Trans. Pattern Anal. Mach. Intell. 7 (1989) 674–693.
[39] D. Donoho and I. Johnstone. Ideal denoising in an orthonormal basis chosen from a library of bases. Comptes Rendus de l'Academie des Sciences-Serie I-Mathematique 12 (1994) 1317–1322.
[40] D. Landgrebe. Multispectral data analysis: A signal theory perspective," Purdue Univ., West Lafayette, IN (1998).
[41] C.-C. Chang and C.-J. Lin. LIBSVM: A library for support vector machines. ACM Trans. Intell. Syst. Technol.2 (2011)

1–27:27.

[42] Q. Yuan, L. Zhang, and H. Shen. Hyperspectral image denoising employing a spectral-spatial adaptive total variation model. IEEE Trans. Geosci. Remote Sens. 99 (2012) 1–18.