Structured Sparse Coding-Based Hyperspectral Imagery Denoising With Intracluster Filtering

Wei Wei, Member, IEEE, Lei Zhang, Student Member, IEEE, Chunna Tian, Member, IEEE, Antonio Plaza, Fellow, IEEE, and Yanning Zhang, Senior Member, IEEE

Abstract—Sparse coding can exploit the intrinsic sparsity of hyperspectral images (HSIs) by representing it as a group of sparse codes. This strategy has been shown to be effective for HSI denoising. However, how to effectively exploit the structural information within the sparse codes (structured sparsity) has not been widely studied. In this paper, we propose a new method for HSI denoising, which uses structured sparse coding and intracluster filtering. First, due to the high spectral correlation, the HSI is represented as a group of sparse codes by projecting each spectral signature onto a given dictionary. Then, we cast the structured sparse coding into a covariance matrix estimation problem. A latent variable-based Bayesian framework is adopted to learn the covariance matrix, the sparse codes, and the noise level simultaneously from noisy observations. Although the considered strategy is able to perform denoising through accurately reconstructing spectral signatures, an inconsistent recovery of sparse codes may corrupt the spectral similarity in each spatial homogeneous cluster within the scene. To address this issue, an intracluster filtering scheme is further employed to restore the spectral similarity in each spatial cluster, which results in better denoising results. Our experimental results, conducted using both simulated and real HSIs, demonstrate that the proposed method outperforms several state-of-the-art denoising methods.

Index Terms—Covariance matrix estimation, hyperspectral images (HSIs) denoising, intracluster filtering, structured sparse coding.

I. INTRODUCTION

HYPER SPECTRAL imaging is a technique that collects the spectral information across a certain range of the electromagnetic spectrum at narrow wavelengths (e.g., 10 nm) [1]. The obtained hyperspectral images (HSIs) thus exhibit an approximately continuous spectrum at each pixel. Such wealth of spectral information enables HSIs to better represent the imaged scene and thus greatly enhance the performance of extensive practical applications, such as target detection [2], [3], scene classification [4], [5], and video tracking [6]. However, HSIs are often affected by noise during image collection, transmission, etc. [7], [8], and the noise level varies across bands [9], [10]. Since the performance of many applications (e.g., classification [11]) is sensitive to noise, HSIs denoising is one of the fundamental steps prior to HSI exploitation. According to the relation to signal, noise on HSIs can be roughly divided into two types, namely, signal-dependent noise and signal-independent noise. Both types have been widely studied in HSIs denoising [8]. In this paper, we mainly focus on methods for signal-independent noise.

To date, many effective HSI denoising methods have been proposed. For example, by considering the HSI as a third tensor, the low-rank tensor approximation (LRAT) was employed in [12] for denoising purposes. Liu et al. [13] exploited the parallel factor (PARAFAC) analysis model to decompose the HSI tensor as a sum of several rank-1 tensors. The noise was expected to be reduced by exploring the low-rank characteristics of the HSIs. As opposed to these methods that consider the whole HSI in the denoising process, Peng et al. [8] collected similar full band patches to form a cluster tensor. Then, a group-block-sparsity constraint was imposed on the cluster tensor to restrict the nonlocal spectrum similarity in a tensor decomposition framework, which led to better denoising performance. In addition to these tensor-based methods, some existing 2-D image denoising methods have been extended for HSIs. Based on the nonlocal means (NLM) filter [14] method, Qian et al. [15] extended the spatially local patches to a spatial–spectral local tensor in order to explore the spectral–spatial correlation in HSIs. Block-matching and 3-D filtering [16], which groups the similar 2-D patches together to enhance the sparsity and employs collaborative filtering, has achieved state-of-the-art denoising performance on traditional images. Inspired by this, Maggioni et al. [17] proposed a block-matching and 4-D (BM4D) filtering method for HSIs.

In recent years, sparse coding (which exploits the intrinsic sparsity of HSIs by representing them as a group of sparse codes on a proper dictionary) has demonstrated effectiveness. For example, Dabov et al. [16] employed \( \ell_1 \) norm to model the intrinsic sparsity of HSIs on an overcomplete 3-D wavelet dictionary. In [18], \( \ell_1 \) norm-based sparsity regularization...
on sparsifying the HSI with a proper spectrum dictionary to explore the sparsity within each spectrum. Given a proper spectrum dictionary \( D \in \mathbb{R}^{n_p \times n_d} \), each spectrum can be represented by a sparse code. The corresponding sparse coding of \( X \) can be formulated as follows:

\[
\min_{Y} \sum_{i=1}^{n_p} \|y_i\|_0, \quad \text{s.t.} \ X = DY
\]

(1)

where \( Y = [y_1, \ldots, y_n] \) \( y_i \in \mathbb{R}^{n_d \times n_p} \) denotes all sparse codes and the column vector \( y_i \) indicates the sparse code for the spectrum associated with the \( i \)th pixel. \( \|y_i\|_0 \) denotes the \( \ell_0 \) norm of \( y_i \), which counts the nonzero components in \( y_i \).

When considering the representation error, the sparse coding can be reformulated as

\[
\min_{Y} \sum_{i=1}^{n_p} \|y_i\|_0, \quad \text{s.t.} \ X = DY \quad \text{Frobenius norm} \leq \epsilon
\]

(2)

where \( \epsilon \) is a predefined scalar determined by the representation error and the Frobenius norm. Due to the \( \ell_0 \) norm, it is NP hard to solve (1) or (2). Thus, when some mild conditions are satisfied, \( \ell_0 \) norm is often substituted by the \( \ell_p \) norm with \( 0 < p \leq 1 \), which casts the sparse coding into a convex or nonconvex optimization problem as

\[
\min_{Y} \sum_{i=1}^{n_p} \|y_i\|_p, \quad \text{s.t.} \ X = DY \quad \text{Frobenius norm} \leq \epsilon
\]

(3)

where \( \|y_i\|_p = (\sum_{j=1}^{n_d} |y_{ij}|^p)^{1/p} \) and \( y_{ij} \) is the \( j \)th component in \( y_i \). For example, when the restricted isometry property holds, \( \ell_0 \) norm can be equally replaced by \( \ell_1 \) norm, which results in a convex optimization problem. Most sparse coding-based HSI denoising methods directly utilize (2), (3), or their variants. Since the noise-free HSI can be sparsely represented on a proper dictionary while noise cannot, sparse coding has exhibited good performance for HSIs denoising. However, sparse coding models each component in the sparse code independently. For example, \( \ell_1 \) norm on \( y_i \) follows the following prior distribution:

\[
p(y_i) \propto \exp(-\alpha \|y_i\|_1) = \exp \left( -\sum_{j=1}^{n_d} \alpha |y_{ij}| \right)
\]

\[
= \prod_{j=1}^{n_d} \exp(-\alpha |y_{ij}|)
\]

(4)

where \( \alpha \) is a predefined scalar. From (4), we can find that \( \ell_1 \) norm on \( y_i \) amounts to imposing the identity independent distribution on each component \( y_{ij} \). Thus, they fail to capture the underlying structure within the sparse code.

To address this problem, structured sparse coding has been adopted in recent years, which has proved to perform better in representation than the standard sparse coding shown in (2) or (3) [24]. Therefore, it is natural to harness the structured sparse coding to enhance the denoising performance of HSIs. To this end, we will engineer a novel structured sparse coding model of HSIs in the Bayesian way in this paper.
III. PROPOSED METHOD

In this paper, we mainly focus on the additive noise corruption on HSIs, and thus the noisy observation model can be formulated as

\[ F = X + N = DY + N \]  

where \( F \in \mathbb{R}^{n_s \times n_p} \) denotes the noisy observation and \( N \in \mathbb{R}^{n_s \times n_p} \) indicates the random noise. According to Section II, \( X \) can be sparsely represented on a given spectrum dictionary \( D \) as \( X = DY \). Each column of \( Y \) denotes a sparse code. Considering that the noise level varies across bands \([9]\), we assume that \( X \) is corrupted by various levels of Gaussian white noise across bands. Thus, \( N \) can be modeled by a matrix normal distribution as \( MN(0, \Sigma_n, I) \), where \( \Sigma_n = \text{diag}(\lambda)^1 \) represents the noise variances across bands with \( \lambda = [\lambda_1, \ldots, \lambda_{n_s}]^T \). \( I \) is an identity matrix with proper size and implies that noise corruption in each column of \( Y \) is independent. By defining a weighted trace norm operation \( \| Q \|_{\Sigma_n} = (\text{tr}(Q^2 \Sigma_n^{-1} Q))^1/2 \), we can formulate the likelihood of the noisy observation as

\[ p(F|Y, \lambda) \propto |\Sigma_n|^{-n_p/2} \exp \left\{ -\frac{1}{2} \| DY - F \|_{\Sigma_n}^2 \right\}. \]  

(6)

A. Structured Sparsity Prior

A zero-mean matrix normal distribution is utilized to model \( Y \) as

\[ p(Y|\Sigma_y) \propto |\Sigma_y|^{-n_p/2} \exp \left\{ -\frac{1}{2} \| Y \|_{\Sigma_y}^2 \right\} \]

\[ = \prod_{i=1}^{n_p} |\Sigma_y|^{-1/2} \exp \left\{ -\frac{1}{2} y_i^T \Sigma_y^{-1} y_i \right\}. \]  

(7)

where \( \Sigma_y \in \mathbb{R}^{n_s \times n_s} \) is the covariance matrix indicating the correlation among different rows of \( Y \). In previous Bayesian sparse learning schemes \([29], [30]\), \( \Sigma_y \) is assumed to be diagonal. When a diagonal component \( \Sigma_y(j, j) \) approaches zero, the \( j \)th dimension \( y_{ji} \) of the sparse code \( y_i \) also tends to be zero and vice versa. Therefore, a plausible diagonal \( \Sigma_y \) can depict the sparsity of the considered signal. However, when \( \Sigma_y \) is diagonal, (7) fails to capture the structure within the sparse code, because (7) in this case amounts to imposing independent priors on each component of the sparse code \( y_i \) as follows:

\[ p(y_i|\Sigma_y) \propto |\Sigma_y|^{-1/2} \exp \left\{ y_i^T \Sigma_y^{-1} y_i \right\} \]

\[ = \prod_{j=1}^{n_s} |\Sigma_y(j, j)|^{-1/2} \exp \left\{ -\frac{1}{2} y_{ji}^2/(2\Sigma_y(j, j)) \right\}. \]  

(8)

To capture the underlying structure within each sparse code, we adopt a full covariance matrix \( \Sigma_y \) for (7). In the following, we will illustrate our motivation using a probabilistic graph model theory. Provided that the sparse code \( y_i \) lies in an undirected graph and each component \( y_{ji} \) denotes a node in the graph, the probability distribution of \( y_i \) can be modeled using Markov random field (MRF) as

\[ p(y_i) = \frac{1}{Z} \exp \left\{ -\sum_j \phi(y_{ji}) - \sum_{j,k \neq j} \psi(y_{ji}, y_{ki}) \right\}. \]  

(9)

where we only consider the unary potential \( \phi(y_{ji}) \) and the pairwise potential \( \psi(y_{ji}, y_{ki}) \). \( Z \) is the normalization term. Since the correlation among different nodes can be well represented by the pairwise correlation, this MRF model is able to represent the underlying graph structure within \( y_i \). When \( \phi(y_{ji}) = \Sigma_y^{-1}(j, j)y_{ji} \) and \( \psi(y_{ji}, y_{ki}) = \Sigma_y^{-1}(j, k)y_{ji}y_{ki} \) are both linear functions, \( p(y_i|\Sigma_y) \) can be reformulated as

\[ p(y_i) = \frac{1}{Z} \exp \left\{ -\sum_{j,k} \Sigma_y^{-1}(j, k)y_{ji}y_{ki} \right\} \]

\[ \propto |\Sigma_y|^{-1/2} \exp \left\{ -\frac{1}{2} y_i^T \Sigma_y^{-1} y_i \right\}. \]  

(10)

This means that the prior in (7) with a full covariance matrix \( \Sigma_y \) depicts an undirected graph structure within each \( y_i \). Specifically, the components in \( \Sigma_y^{-1} \) denote the weights in the linear unary and pairwise potentials, and the corresponding pairwise potentials model the correlation among components in each \( y_i \). In addition, the sparsity of each \( y_i \) is represented by the unary term as traditional Bayesian sparse learning \([29], [30]\). Therefore, the prior in (7) with a full covariance matrix \( \Sigma_y \) can model the structured sparsity of \( Y \) more suitably, compared with a diagonal \( \Sigma_y \).

To further clarify this point, we sparsify each spectrum from a real HSI [shown in Fig. 1(a)] on an orthogonal wavelet dictionary and the corresponding sparse codes \( Y \) of the first 200 pixels are shown in Fig. 1(b). It has shown that a tree structure exists within each obtained sparse code \([20], [21]\). To illustrate that those structured sparse codes exhibit a non-diagonal covariance matrix, we obtain the empirical covariance matrix \(^2\) on all sparse codes, which represents the correlation among different components of each sparse code [shown in Fig. 1(c)]. It can be seen that the obtained covariance matrix is non-diagonal. Therefore, in the following, we will mainly focus on how to estimate a plausible covariance matrix \( \Sigma_y \) to represent the structured sparsity of \( Y \).

B. Latent Variable-Based Structured Sparse Coding

To infer the clean image \( X \) from the noisy observation \( F \) with the structured sparsity prior in (7), we have to infer the graph structure (i.e., \( \Sigma_y \)) as a prior. Many previous works learned a general graph structure from extensive training examples \([31]–[33]\). However, the learned general structure cannot well fit the specific data distribution and thus leads to limited denoising performance. In this paper, we instead try to learn the data-dependent graph structure directly from the noisy observations. To this end, we first introduce a hyperprior on the covariance matrix \( \Sigma_y \) for learning more flexible graph structure. There are mainly two kinds of priors. When there is some known prior information

\(^1\text{diag}() \) denotes the ‘diag()’ function in MATLAB.

\(^2\)The covariance matrix is obtained as \( \text{cov}(Y) \) in MATLAB.
of $\Sigma_y$ (e.g., specific structure [34]), we can impose an inverse-Wishart distribution on $\Sigma_y$ as
\[
p(\Sigma_y) \propto |\Sigma_y|^{(n_p+l-1)/2} \exp\left( -\frac{1}{2} \text{tr}(\Theta \Sigma_y^{-1}) \right) \tag{11}
\]
where $\Theta$ is a reference matrix with prior information of $\Sigma_y$ and $l$ denotes the degree of freedom. This prior encourages $\Sigma_y$ to approach the reference matrix $\Theta$. However, only some special applications (see [34]) have available prior information. In most cases, there is no prior information on $\Sigma_y$. To handle the cases without any prior information, we impose the noninformative prior on each component of $\Sigma_y$ as
\[
p(\Sigma_y(i,j)) \propto \frac{1}{t} \tag{12}
\]
where $t$ is a given scalar. This prior encourages to learn the $\Sigma_y$ directly from the observed data without any prior information. Since we expect to adapt the imposed structured sparsity prior to the data distribution, we adopt the second prior in this paper.

With this noninformative prior on $\Sigma_y$, we then employ a latent variable-based Bayesian framework to estimate $\Sigma_y$ and noise level $\lambda$ from the noisy observation $F$ as
\[
\max_{\Sigma_y, \lambda} p(\Sigma_y, \lambda|F) \propto \int p(F|Y, \lambda)p(Y|\Sigma_y) \prod_{ij} p(\Sigma_y(i,j))dY. \tag{13}
\]
where $Y$ acts as the latent variable. To clarify the relationship between variables, the hierarchical structure of this model is given in Fig. 2. By introducing the negative logarithm operation, (13) can be formulated as
\[
\min_{\Sigma_y, \lambda} \text{tr}(F^T \Sigma_m^{-1} F) + n_p \log |\Sigma_m| \tag{14}
\]
where $\Sigma_m = \Sigma_n + D \Sigma_y D^T$. If the optimal $\Sigma_y$ and $\lambda$ have been learned from (14), the sparse codes $Y$ can be inferred by the maximum a posteriori (MAP) estimation based on the likelihood in (6) and structured sparsity prior in (7) as
\[
\max_Y p(Y|F) \propto p(F|Y, \lambda)p(Y|\Sigma_y). \tag{15}
\]

Fig. 1. Structured sparse codes of the spectra in a real HSI on an orthogonal wavelet dictionary. (a) 3-D data cubes. (b) Sparse codes of the spectra on the first 200 pixels in the HSI, where each column denotes a sparse code. (c) Nondiagonal covariance matrix of the sparse codes, which represents the correlation among different components within each sparse code. It can be seen that the structured sparsity within each code results in a nondiagonal covariance matrix.

![Hierarchical structure of the proposed latent variable-based structured sparse coding.](Image1)

However, it is intractable to solve (14) directly. Inspired by [26], we have the following relationship:
\[
\text{tr}(F^T \Sigma_m^{-1} F) = \min_{Y} \|DY - F\|_{\Sigma_n}^2 + \|Y\|_{\Sigma_y}^2 \tag{16}
\]
which leads to a restrictive upper bound of $\text{tr}(F^T \Sigma_m^{-1} F)$ as
\[
\forall Y, \text{tr}(F^T \Sigma_m^{-1} F) \leq \|DY - F\|_{\Sigma_n}^2 + \|Y\|_{\Sigma_y}^2. \tag{17}
\]
Substituting this upper bound into (14), we obtain a unified optimization formula as
\[
\min_{Y, \Sigma_y, \lambda} \|DY - F\|_{\Sigma_n}^2 + \|Y\|_{\Sigma_y}^2 + n_p \log |\Sigma_m| \tag{18}
\]
where the sparse codes $Y$, covariance matrix $\Sigma_y$, and noise level $\lambda$ are jointly modeled. More importantly, (18) can be effectively solved as seen in Section IV. Inspired by [27] and [26], we can prove that the optimum $Y$ from (18) equals to that from the MAP estimation in (15) with those learned $\Sigma_y$ and $\lambda$ from (14). Therefore, we need to solve only the unified optimization in (18) for structured sparse coding.

Since the sparse codes $Y$, covariance matrix $\Sigma_y$, and noise level $\lambda$ can be jointly optimized from the noisy observation, the learned structure of $Y$ is data dependent and robust to the unknown noise, which guarantees to reconstruct the sparse codes accurately. Given the learned sparse codes $Y$ from (18), the denoised HSI can be obtained as $\hat{X} = DY$. 

C. Intracluster Filtering

Although the proposed structured sparse coding can recover a clean HSI from its noisy observation, the inconsistent recovery of sparse codes caused by modeling them independently
in (7) may limit the denoising performance. Specifically, natural scenes often exhibit spatially local and nonlocal similarity [8], [35]. Since homogeneous materials share similar spectral signatures [1], spatially local and nonlocal similarity in the spatial domain, we assume that a clean HSI can be spatially divided into \( K \) homogeneous clusters according to the spectral similarity. Thus, each cluster contains similar spectra from local and nonlocal similar pixels. This provides a strong prior for clean HSIs. However, the inconsistent recovery of sparse codes corrupts the spectral similarity in each spatial homogeneous cluster within the HSI, which may produce an unnatural spatial appearance, as illustrated in the first row of Fig. 3(b)–(d).

To restore the spectral similarity in each cluster, we develop an intracluster filtering method. Specifically, we employ a clustering method (e.g., K-means++ [36]) to spatially divide the HSI \( \hat{X} \) reconstructed in Section III-B into \( K \) clusters based on the spectral similarity. Let \( \hat{X}_k = \{ \hat{x}_k^1, \ldots, \hat{x}_k^{nk} \} \in \mathbb{R}^{nk \times nk} \) denote the spectra of pixels in the \( k \)th cluster, where \( nk \) denotes the pixels number in this cluster. \( \hat{x}_k^i \) is the spectrum of the \( i \)th pixel in \( \hat{X}_k \), which is filtered as

\[
\hat{\hat{x}}_k^i = \sum_{j \neq i} w_{ij} \hat{x}_k^j, \quad w_{ij} = \frac{1}{W_i} \exp \left( -\frac{\| \hat{x}_k^i - \hat{x}_k^j \|^2}{2h} \right) \tag{19}
\]

where \( \hat{x}_k^i \) denotes the filtered spectrum of the \( i \)th pixel. \( W_i = \sum_{j \neq i} w_{ji} \) is a normalization factor and \( h \) is a predetermined scalar. It can be seen that spectrum \( \hat{x}_k^i \) will be allocated a larger weight when it is more similar to \( \hat{x}_k^j \). With this filtering scheme, the unnatural spatial appearance produced by the structured sparse coding can be properly restored, as shown in the second row of Fig. 3(b)–(d). Moreover, the filtering scheme is able to remove the non-Gaussian noise (e.g., stripe noise) in real noisy HSIs, which will be illustrated in Section V-B. The proposed intracluster filtering scheme is different with regard to NLM [14], which employs the whole image to restore each pixel. In the proposed method, the spectrum of each pixel is restored only by the spectra of pixels in the same cluster, which results in an efficient algorithm.

IV. OPTIMIZATION AND FULL PROCEDURE

In this section, we first describe the optimization for (18), and then provide the full procedure of our newly developed denoising algorithm. Since the optimization problem in (18) involves several unknown variables, it is difficult to optimize it directly. Inspired by [27] and [26], we utilize the alternative minimization scheme [27] to reduce the original problem in (18) to several subproblems, each of which involves only one unknown variable. We name these subproblems sparse coding, graph structure learning, and noise estimation. Then, those subproblems are alternatively optimized till convergence.

A. Sparse Coding: Optimization of \( Y \)

By removing those irrelevant terms, we obtain the subproblem of \( Y \) as

\[
\min_Y \|DY - F\|_{\Sigma_n}^2 + \|Y\|_{\Sigma_y}^2. \tag{20}
\]

Given \( \Sigma_y \) and \( \lambda \), this quadratic optimization problem has a closed-form solution as

\[
Y = \Sigma_y D^T (\Sigma_n + D \Sigma_y D^T)^{-1} F. \tag{21}
\]

B. Graph Structure Learning: Optimization of \( \Sigma_y \)

Given \( Y \) and \( \lambda \), the subproblem of \( \Sigma_y \) can be written as

\[
\min_{\Sigma_y} \|Y\|_{\Sigma_y}^2 + n_p \log |\Sigma_n + D \Sigma_y D^T|. \tag{22}
\]
Since this problem is nonconvex, it is intractable to solve it directly. We turn to find a strict upper bound of the cost function in (22). According to the algebraic relation

\[ \| \Sigma_n + D \Sigma_y D^T \| = \| \Sigma_n \| + \| \Sigma_y \| + \| D^T \Sigma_n^{-1} D + \Sigma_y^{-1} \| \]  

we can further simplify (22) as

\[ \min_{\Sigma_y} \| Y \|_{\Sigma_y} + n_p \log \| \Sigma_y \| + n_p \log | D^T \Sigma_n^{-1} D + \Sigma_y^{-1} |. \]

(24)
Letting $f(\Sigma^{-1} y) = \log |D^T \Sigma^{-1} D + \Sigma^{-1}|$, we have the following relationship:

$$f(\Sigma^{-1} y) = \log |D^T \Sigma^{-1} D + \Sigma^{-1}| \leq \text{tr}(Z^T \Sigma^{-1} D + \Sigma^{-1})^{-1}.$$  \hspace{1cm} (25)

where $f^*(Z)$ is the concave conjugate function of $f(\Sigma^{-1} y)$ with an intermediate variable $Z$. It can be proved that the equality of the right part of (25) holds iff

$$Z = \nabla_{\Sigma^{-1} y} f(\Sigma^{-1} y) = (D^T \Sigma^{-1} D + \Sigma^{-1})^{-1}.$$  \hspace{1cm} (26)

Introducing $Z$ and the upper bound in (25), (24) can be finally reduced to

$$\min_{\Sigma, y} \|Y\|_\Sigma^2 + n_p \log |\Sigma| + n_p \text{tr}(Z^T \Sigma^{-1} D + \Sigma^{-1})^{-1}$$  \hspace{1cm} (27)

which is a convex optimization of $\Sigma, y$ and thus have a closed-form solution as

$$\Sigma = n_p^{-1} Y Y^T + Z.$$  \hspace{1cm} (28)

C. Noise Estimation: Optimization of $\lambda$

Given $Y$ and $\Sigma$, we have the following subproblem on $\lambda$:

$$\min_{\lambda} \|DY - F\|_\Sigma^2 + n_p \log |\Sigma| + D \Sigma y D^T.$$

To optimize this nonconvex problem, we try to find a strict upper bound of the cost function in (29) similar to the graph structure learning procedure. Let $\varphi(\lambda) = \log |\Sigma + D \Sigma y D^T|$. We obtain the following relationship:

$$\varphi(\lambda) = \log |\Sigma + D \Sigma y D^T| \leq \beta^T \lambda - \varphi^*(\beta)$$  \hspace{1cm} (30)

where $\varphi^*(\beta)$ is the concave conjugate function of $\varphi(\lambda)$ with an intermediate variable $\beta$. It can be proved that the equality of the right part of (30) holds iff

$$\beta = \nabla_{\lambda} \varphi(\lambda) = \text{diag}[(\Sigma_n + D \Sigma y D^T)^{-1}].$$  \hspace{1cm} (31)

Given $\beta$ and the upper bound in (30), (29) can be reduced to a convex optimization as

$$\min_{\lambda} \|DY - F\|_\Sigma^2 + n_p \beta^T \lambda$$  \hspace{1cm} (32)

which has a closed-form solution for each component $\lambda_i$ as

$$\hat{\lambda}_i = \sqrt{\frac{q_i}{n_p \beta_i}}.$$  \hspace{1cm} (33)

$q_i$ is the $i$th component of the vector $q = \text{diag}(Q Q^T)$ with $Q = DY - F$ and $\beta_i$ is the $i$th component of $\beta$.

To further improve the reconstruction accuracy, a principal component analysis (PCA) dictionary $D$ is learned from the reconstructed $\hat{X}$ after solving (18). The PCA dictionary indicates that $D$ consists of the principle components of PCA. Specifically, considering all spectra (columns) in $\hat{X}$ as samples, PCA is conducted...
Fig. 8. Visual comparison on the denoised 50th band of Wdc from different denoising methods. (a) Original band. (b) Noisy band. (c) NLM3D. (d) BM4D. (e) PARAFAC. (f) LRTA. (g) TensorDL. (h) Sparsity + Filter.

Algorithm 1. Structured Sparse Coding-Based Hyperspectral Imagery Denoising With Intracluster Filtering

**Input:** Noisy observation $F$, initialized dictionary $D$, covariance matrix $\Sigma_y$, and noise level $\lambda$.

**while** Outer stopping criteria are not satisfied **do**

**while** Inner stopping criteria are not satisfied **do**

1. Update $\Sigma_m = \Sigma_m + D \Sigma_y D^T$;
2. Update sparse codes $Y$ as (21);
3. Update intermediate variable $Z$ as (26);
4. Learn the graph structure $\Sigma_y$ as (28);
5. Update intermediate variable $\beta$ as (31);
6. Estimate the noise level $\lambda$ as (33);
7. Reconstruct the HSI as $\hat{X} = DY$;
8. Learn the PCA dictionary $D$ from $\hat{X}$;

9. Divide the spectra of $\hat{X}$ into $K$ clusters with K-means;
10. Filter the reconstructed HSI $\hat{X}$ as (19);

**Output:** Denoised HSI $X_{rec}$.

on those samples to optimize $n_b$ principle components $d_i \in \mathbb{R}^{n_b}$, which form the dictionary $D = \{d_1, \ldots, d_{n_b}\}$. With such a PCA dictionary, each spectrum in the latent HSI $X$ will be approximately sparsely represented. A similar PCA dictionary had been employed for image reconstruction in various applications [10], [37]. With the learned PCA dictionary $D$, (18) is solved again to refine $\hat{X}$. The dictionary learning and optimization in (18) can be conducted iteratively until convergence. Then, the intracluster filtering as (19) is employed on the reconstructed HSI $\hat{X}$ to obtain the finally denoised result. The entire denoising procedure is summarized in Algorithm 1.

V. EXPERIMENTS AND ANALYSIS

In this section, we evaluate the denoising performance of the proposed method using both simulated and real HSIs. The proposed method (denoted by ‘Sparsity+Filter’) is compared with five state-of-the-art HSI denoising methods, including NLM3D [15], BM4D [17], LRTA [38], PARAFAC [13], and TensorDL [8]. All of them are implemented with the codes published by the authors. Their corresponding parameters have been optimized for the best performance. For the proposed method, the spectrum dictionary $D$ is initialized by an over-complete discrete cosine transformation dictionary, where $n_d$ is set as $4 \times n_b$ as suggested in [39]. The noise level $\lambda$ and covariance matrix $\Sigma_y$ are initialized as a vector with all ones and an identity matrix, respectively. In Algorithm 1, the inner loop is terminated when either the maximum iteration number $N_{\text{max}} = 300$ or the minimum update difference $\eta_{\text{min}} = 10^{-3}$ is reached. The update difference is defined as
Fig. 9. Visual comparison on the denoised 120th band of Wdc from different denoising methods. (a) Original band. (b) Noisy band. (c) NLM3D. (d) BM4D. (e) PARAFAC. (f) LRTA. (g) TensorDL. (h) Sparsity+Filter.

Fig. 10. Spectral reflectance difference curves of all methods on two HSIs. (a) HSI with three marked positions. (b)–(d) Curves corresponding to those three marked positions in each HSI. (First row) Curves on PaviaU. (Second row) Curves on Wdc.

\[ \frac{\|Y_{\text{new}} - Y\|_F}{\|Y\|_F}, \] where \(Y_{\text{new}}\) and \(Y\) denote the updated sparse codes in the current and last iterations. In the outer loop, we run only one round of the reconstruction with the leaned PCA dictionary. The K-means++ algorithm [36] is adopted to cluster pixels in the reconstructed HSI. The cluster number \(K = 30\) and the scalar \(h = 0.02\) are adopted intracluster filtering. For simplicity, those settings are fixed in the following experiments.

To comprehensively assess the denoising performance of all methods, three HSI quality evaluation indexes are adopted in
this paper, which are the peak signal-to-noise ratio (PSNR), structure similarity (SSIM), and spectral angle mapper (SAM) [8]. PSNR and SSIM are two commonly utilized image quality indexes, which measure the similarity between two considered images based on the mean squared error and spatial structure discrepancy, respectively. Different from PSNR and SSIM, SAM is specialized in measuring the spectrum similarity of HSIs through calculating the average angle between corresponding spectrum vectors at the same position from those two considered HSIs. In the simulated denoising experiments, those three indexes are calculated for each method. Moreover, larger PSNR and SSIM and smaller SAM demonstrate better denoising performance.

A. Experiments on Simulated Noisy HSIs

In this experiment, part of ROSIS image of Pavia University and HYDICE image of Washington DC Mall are utilized as the experimental data. Their 3-D data cubes are shown in Fig. 4. For simplicity, we term those two HSIs PaviaU and Wdc. The size of PaviaU is 200 × 200 × 103, while the size of Wdc is 200 × 200 × 191. Each HSI \( X \) is normalized between \([0, 1]\) before the simulation process. In the simulation process, different levels of Gaussian white noise are added across bands of \( X \) to obtain the simulated noisy HSI \( F \), and the resulted signal-to-noise ratio of each band image is uniformly distributed in the range from 10 to 30 dB. Given the noisy HSI \( F \), all denoising methods are conducted to reconstruct the clean \( X \). To meet the requirements of BM4D and TensorDL, the real standard deviation of noise calculated between the ground truth \( X \) and the noisy observation \( F \) is provided as the input parameters for them.

1) Comparison With Other Competitors: For each method, Fig. 5(a) and (b) plots their curves of PSNR and SSIM values on each band of two experimental HSIs, while Fig. 5(c) gives the curves of the SAM values on the first 200 pixels. On PaviaU, the PSNR and SAM values of the proposed method (denoted by “ours”) are higher than that of other
Fig. 13. Effect of cluster number on the denoising performance of the proposed method.

Fig. 14. Effect of parameter $h$ on the denoising performance of the proposed method.

Fig. 15. 3-D cubes of (a) Indiana and (b) Urban with partial bands.

The proposed method outperforms its competitors in most cases, while the SSIM curve of the proposed method stays above 0.9 almost in all bands, which is the highest among all methods. NLM3D, BM4D, and PARAFAC produce relatively stable curves only with small fluctuations across bands, while TensorDL and LRTA perform fluctuantly in an obvious way. On Wdc, all comparison methods perform fluctuantly in terms of SSIM and SAM, while the proposed method produces much better and more stable results. Although the proposed method also performs fluctuantly in PSNR as others, it still outperforms others in most bands. Therefore, we can conclude that the proposed method performs more stably and better than other competitors on simulated noisy HSIs. To further clarify this point, the corresponding average numerical results on PSNR, SSIM, and SAM are given in Table I. Compared with the second runner BM4D, the proposed method improves the PSNR by 3.84 dB on PaviaU and 3.42 dB on Wdc, while the corresponding SAM values decrease 1.22° on PaviaU and 0.72° on Wdc. In addition, it can be seen that the proposed method is the only one whose average SSIM values on two HSIs are larger than 0.9.

TABLE I

<table>
<thead>
<tr>
<th>Method</th>
<th>PSNR</th>
<th>SSIM</th>
<th>SAM</th>
<th>PSNR</th>
<th>SSIM</th>
<th>SAM</th>
</tr>
</thead>
<tbody>
<tr>
<td>NLM3D</td>
<td>28.0317</td>
<td>0.7489</td>
<td>5.9968</td>
<td>51.0143</td>
<td>0.8212</td>
<td>4.2619</td>
</tr>
<tr>
<td>BM4D</td>
<td>31.8579</td>
<td>0.8009</td>
<td>5.3051</td>
<td>34.2876</td>
<td>0.8309</td>
<td>3.4670</td>
</tr>
<tr>
<td>PARAFAC</td>
<td>28.7889</td>
<td>0.7800</td>
<td>6.7460</td>
<td>27.9469</td>
<td>0.7834</td>
<td>9.1820</td>
</tr>
<tr>
<td>LRTA</td>
<td>20.1667</td>
<td>0.4960</td>
<td>25.7895</td>
<td>21.9832</td>
<td>0.4212</td>
<td>18.4386</td>
</tr>
<tr>
<td>TensorDL</td>
<td>19.8981</td>
<td>0.5287</td>
<td>26.5089</td>
<td>21.1854</td>
<td>0.4197</td>
<td>20.1141</td>
</tr>
<tr>
<td>Sparsity+Filter</td>
<td>35.7066</td>
<td>0.9470</td>
<td>4.0842</td>
<td>37.7076</td>
<td>0.9729</td>
<td>2.7428</td>
</tr>
</tbody>
</table>

To further clarify the superiority of the proposed method in terms of denoising, we compare the denoised results of all methods visually. Figs. 6 and 7 provide the denoised 58th and 13th bands of PaviaU by all methods, where two interested areas are zoomed for details’ comparison. From Figs. 6(b) and 7(b), we can find that those two selected bands are corrupted with different levels of noise. NLM3D and BM4D remove the noise corruption very well but oversmooth the spatial structures, e.g., road across the parking zone and the sharp building edges, which can be seen in the zoomed-in areas in Figs. 6(c) and (d) and 7(c) and (d). For PARAFAC, LRTA, and TensorDL, obvious noise still remains in their results. In contrast, the proposed method not only effectively removes the noise corruption but also restores the spatial structure well. Specifically, from the zoomed-in areas in Figs. 6(h) and 7(h), we can see that the road across the parking zone and the sharp building edges are well restored by the proposed method. Similarly, the proposed method also produces sharper and cleaner results for the 50th and 120th bands of Wdc than other methods, shown in Figs. 8 and 9. Especially for the 120th band in Fig. 9 where the spatial structures in the noisy band...
image are severely corrupted, all competitors fail to produce a relatively clean image and obvious noise remains in their denoised results. However, the proposed method still produces clean image with well-restored spatial structure shown as those zoomed-in areas in Fig. 9(h), which show the robustness of the proposed method to strong noise.

In addition, we plot the spectral reflectance difference curves for each method in Fig. 10 to illustrate the denoising performance of the proposed method on spectrum recovery [8]. The reflectance difference curve of each method is interpolated by the discrete reflectance difference between the spectrum vector from the denoised HSI and that from the ground truth at the given pixel. We select three pixels centered at the colored window for each HSI, shown in Fig. 10(a). To show a stable result, the average reflectance difference within a 3 × 3 neighboring area is adopted for each selected pixel. Those obtained reflectance difference curves are given in Fig. 10(b)–(d). It can be seen that the average reflectance difference of the proposed method is the closest to zero among all methods. This demonstrates that the proposed method reconstructs the spectrum more accurately than other competitors.

From the above results, we find that the proposed method performs better than others in denoising HSIs. There are two reasons. First, the proposed method unifies the structured sparse coding on spectrum domain and the intracluster filtering together, which simultaneously models the spectral correlation and spatial local and nonlocal similarity. In addition, the general noise model in (6) adapts the proposed method to various levels of noise across bands.

2) Effectiveness of the Proposed Method: The proposed method contains two crucial ingredients, namely, the structured sparse coding and the intracluster filtering. It is natural to validate whether those two parts are both crucial for good denoising performance. To this end, we compare the proposed method with its two variants on denoising the noisy PaviaU and Wdc, where the noisy HSIs are obtained in the same way as above. The first variant is termed Sparsity, which employs only the structured sparse coding, while the second variant termed Filter utilizes only the intracluster filtering for denoising. Similarly, we give the PSNR and SSIM curves of those three methods across bands of two HSIs in Fig. 11. We can find that Sparsity and the proposed method perform much better than Filter, while the proposed method stably outperforms Sparsity across bands of two HSIs. To further illustrate this point, the bar charts of the average numerical results of those three methods are given in Fig. 12 where we can obtain a similar conclusion. Specifically, Filter with intraclustering filtering alone produces the poorest results among those three methods, e.g., the obtained PSNR values on both PaviaU and Wdc are smaller than 20 dB. This is because the basic clustering procedure in intraclustering filtering is often
misled by the noisy spectra. Thus, its denoising performance is restricted accordingly. Although Sparsity with structured sparse coding alone can well reconstruct the spectrum and gives much better results than that of Filter, e.g., compared with Filter, Sparsity improves the PSNR by 15 dB on PaviaU and 16 dB on Wdc, the inconsistent recovery of the sparse codes caused by modeling them independently often corrupts the intracluster spectral similarity, which limits its denoising performance, e.g., the PSNR of Sparsity on Wdc is smaller than the proposed method by 3 dB. When fusing the structured sparse coding and intraclustering filtering, the proposed method gives the best results among all three methods. This is because the spectra reconstructed by the structured sparse coding help to improve the clustering accuracy for a better filtering and, in turn, the filtering is able to restore the intracluster spectral similarity corrupted by the structured sparse coding. Therefore, we can conclude that both the structured sparse coding and the intracluster filtering steps are crucial to denoising.

First, we test the performance of the proposed method with different cluster numbers, e.g., 1, 5, 10, 20, 30, 50, and 100, and the parameter $h$ is fixed at 0.02. The resulted PSNR, SSIM, and SAM are shown in Fig. 13. We can find that the performance is roughly insensitive to the cluster number and slightly decreases with the increasing cluster number. The reason is intuitive. According to 19, filtering results mainly depend on the top-$k$ (e.g., $k \leq 10$) similar pixels to the considered one. When cluster number varies from 1 to 100, those top-$k$ pixels often can be collected into the same cluster as the considered one. Thus, the denoising performance can be roughly preserved. However, when the cluster number increases, pixels that are slightly similar to the considered one may be collected into different clusters and their contributions to the filtering results are eliminated. Thus, the denoising performance slightly decreases. In an extreme case where the cluster number reaches the total number of pixels in an HSI, namely, each pixel is collected into an individual cluster, the filtering step is infeasible and the proposed method degenerates to its variant “Sparsity.” While the performance of the proposed method is roughly insensitive to the cluster number, the computational cost is obviously different. When the cluster number is reduced, the amount of pixels in each cluster will increase, and the filtering step will cost more on computing the weights in 19, especially when the cluster number is 1. To balance the computational cost and the performance, we fix the cluster number.
number at 30 in all experiments. In addition, the adopted K-means++ algorithm [36] is able to automatically determine the initialized cluster centers for K-means based on the data distribution. Therefore, the proposed method is insensitive to the initialization of the K-means.

Second, we conduct similar experiments with varying parameter $h$ (e.g., 0.001, 0.005, 0.01, 0.02, 0.03, 0.05, and 0.1) in the filtering step. The cluster number is fixed at 30. We provide the corresponding results in Fig. 14. It can be seen that there exists an optimal $h^*$ for each evaluation index. When $h > h^*$ or $h < h^*$, the performance decreases. To give the best average performance in three evaluation indices, we set $h = 0.02$ for all experiments in the main manuscript.

For simplicity, the cluster number and $h$ are fixed for all data sets. Of course, better results can be obtained by choosing the best parameters for each data set.

B. Experiments on Real HSIs

In this section, two real HSIs are utilized to test the proposed method, which are a HYDICE Urban image and the well-known AVIRIS Indian Pines image. Hereinafter, we refer to them as Urban and Indiana for simplicity. In the following experiments, the whole image of Indiana of size $145 \times 145 \times 220$ and part of Urban of size $200 \times 200 \times 210$ are employed as the experimental data. Their 3-D data cubes are shown in Fig. 15. It has been shown that both HSIs contain some noisy bands, e.g., bands 1–4, bands 103–112, and bands 217–220, in Indiana [40]. More importantly, the noise levels are various across bands and the noise may be beyond Gaussian-like distribution. For example, the 109th and 219th bands in Indiana are obviously corrupted with different levels of Gaussian-like noise shown in Figs. 16(a) and 17(a), while the 104th and 207th bands in Urban are affected by two different levels of non-Gaussian noise (e.g., stripe noise) shown in Figs. 18(a) and 19(a). In the denoising process, those two noisy HSIs are directly employed as the noisy observation $F$ for the proposed method and other five competitors.

For the Indiana scene, the denoised results of the 109th and 219th bands from all methods are visually compared in Figs. 16 and 17, where two areas of interest are zoomed for a detailed comparison. We can observe that the 109th and 219th bands are corrupted by different levels of Gaussian-like random noise. NLM3D, LRTA, and TensorDL fail to remove the noise corruption. Although BM4D and PARAFAC remove the noise corruption to some extent, they both oversmooth the sharp edges. Moreover, BM4D produces some artifacts, e.g., the undesired stripes seen as the zoomed-in areas in Figs. 16(c) and 17(c), while PARAFAC exhibits the blocking effect shown as the zoomed-in areas in Figs. 16(d) and 17(d). Compared with those competitors, the proposed method properly removes the noise corruption and well restores the sharp edges, shown as the zoomed-in areas in Figs. 16(g) and 17(g).
Fig. 19. Visual comparison on the denoised 207th band of Urban from different denoising methods. (a) Original band. (b) NLM3D. (c) BM4D. (d) PARAFAC. (e) LRTA. (f) TensorDL. (g) Sparsity + Filter.

For the Urban scene, we observe that the original noisy bands contain at least two different types of noises. One is Gaussian-like noise and the other is the non-Gaussian stripe noise in Figs. 18(a) and 19(a). From Figs. 18(b)–(d) and 19(b)–(d), we can find that NLM3D, BM4D, and PARAFAC perform well in removing the Gaussian-like noise, but fail on the stripe noise. While LRTA and TensorDL fail in removing both two kinds of noises shown in Figs. 18(e) and (f) and 19(e) and (f), the proposed method well removes both the Gaussian-like noise and stripe noise. In addition, the sharp edges are properly preserved with the proposed method, shown as the zoomed-in areas in Figs. 18(g) and 19(g). Although the structured sparse coding is derived based on the Gaussian noise assumption, the intracluster filtering procedure can exploit the spatial similarity to remove the non-Gaussian noise. According to the results obtained for the two considered HSIs, we can conclude that the proposed method is effective in denoising real HSIs.

VI. CONCLUSION

In this paper, we present a new denoising method for HSIs that combines structured sparse coding and intracluster filtering. By representing the image as a group of sparse codes on a given spectrum dictionary, we model all sparse codes together, which casts the structured sparse coding into a covariance matrix estimation problem. Then, a latent variable-based Bayesian framework is utilized to robustly capture the data-dependent structured sparsity within each spectrum under unknown noise, where the covariance matrix, sparse codes, and noise level can be jointly learned from the noisy observations. To restore the corrupted spectrum similarity in each spatial homogeneous cluster within the HSI, we further employ an intracluster filtering scheme on the reconstructed image from the structured sparse coding. Our experiments using both simulated and real images reveal that the proposed method is comparable or better than several state-of-the-art denoising methods in the literature.

REFERENCES


Antonio Plaza (M’05–SM’07–F’15) received the M.Sc. and Ph.D. degrees in computer engineering from the University of Extremadura, Badajoz, Spain, in 1999 and 2002, respectively. He is currently the Head with the Hyperspectral Computing Laboratory, Department of Technology of Computers and Communications, University of Extremadura. He has authored more than 600 publications, including 197 JCR journal papers (over 140 in IEEE journals), 23 book chapters, and 285 peer-reviewed conference proceeding papers. His research interests include hyperspectral data processing and parallel computing of remote sensing data.

Prof. Plaza was a member of the Editorial Board of the IEEE GEOSCIENCE AND REMOTE SENSING NEWSLETTER (2011–2012) and the IEEE GEO-SCIENCE AND REMOTE SENSING MAGAZINE (2013). He was also a member of the Steering Committee of the IEEE JOURNAL OF SELECTED TOPICS IN APPLIED EARTH OBSERVATIONS AND REMOTE SENSING (JSTARS). He was a recipient of the recognition of the Best Reviewers for the IEEE GEOSCIENCE AND REMOTE SENSING LETTERS (in 2009) and the IEEE TRANSACTIONS ON GEOSCIENCE AND REMOTE SENSING (in 2010), for which he served as an Associate Editor in 2007–2012. He was also a recipient of the Best Column Award of the IEEE SIGNAL PROCESSING MAGAZINE in 2015, the 2013 Best Paper Award of the JSTARS Journal, the most highly cited paper (2005–2010) in the Journal of Parallel and Distributed Computing, and the Best Paper Awards at the IEEE International Conference on Space Technology and the IEEE Symposium on Signal Processing and Information Technology. He was a Guest Editor of 10 special issues on hyperspectral remote sensing for different journals. He is an Associate Editor of the IEEE ACCESS. He served as the Director of Education Activities for the IEEE Geoscience and Remote Sensing Society (GRSS) from 2011 to 2012 and the President of the Spanish Chapter of the IEEE GRSS in 2012–2016. He was a reviewer for more than 500 manuscripts for over 50 different journals. He is currently serving as the Editor-in-Chief of the IEEE TRANSACTIONS ON GEOSCIENCE AND REMOTE SENSING.

Yanning Zhang (SM’12) received the B.S. degree from the Dalian University of Technology, Dalian, China, in 1988, the M.S. degree from the School of Electronic Engineering, Northwestern Polytechnical University, Xi’an, China, in 1993, and the Ph.D. degree from the School of Marine Engineering, Northwestern Polytechnical University, Xian, China, in 1996.

She is currently a Professor with the School of Computer Science, Northwestern Polytechnical University. Her research interests include computer vision and pattern recognition, image and video processing, and intelligent information processing.

Dr. Zhang was the Organization Chair of the Asian Conference on Computer Vision 2009 and served as the Program Committee Chair of several international conferences.