Low-Rank Tensor Approximation with Laplacian Scale Mixture Modeling for Multidimensional Image Denoising

Anonymous ICCV submission

Paper ID 987

Abstract

The patch-based low-rank approximation model has shown to be very effective in exploiting the spatial redundancy of natural images and achieves impressive image denoising performance. However, the two-dimensional low-rank model can not fully exploit the correlations among multidimensional data, such as multispectral images and dynamic MRI image sequences. To effectively exploit the multidimensional correlations for multidimensional data, we propose a novel low-rank tensor approximation model with Laplacian Scale Mixture (LSM) modeling. Specifically, similar multidimensional patches are first grouped to form a tensor of d*-order and the the high order Singular Value Decomposition (HOSVD) is then applied to the resulted tensor. The resulting coefficients array are modeled with the LSM distribution. The sparse estimation problem is then formulated as a maximum a Posterior (MAP) estimation problem with the LSM prior. We show that both the sparse coefficients array and the scalar variables can be efficiently estimated via alternative optimization. Specifically, both subproblems admit closed-form solutions. Experimental results on spectral images and 3D MRIs show that the proposed denoising algorithm can well preserve the edge sharpness and substantially outperforms the current state-of-the-art image denoising methods.*

1. Introduction

The past decade has witnessed a considerable progress in the field of image denoising. Substantial advanced image denoising algorithms have been proposed. The sparse representation based methods [19, 1], especially combined with dictionary learning [8, 15, 24], have shown the popularity and effectiveness in removing the noise. Combined with another popular prior of natural images, i.e., the nonlocal self-similarity [3], the denoising performance of the sparsity-based methods can be significantly improved. Research along this line has led to the success of learned simultaneous/structured sparse coding methods [5, 14, 6]. Moreover, the recent development of the two-dimensional low-rank matrix approximation techniques have also motivated the patch-based nonlocal low-rank image denoising methods [7, 10], which are among the current state-of-the-art denoising methods.

For multidimensional images, directly applying the popular sparse and low-rank denoising methods to each band or frame separately fail to exploit the correlations across the third dimension, leading to unsatisfied results. Another more effective extension is to use the multidimension patches. By representing the multidimension patches into a very high-dimension 1-dimensional (1D) image vector, the sparsity and low-rank methods can then be applied to the multidimensional data. However, due to the very large size of the vector, e.g., $5 \times 5 \times 30 = 750$ for a multispectral image (MSI) consisting of 30 spectral bands, it is difficult to train a very large dictionary or construct a low-rank matrix due to the lack of enough similar samples.

In this paper, we propose a high order low-rank approximation method with Laplacian Scale Mixture (LSM) modeling for multidimensional image denoising, which generalized the popular nonlocal low-rank matrix approximation method to multidimensional data. First, overlapping 3D patches are extracted from the input volumetric data. Then, for each exemplar 3D patch, a set of similar 3D patches are grouped. As the group of 3D patches contain similar structures, they can be well approximated by a low-rank ”tensor”. The high order SVD (HOSVD) technique is used for the low-rank approximation. By thresholding the resulting coefficient array, the noise can be effectively removed. Instead of choosing the shrinkage function manually, we propose to use the Laplacian Scale Mixture distribution to model the coefficient array. The sparse coefficients estimation is then formulated as a Maximum a Posterior (MAP) estimation problem. We show that both the sparse coefficients and the scalar variables can be jointly estimated via a alternative optimization. Experimental results show that the proposed HOSVD method substantially outperforms the current state-of-the-art volumetric data denoising methods, e.g., the recent tensor dictionary learning method [18] and the BM4D method [13].

2. Related Works

In this section, we will briefly review the related sparse and low-rank methods, as well as some recently developed tensor based image denoising methods.

The sparse methods exploit the fact that natural image patches can be well approximated by a linea combination of a small set of atoms from a dictionary. Instead of using the off-the-shelf dictionaries, it has been shown that adapting to the local image structures via dictionary learning can substantially improve the denoising performance [8, 15, 24]. The sparse methods can become even more effectively by considering the nonlocal self-similarity [3] between the similar patches [5, 14, 6]. For volumetric images, processing each band/frame separately obviously ignores the rich correlations across the third dimension. A better extension is to use the 3D patches. By representing the 3D patches as high-dimension 1D vectors, existing sparse methods can be used. However, for the volumetric images containing a number of bands/frames, the dimension of the 1D vectors will become too high to find enough samples to learn a large dictionary, leading to the decrease of the denoising performance.

The low-rank methods recover the clean images by low-rank matrix approximation [7, 10]. Similar image patches are first grouped for each exemplar patch to form a data matrix $\mathbf{Y}$. As each patch contains similar structures, the rank of $\mathbf{Y}$ is low. Then, the noiseless data matrix can be accurately reconstructed via singular value threholding $\hat{\mathbf{X}} = \mathbf{U}\mathcal{S}_\tau(\mathbf{\Sigma})\mathbf{V}^\top$, where $\mathbf{U}_i\mathbf{\Sigma}_i\mathbf{V}_i^\top$ is the SVD of $\mathbf{Y}_i$. By designing an appropriate shrinkage function $\mathcal{S}_\tau(\cdot)$, sate-of-the-art image denoising performances have been achieved [7, 10]. For volumetric data, a straightforward extension of the low-rank methods is to use 3D patches. By grouping similar 3D patches, we can also form the data matrix $\mathbf{Y}$, where each column of $\mathbf{Y}$ corresponds to the 1D vector representation of the 3D patch. Then, the volumetric data can also be reconstructed via singular value thresholding. However, this doesn't mean the noiseless matrix can be accurately estimated as in the case of natural images. The reason is that in the low-rank matrix reconstruction the left singular vectors $\mathbf{U}$ and the right singular vectors $\mathbf{V}$ are statistically determined by the covariance matrix $\mathbf{Y}^\top\mathbf{Y}$ and $\mathbf{Y}\mathbf{Y}^\top$, respectively. Since the dimension of the column vectors of $\mathbf{Y}$ is very high for volumetric data, it is difficult to estimate the covariance matrixes accurately due to the lack of sufficient similar samples. Consequently, the denoising performance of the low-rank method will be decreased.

The tensor methods have also been proposed for volumetric data denoising. In [21, 12], by treating the whole multispectral image (MSI) as a tensor, the low-rank tensor approximation methods have been proposed for MSI denoising. These methods can fully exploit correlations across the spectral bands. However, they ignore the rich nonlocal repetitive structures among MSI. Recently, Peng *et al*. [18] proposed an effective MSI denoising method using nonlocal tensor dictionary learning. To exploit the nonlocal redundancy, the 3D MSI patches are clustered into many clusters via k-means clustering. Each set of similar 3D patches are then linearly approximated by low-rank tensor approximation. Specifically, the AIC/MDL criteria [22] is used to determine the ranks for each model of the tensor. The HOSVD has also been used for natural image denoising [20], where the similar patches are stacked into a 3D array and the HOSVD is applied for low-rank tensor approximation. Similar to BM3D method, the coefficient array is first processed with a hard thresholding followed by the Wiener filtering in the second denoising stage. Our proposed low-rank tensor approximation differs from both the methods of [18, 20] in that an adaptive sparse estimation is developed for the estimation of the coefficient array using the Laplacian Scale Mixture distribution. Experimental results show that the proposed method performs substantially better than the current state-of-the-art methods, i.e., [13, 18].

3. Low-rank Tensor Approximation with Laplacian Scale Mixture Modeling

In this section, we first introduce the low-rank tensor approximation method for multidimensional image denoising, and then present the proposed Laplacian Scale Mixture Modeling for nonlinear low-rank tensor approximation.

3.1. Nonlocal low-rank tensor approximation

Nonlocal low-rank based image denoising consists of two steps: patch grouping and low-rank approximation. For a noisy 3D image of size $H \times W \times L$, 3D patches are extracted. For each exemplar 3D patch $\mathcal{P}_i$ of size $\sqrt{n} \times \sqrt{n} \times L$ extracted at spatial position i, we search for the similar patches via the k-nearest neighbor (k-NN) search in a large window (e.g., 40×40), i.e.,

$$G_i = \{i_j | \|\mathcal{P}_i - \mathcal{P}_{i_j}\| < T\}, \tag{1}$$

where T is the predefined threshold and G_i denotes the collection of the positions of the similar patches. Alternatively, we can also form G_i by selecting the patches that are within the first m closest to $\mathcal{P}_i$ (including $\mathcal{P}_i$ itself). After patch grouping, we can combine the similar 3D patches into a 3^{rd} order tensor by representing the matrix slices of each 3D patch into vectors, i.e., $\mathcal{Y}_i \in \mathbb{R}^{n \times m \times L}$[1]. Given the noisy

[1] Instead of forming a 4^{th} order tensor for the set of similar 3D patches, we found that combining them into a 3^{rd} order tensor leads to better denoising performance.

tensor $\mathcal{Y}_i$, its HOSVD is given as follows [11, 2],

$$\begin{aligned}\mathcal{Y}_i &= \sum_{r=1}^{n}\sum_{c=1}^{m}\sum_{l=1}^{L}\tilde{\mathcal{S}}_i(r,c,l)\boldsymbol{u}_{i,r}\times\boldsymbol{v}_{i,c}\times\boldsymbol{w}_{i,l}\\ &= \tilde{\mathcal{S}}_i\times_1\mathbf{U}_i\times_2\mathbf{V}_i\times_3\mathbf{W}_i,\end{aligned} \tag{2}$$

where $\mathbf{U}_i = [\boldsymbol{u}_{i,1},\cdots,\boldsymbol{u}_{i,n}] \in \mathbb{R}^{n\times n}$, $\mathbf{V}_i = [\boldsymbol{v}_{i,1},\cdots,\boldsymbol{v}_{i,m}] \in \mathbb{R}^{m\times m}$ and $\mathbf{W}_i = [\boldsymbol{w}_{i,1},\cdots,\boldsymbol{w}_{i,L}] \in \mathbb{R}^{L\times L}$ are orthogonal matrixes, $\tilde{\mathcal{S}}_i \in \mathbb{R}^{n\times m\times L}$ is the 3D coefficient array (also called core tensor), $\tilde{\mathcal{S}}_i(r,c,l)$ are the components of $\tilde{\mathcal{S}}_i$, $\times$ denotes the tensor product, i.e., $\boldsymbol{x}\times\boldsymbol{y} = \boldsymbol{x}\boldsymbol{y}^\top$, and $\times_j$ denotes the $j-th$ model tensor product. The orthogonal matrixes $\mathbf{U}_i$, $\mathbf{V}_i$ and $\mathbf{W}_i$ are computed from the SVD of the model-j ($j=1,2,3$) flattening of $\mathcal{Y}_i$, respectively. Since the similar patches contain similar structures, $\mathcal{Y}_i$ can be approximated with a low-rank tensor, i.e.,

$$\begin{aligned}\hat{\mathcal{X}}_i &= \sum_{r=1}^{r_1}\sum_{c=1}^{r_2}\sum_{l=1}^{r_3}\hat{\mathcal{S}}_i(r,c,l)\boldsymbol{u}_{i,r}\times\boldsymbol{v}_{i,c}\times\boldsymbol{w}_{i,l}\\ &= \hat{\mathcal{S}}_i\times_1\hat{\mathbf{U}}_i\times_2\hat{\mathbf{V}}_i\times_3\hat{\mathbf{W}}_i,\end{aligned} \tag{3}$$

where $\hat{\mathbf{U}}_i = [\boldsymbol{u}_{i,1},\cdots,\boldsymbol{u}_{i,r_1}] \in \mathbb{R}^{n\times r_1}$, $\hat{\mathbf{V}}_i = [\boldsymbol{v}_{i,1},\cdots,\boldsymbol{v}_{i,r_2}] \in \mathbb{R}^{m\times r_2}$ and $\hat{\mathbf{W}}_i = [\boldsymbol{w}_{i,1},\cdots,\boldsymbol{w}_{i,r_3}] \in \mathbb{R}^{L\times r_3}$ are the thin matrices associated with $\mathbf{U}_i$, $\mathbf{V}_i$ and $\mathbf{W}_i$, respectively, $r_1 \le n$, $r_2 \le m$ and $r_3 \le L$, and $\hat{\mathcal{S}}_i \in \mathbb{R}^{r_1\times r_2\times r_3}$ denotes the smaller core tensor. The triple (r_1, r_2, r_3) is called the multirank of $\mathcal{Y}_i$. To estimate the multirank of the tensor, the Akaike's Information Criterion (AIC)/Minimum Description Length(MDL) method has been used for different modes flattening of the tensor [22]. With the estimated rank parameters (r_1, r_2, r_3), the low-rank tensor approximation can be easily obtained by setting the last $n-r_1$, $m-r_2$ and $L-r_3$ slices along the different modes in $\tilde{\mathcal{S}}_i$ to be zero matrices.

Instead of explicitly estimate the multirank parameters, we can also obtain the low-rank tensor approximation by inducing the sparsity on the coefficient array, as

$$\begin{aligned}\hat{\mathcal{S}}_i &= \underset{\mathcal{S}_i}{\operatorname{argmin}}\,\psi(\mathcal{S}_i),\\ &\quad s.\,t.,\ ||\mathcal{Y}_i - \mathcal{S}_i\times_1\mathbf{U}_i\times_2\mathbf{V}_i\times_3\mathbf{W}_i||_F^2 \le \sigma_w^2,\end{aligned} \tag{4}$$

where $\psi(\cdot)$ is a sparse regularization function that induces the sparsity in the components of $\mathcal{S}_i$, $\mathbf{U}_i$, $\mathbf{V}_i$ and $\mathbf{W}_i$ are the orthogonal matrices obtained via HOSVD of $\mathcal{Y}_i$. Due to the orthogonality of the matrices, Eq. (4) can be reexpressed as

$$\hat{\mathcal{S}}_i = \underset{\mathcal{S}_i}{\operatorname{argmin}}\,\psi(\mathcal{S}_i),\ s.\,t.,\ ||\tilde{\mathcal{S}}_i - \mathcal{S}_i||_F^2 \le \sigma_w^2, \tag{5}$$

where $\tilde{\mathcal{S}}_i = \mathcal{Y}_i\times_1\mathbf{U}_i^\top\times_2\mathbf{V}_i^\top\times_3\mathbf{W}_i^\top$. The above problem is often formulated in Lagrangian form,

$$\hat{\mathcal{S}}_i = \underset{\mathcal{S}_i}{\operatorname{argmin}}\,||\tilde{\mathcal{S}}_i - \mathcal{S}_i||_F^2 + \lambda\psi(\mathcal{S}_i). \tag{6}$$

Common choices of $\psi(\cdot)$ include the pseudo-norm ℓ_0 and the ℓ_1 norm, which exactly lead to the hard thresholding and soft thresholding of the coefficients array $\tilde{\mathcal{S}}_i$, respectively. Generally, the selection of the thresholds λ is a non-trial task. For better performance, in [20] a heuristic two-stage method has been proposed for natural image denoising. The hard thresholding is first applied to threshold the coefficient array for initial image denoising, followed by the Wiener filtering in the second stage.

3.2. Laplacian scale mixture modeling for low-rank tensor approximation

Form Eq. (6), we can see that the selection of the sparsity regularization function $\psi(\cdot)$ is critical for the low-rank tensor approximation. In this subsection, we propose a Maximum a Posterior (MAP) estimation method to estimate $\mathcal{S}_i$ from $\tilde{\mathcal{S}}_i$. For simplicity, we will drop the subscript index i and let $\tilde{\boldsymbol{s}} \in \mathbb{R}^{n\cdot m\cdot L}$ and $\boldsymbol{s} \in \mathbb{R}^{n\cdot m\cdot L}$ denote the one-dimensional representations of $\tilde{\mathcal{S}}$ and $\mathcal{S}$, respectively. $\boldsymbol{s}$ is the noiseless version of $\tilde{\boldsymbol{s}}$, i.e., $\tilde{\boldsymbol{s}} = \boldsymbol{s} + \boldsymbol{n}$, where $\boldsymbol{n} \in \mathbb{R}^{n\cdot m\cdot L}$ denotes additive Gaussian noise. The MAP estimation of $\boldsymbol{s}$ from $\tilde{\boldsymbol{s}}$ amounts to solve the following optimization problem

$$\boldsymbol{s} = \underset{\boldsymbol{s}}{\operatorname{argmin}}\{-\log P(\tilde{\boldsymbol{s}}|\boldsymbol{s}) - \log P(\boldsymbol{s})\}, \tag{7}$$

where $\log P(\tilde{\boldsymbol{s}}|\boldsymbol{s})$ is given as the Gaussian distribution, i.e.,

$$P(\tilde{\boldsymbol{s}}|\boldsymbol{s}) \propto \exp(-\frac{1}{2\sigma_w^2}||\tilde{\boldsymbol{s}} - \boldsymbol{s}||_2^2), \tag{8}$$

and *a prior* distribution on $\boldsymbol{s}$ is given with the form

$$P(\boldsymbol{s}) \propto \prod_j \exp(-\frac{\psi(s_j)}{\theta_j}). \tag{9}$$

It is easy to verify that the MAP estimator leads to the following weighted ℓ_1 norm minimization problem when $P(\boldsymbol{s})$ is chosen to be an IID Laplaican prior,

$$\boldsymbol{s} = \underset{\boldsymbol{s}}{\operatorname{argmin}}\,||\tilde{\boldsymbol{s}} - \boldsymbol{s}||_2^2 + 2\sigma_w^2\sum_j \frac{1}{\theta_j}|s_j|, \tag{10}$$

where θ_j denotes the standard derivation of s_j. It has been shown that the weighted ℓ_1 norm is more effective than ℓ_1 norm in sparse estimation [4]. Now the task is how to estimate the variance parameters θ_j. Generally, it is difficult to accurately estimate the variance θ_j for each s_j from the noisy observation $\tilde{\boldsymbol{s}}$.

In this paper, we propose a Laplacian Scale Mixture (LSM) prior to model $\boldsymbol{s}$. With LSM prior, we decompose $\boldsymbol{s}$ into the point-wise product of a Laplacian vector $\boldsymbol{\alpha}$ and a positive hidden scalar multiplier $\boldsymbol{\theta}$ with probability $P(\theta_j)$, i.e., $s_j = \theta_j\alpha_j$, which is analogue to the Gaussian Scale

Mixture [19]. Conditioned on θ_j, s_j is Laplacian with standard devriation θ_j. Assume that θ_j and α_j are independent, the LSM prior of $\boldsymbol{s}$ can be expressed as

$$P(\boldsymbol{s}) = \prod_i P(s_j),\ P(s_j) = \int_0^\infty P(s_j|\theta_j)P(\theta_j)d\theta_j. \tag{11}$$

It should be note that for most choices of $P(\theta_j)$ there is no analytic expression of $P(\boldsymbol{s})$. Thus, it is difficult to compute the MAP estimates of $\boldsymbol{s}$ with the LSM prior. However, such difficulty can be overcome by using the joint prior model $P(\boldsymbol{s}, \boldsymbol{\theta})$. By substituting $P(\boldsymbol{s}, \boldsymbol{\theta})$ into the MAP estimator of Eq. (7), we obtain

$$(\boldsymbol{s}, \boldsymbol{\theta}) = \underset{\boldsymbol{s},\boldsymbol{\theta}}{\operatorname{argmin}}\{-\log P(\tilde{\boldsymbol{s}}|\boldsymbol{s}) - \log P(\boldsymbol{s}|\boldsymbol{\theta}) - \log P(\boldsymbol{\theta})\}. \tag{12}$$

In this paper we adopt a factorial distribution for the multipliers. Specifically, the noninformative Jeffrey's prior, i.e., $P(\theta_j) = \frac{1}{\theta_j}$ is adopted. With this Jeffrey's prior, Eq. (12) can be expressed as

$$(\boldsymbol{s}, \boldsymbol{\theta}) = \underset{\boldsymbol{s},\boldsymbol{\theta}}{\operatorname{argmin}}\, ||\tilde{\boldsymbol{s}} - \boldsymbol{s}||_2^2 + 2\sqrt{2}\sigma_w^2 \sum_j \frac{|s_j|}{\theta_j} + 2\sigma_w^2 \sum_j \log\theta_j. \tag{13}$$

Note that in LSM we have $\boldsymbol{s} = \boldsymbol{\Lambda\alpha}$, where $\boldsymbol{\Lambda} = \text{diag}(\theta_j) \in \mathbb{R}^{n\cdot m\cdot L\times n\cdot m\cdot L}$. Then Eq. (13) can be rewritten as

$$\begin{aligned}(\boldsymbol{\alpha}, \boldsymbol{\theta}) = &\underset{\boldsymbol{s},\boldsymbol{\theta}}{\operatorname{argmin}}\, ||\tilde{\boldsymbol{s}} - \boldsymbol{\Lambda\alpha}||_2^2 + 2\sqrt{2}\sigma_w^2 \sum_j |\alpha_j| \\ &+ 4\sigma_w^2 \sum_j \log(\theta_j + \epsilon),\end{aligned} \tag{14}$$

where ϵ is a small constant for numerical stability. From Eq. (14), we can see that with LSM prior the sparse estimation of $\boldsymbol{s}$ has been translated into the joint estimation of $\boldsymbol{\alpha}$ and $\boldsymbol{\theta}$.

3.3. Alternative Optimization

A straightforward approach to solve Eq.(14) is to adopt the alternative optimization, which consists of the iterations of solving two sub-problems. Specifically, both sub-problems admit closed-form solutions. For an initial estimate of $\boldsymbol{\alpha}$, we solve for $\boldsymbol{\theta}$ by optimizing

$$\boldsymbol{\theta} = \underset{\boldsymbol{\theta}}{\operatorname{argmin}}\, ||\tilde{\boldsymbol{s}} - \mathbf{A}\boldsymbol{\theta}||_2^2 + 4\sigma_w^2 \sum_j \log(\theta_j + \epsilon), \tag{15}$$

where $\mathbf{A} = \text{diag}(\boldsymbol{\alpha})$. Equivalently, Eq. (15) can also be rewritten as

$$\boldsymbol{\theta} = \underset{\boldsymbol{\theta}}{\operatorname{argmin}} \sum_j \{a_j\theta_j^2 + b_j\theta_j + c\log(\theta_j + \epsilon)\}, \tag{16}$$

where $a_j = \alpha_j^2$, $b_j = 2\alpha_j\tilde{s}_j$ and $c = 4\sigma_w^2$. Thus, Eq. (16) can be solved by solving a sequence of scalar minimization problem

$$\theta_j = \underset{\theta_j}{\operatorname{argmin}}\, a_j\theta_j^2 + b_j\theta_j + c\log(\theta_j + \epsilon), \tag{17}$$

which can be solved by taking $\frac{df(\theta_j)}{d\theta_j} = 0$, where $f(\theta)$ denotes the right hand side of Eq. (17). By taking $\frac{df(\theta_j)}{d\theta_j} = 0$, two stationary points can be obtained, i.e.,

$$\theta_{j,1} = -\frac{b_j}{4a_j} + \sqrt{\frac{b_j^2}{16} - \frac{c}{2a_j}}, \theta_{j,2} = -\frac{b_j}{4a_j} - \sqrt{\frac{b_j^2}{16} - \frac{c}{2a_j}} \tag{18}$$

when $b_j^2/(16a_j^2) - c/(2a_j) \geq 0$. Thus, the global minimizer of Eq. (17) can be obtained by comparing $f(0)$, $f(\theta_{j,1})$ and $f(\theta_{j,2})$.

When $b_j^2/(16a_j^2) - c/(2a_j) < 0$, there are no stationary points in the range of $[0, \infty)$. Since ϵ is a very small positive constant, $g(0) = b_j + c/\epsilon$ is always positive. Therefore, $f(0)$ is the global minimizer for this case. The solution to Eq. (17) can then be written as

$$\theta_j = \begin{cases} 0, & \text{if } b_j^2/(16a_j^2) - c/(2a_j) < 0, \\ t_j, & \text{otherwise} \end{cases} \tag{19}$$

where $t_j = \operatorname{argmin}_{\theta_j}\{f(0), f(\theta_{j,1}), f(\theta_{j,2})\}$.

For fixed $\boldsymbol{\theta}$, $\boldsymbol{\alpha}$ can be solved by solving

$$\boldsymbol{\alpha} = \underset{\boldsymbol{\alpha}}{\operatorname{argmin}}\, ||\tilde{\boldsymbol{s}} - \boldsymbol{\Lambda\alpha}||_2^2 + 2\sqrt{2}\sigma_w^2 \sum_j |\alpha_j|, \tag{20}$$

which admits a closed-form solution, as

$$\alpha_j = \mathcal{S}_{\tau_j}(\frac{\tilde{s}_j}{\theta_j}), \tag{21}$$

wherein $\mathcal{S}_{\tau_j}(\cdot)$ denotes the soft-thresholding function with threshold $\tau_j = \frac{\sqrt{2}\sigma_w^2}{\theta_j^2}$.

By alternatively solving the sub-problems of Eqs.(15) and (20), the sparse coefficients $\boldsymbol{s}$ can be estimated as $\hat{\boldsymbol{s}} = \hat{\boldsymbol{\Lambda}}\hat{\boldsymbol{\alpha}}$, wherein $\hat{\boldsymbol{\Lambda}}$ and $\hat{\boldsymbol{\alpha}}$ denotes the estimates of $\boldsymbol{\Lambda}$ and $\boldsymbol{\alpha}$, respectively. Then, the reconstructed tensor can be obtained by

$$\hat{\mathcal{X}} = \hat{\mathcal{S}} \times_1 \mathbf{U} \times_2 \mathbf{V} \times_3 \mathbf{W}, \tag{22}$$

where $\hat{\mathcal{S}}$ is the coefficient array correspond to $\hat{\boldsymbol{s}}$.

4. Multidimensional Image Denoising with Low-rank Tensor Approximation

In this section, we apply the proposed low-rank tensor approximation to multidimensional image denoising. Here, without loss of generality, we only consider the volumetric image. Let the noisy multidimensional image be denoted as

$\mathcal{Y} = \mathcal{X} + \mathcal{N}$, where $\mathcal{X} \in \mathbb{R}^{H\times W\times L}$ and $\mathcal{N} \in \mathbb{R}^{H\times W\times L}$ denote the noiseless multidimensional image and additive noise, respectively. Let $\mathcal{Y}_i = \tilde{\mathcal{R}}_i\mathcal{Y}$ denote the 3^{rd} tensor formed by the set of similar 3D patches to exemplar patch $\mathcal{P}_i$, where $\tilde{\mathcal{R}}_i$ denotes the operator grouping the set of patches similar to $\mathcal{P}_i$ into a 3^{rd} tensor. Then, the image denoising of the whole multidimensional image can be expressed by

$$\begin{aligned}(\mathcal{X}, \{\mathcal{S}_i\}) = \underset{\mathcal{X},\{\mathbf{D}_i\},\{\mathcal{S}_i\}}{\operatorname{argmin}} & ||\mathcal{Y} - \mathcal{X}||_F^2 \\ & + \eta \sum_i ||\tilde{\mathcal{R}}_i\mathcal{X} - \mathcal{S}_i \times_1 \mathbf{U}_i \times_2 \mathbf{V}_i \times_3 \mathbf{W}_i||_F^2 \\ & + 2\sqrt{2}\sigma_w^2 \sum_i ||\mathbf{\Lambda}_i \boldsymbol{s}_i||_1 + 2\sigma_w^2 \sum_i \log\boldsymbol{\theta}_i,\end{aligned} \tag{23}$$

where $\mathbf{U}_i, \mathbf{V}_i, \mathbf{W}_i$ denotes the set of orthogonal matrixes computed via HOSVD. Similar to the matrix SVD-based image denoising methods [7, 9, 10], the orthogonal matrixes $\mathbf{D}_i$ are also computed from the noisy input tensor $\mathcal{Y}_i$. Adopting the alternative optimization approach again, we solve for the whole multidimensional image denoising problem by solving the following two sub-problems.

4.1. Solving for whole image

Let $\hat{\mathcal{X}}_i = \mathcal{X}_i \times_1 \mathbf{U}_i \times_2 \mathbf{V}_i \times_3 \mathbf{W}_i$ denote the reconstructed low-rank tensor with initial estimate of $\mathcal{S}_i$. Then, for fixed $\{\mathcal{S}_i\}$, the whole image $\mathcal{X}$ can be estimated by solving the following ℓ_2-minimization problem

$$\mathcal{X} = \underset{\mathcal{X}}{\operatorname{argmin}} ||\mathcal{Y} - \mathcal{X}||_F^2 + \eta \sum_{i=1}^N ||\tilde{\mathcal{R}}_i\mathcal{X} - \hat{\mathcal{X}}_i||_F^2, \tag{24}$$

which is equivalent to the following equation by representing the tensors into long vectors

$$\boldsymbol{x} = \underset{\boldsymbol{x}}{\operatorname{argmin}} ||\boldsymbol{y} - \boldsymbol{x}||_2^2 + \eta \sum_{i=1}^N ||\tilde{\mathbf{R}}_i\boldsymbol{x} - \hat{\boldsymbol{x}}_i||_2^2, \tag{25}$$

where $\boldsymbol{y} \in \mathbb{R}^{H\cdot W\cdot L}, \boldsymbol{x} \in \mathbb{R}^{H\cdot W\cdot L}, \hat{\boldsymbol{x}} \in \mathbb{R}^{\sqrt{n}\cdot\sqrt{n}\cdot L}$ correspond to the vector representations of the tensors $\mathcal{Y}, \mathcal{X}, \hat{\mathcal{X}}_i$, respectively, and $\tilde{\mathbf{R}}_i \doteq [\tilde{\mathbf{R}}_{i_0}, \tilde{\mathbf{R}}_{i_1}, \cdots, \tilde{\mathbf{R}}_{i_{m-1}}]$ denotes the operator extracting the patches similar to $\boldsymbol{y}_i$. Eq. (25) can be solved in a closed-form, as

$$\boldsymbol{x} = (\mathbf{I} + \eta \sum_{i=1}^N \tilde{\mathbf{R}}_i^\top \tilde{\mathbf{R}}_i)^{-1}(\boldsymbol{y} + \eta \sum_{i=1}^N \tilde{\mathbf{R}}_i^\top \hat{\boldsymbol{x}}_i), \tag{26}$$

where the matrix to be inverted is diagonal and can be calculated easily. Similar to the K-SVD approach, Eq. (26) can be computed by averaging the reconstructed 3D patches sets $\hat{\mathcal{X}}_i$.

4.2. Solving for $\{\boldsymbol{s}_i\}$ and $\{\boldsymbol{\theta}_i\}$

For fixed $\mathcal{X}$, Eq. (23) reduces to a set of sequence of low-rank tensor approximation problems for each exemplar 3D patch, i.e.,

$$(\boldsymbol{s}_i, \boldsymbol{\theta}_i) = \underset{\boldsymbol{s}_i, \boldsymbol{\theta}_i}{\operatorname{argmin}} ||\tilde{\boldsymbol{s}}_i - \boldsymbol{s}_i||_2^2 + 2\sqrt{2}\frac{\sigma_w^2}{\eta}||\mathbf{\Lambda}_i \boldsymbol{s}_i||_1 + 2\frac{\sigma_w^2}{\eta}\log\boldsymbol{\theta}_i, \tag{27}$$

where we have used $\tilde{\mathcal{S}}_i = \mathcal{X}_i \times_1 \mathbf{U}_i^\top \times_2 \mathbf{V}_i^\top \times_3 \mathbf{W}_i^\top$. This is exactly the problem we have studied in previous section.

The overall multidimensional image (MDI) denoising algorithm based on nonlocal low-rank tensor approximation with Laplacian Scale Mixture (NLTA-LSM) is summarized in **Algorithm 1**. We found that the inner iteration converges in just a few iterations ($J = 2$ in our implementation). In **Algorithm 1**, we used the iterative regularization. The noise as well as the removed image details are fed back to the denoised image. The amount of noise is controlled by a small positive parameter δ.

Algorithm 1 NLTA-LSM based MDI denoising

- **Initialization**:
  - (a) Set the initial estimate $\hat{\mathcal{X}} = \mathcal{Y}$ and the parameter η;
  - (b) Obtain the set of tensors $\{\mathcal{X}_i\}$ from $\hat{\mathcal{X}}$ via k-NN search for each exemplar patch.
- **Outer loop**: for $k = 1, 2, \ldots, K_{max}$ **do**
  - (a) Tensor dataset $\mathcal{X}_i$ construction: grouping a set of similar 3D patches into a 3^{rd} tensor for each exemplar patch;
  - (b) **Inner loop** (Low-rank tensor approximation by solving Eq. (27)): for $j = 1, 2, \ldots, J$ **do**
    - (I) Compute $\boldsymbol{\theta}_i$ for fixed $\boldsymbol{\alpha}_i$ via Eq.(19);
    - (II) Compute $\boldsymbol{\alpha}_i$ for fixed $\boldsymbol{\theta}_i$ via Eq.(21);
    - (III) Output $\boldsymbol{s}_i = \operatorname{diag}(\boldsymbol{\theta}_i)\boldsymbol{\alpha}_i$ if $j = J$.
  - **End for**
  - (c) Reconstruct $\{\mathcal{X}_i\}$ from $\{\mathcal{S}_i\}$ via Eq.(22).
  - (d) Reconstruct the whole image $\hat{\mathcal{X}}^{(k+1)}$ from $\{\mathcal{X}_i\}$ by solving Eq.(26).
  - (e) If $k < K_{max}$ set $\hat{\mathcal{X}}^{(k+1)} = \hat{\mathcal{X}}^{(k+1)} + \delta(\mathcal{Y} - \hat{\mathcal{X}}^{(k+1)})$
- **End for**
- Output $\hat{\mathcal{X}}^{(k+1)}$

---

5. Experimental results

We have implemented the proposed algorithm under MATLAB. Both the multispectral images and the MR image sequences are used to verify the denoising performance of the proposed algorithm with comparison to existing state-of-the-art denoising methods. There are only a few parameters needed to be set in the proposed algorithm: block size $5 \times 5 \times L$ (L denotes the number of spectral bands or the number of MRI frames), the number of similar patches

$m = 60$, the regularization parameter $\delta = 0.12$, and iteration numbers $K_{max} = 7$ and $J = 2$.

5.1. Multispectral image denoising

The CAVE database [23] consisting of 32 hyperspectral images of common objects are used to verify the performance of the proposed method. The images of size $512 \times 512 \times 31$ are captured with the wavelengths in the range of $400 - 700$ nm at a interval of 10 nm. We select 10 hyperspectral images as the test set. Both the additive Gaussian noise and the mixed noise of Gaussian and Poisson noise used in [18] are used to simulate the noisy spectral images. Two sets of experiments are conducted. In the first experiment, the additive Gaussian noise with different standard derivations is added to the hyperspectral images to simulate the noisy images. In the second experiment, the mixed noise of additive Gaussian and Poisson noise is added. In this setting, the standard derivations of the Gaussian noise are varied from 10 to 100, and the Poisson noise is fixed with variance $\boldsymbol{y}/2^k$, wherein $k = 5$. We compared the proposed method with several recently developed multispectral image denoising methods, including the tensor dictionary learning (TensorDL) method [18], the BM4D method [13], the PARAFAC method [12], the low-rank tensor approximation (LATA) method [21], the ANLM3D method [17] and the band-wise BM3D method [5] [2]. For the mixture noise, we applied the variance-stabilizing transformation (VST) [16] to the noisy spectral images before applying a test method, followed by the inverse VST after denoising, as done in [18].

The average PSNR results for each noise level are reported in Table 1. From Table 1, it can be seen that the proposed method consistently outperforms other competing methods. The average PSNR improvements over the TensorDL and BM4D methods, which are respectively among the 2^{nd} and 3^{rd} best methods in the comparison study group, are larger than 2 dB. In Fig. 1 we show the parts of the denoised images at 410nm band of *Toy* and *Painting* with Gaussian noise of $\sigma_w = 10$. It can be seen that the other five test methods tend to generate visual artifacts. Clearly, the proposed method reconstructed the images with much less artifacts than the other methods.

5.2. 3D MRIs denoising

We also applied the proposed method for 3D MRIs denoising. The $T1$-weighted 3D MRIs are obtained from the Brainweb database[3]. The 3D MRIs is of size $181 \times 181 \times 10$ with $1 \times 1 \times 1mm^3$ resolution. Additive Gaussian noise with different noise levels σ_w is added to simulate the noisy 3D MRIs [4] The proposed method is compared with some recently developed 3D MRIs denoising methods, including the ANLM3D method [17], the band-wise BM3D method [5], and the BM4D method [13]. The LATA [21] and TensorDL [18] methods for spectral images denoising are also included for comparison study.

Table 2 show the PSNR results for each noise level. From Table 2 we can see that the BM4D method [13] performs much better than the TensorDL method [18]. The reason is that the correlations between the slices are not strong and smaller 3D patches (i.e., $4 \times 4 \times 4$) used in BM4D can better exploit the local correlations. Even though the full slices 3D patches (i.e., $5 \times 5 \times 10$) are used, the proposed method still outperforms the BM4D [13] for all noise levels. The PSNR gain over BM4D method can be up to 1.13 dB. Parts of the reconstructed MRI by the test methods are shown in Fig. 2. We can see that the MRI reconstructed by the propose method contains less visual artifacts than other methods.

[2]We thank the authors of [18, 13, 12, 21, 17, 5] for providing their source codes in their websites.

[3]http://brainweb.bic.mni.mcgill.ca/brainweb/

[4]Our method can also be used for Rician noise by using the VST [16].

6. Conclusions

In this paper we proposed a low-rank tensor approximation approach for multidimensional image denoising. To fully exploit the correlations across all the dimensions, 3D image patches are extracted and grouped into 3^{rd} tensors, which can be effectively approximated with low-rank tensors by HOSVD followed by the thresholding of the resulting coefficient arrays. For adaptive low-rank tensor approximation, we propose a new sparse regularization term for the sparse coefficient array using the Laplacian scale mixture model (LSM). With LSM modeling, the low-rank tensor approximation problem is translated into the alternative optimization of the sparse coefficient array and the scalar variables. We show that both subproblems can be solved in closed-form. Experimental results on both the hyperspectral images and the MRI volumetric data show that proposed method performs significantly better than existing methods.

References

[1] A. Beck and M. Teboulle. Fast gradient-based algorithms for constrained total variation image denoising and deblurring problems. *IEEE Trans. On Image Process.*, 18(11):2419–2434, Nov. 2009. 1

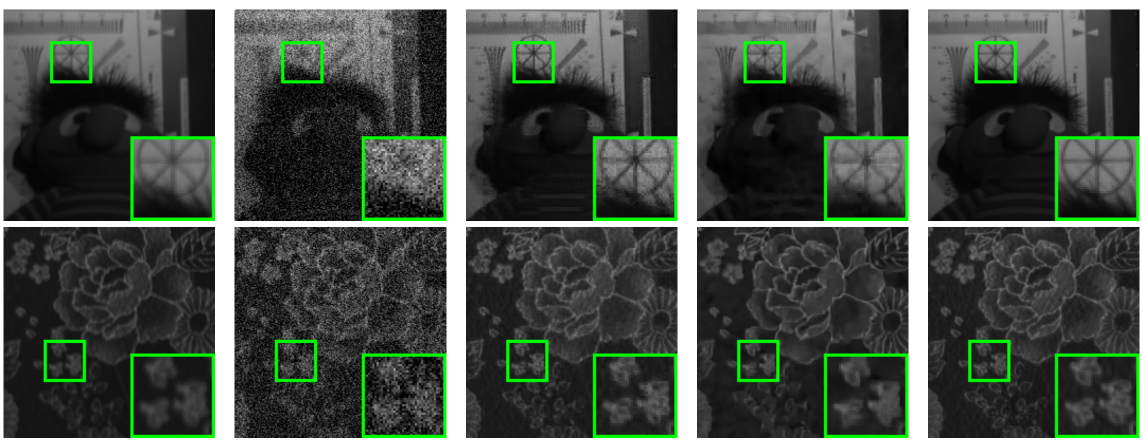
[2] G. Bergqvist and E. G. Larsson. The higher-order singular value decomposition: Theory and an application. *IEEE Signal Processing Magazine*, pages 151–154, May 2010. 3

[3] A. Buades, B. Coll, and J. M. Morel. A non-local algorithm for image denoising. In *IEEE Int. Conf. on CVPR*, volume 2, pages 60–65, 2005. 1, 2

[4] E. Candes, M. Wakin, and S. Boyd. Enhancing sparsity by reweighted l_1 minimization. *Journal of Fourier Analysis and Applications*, 14(5):877–905, 2008. 3

Table 1. Average PSNR results of the competing methods for different noise levels on the set of test hyperspectral images.

method	Gaussian noise				
	$\sigma_w = 10$	$\sigma_w = 20$	$\sigma_w = 30$	$\sigma_w = 50$	$\sigma_w = 100$
PARAFAC [12]	30.88	30.71	30.54	29.21	25.90
ANLM3D [17]	38.86	35.22	34.45	31.91	29.22
LRTA [21]	39.43	36.16	34.16	31.56	28.07
BwBM3D [5]	40.20	36.50	34.36	31.84	28.43
BM4D [13]	43.23	39.48	37.21	34.33	30.45
TensorDL [18]	42.92	39.25	37.19	34.56	31.17
Proposed NLTA-LSM	**45.32**	**41.84**	**39.74**	**37.05**	**33.190**
method	Mixture of Poisson and Gaussian noise with fixed $k = 4$				
	$\sigma_w = 10$	$\sigma_w = 20$	$\sigma_w = 30$	$\sigma_w = 50$	$\sigma_w = 100$
PARAFAC [12]	30.95	28.47	27.34	23.87	20.09
ANLM3D [17]	33.36	32.66	32.23	31.17	29.18
LRTA [21]	33.22	33.01	31.97	30.68	27.99
BwBM3D [5]	33.94	32.94	31.85	30.34	27.38
BM4D [13]	36.95	36.00	34.84	33.26	30.31
TensorDL [18]	36.69	35.64	34.89	33.26	30.76
Proposed NLTA-LSM	**39.50**	**38.37**	**37.14**	**35.87**	**32.95**



(a) Original (b) Noisy (c) TensorDL [18] (d) BM4D [13] (e) **NLTA-LSM**

Figure 1. (a)Original images at 410nm band of *Toy* and *Cloth* in CAVE dataset [23]; (b) The images corrupted by Gussian noise of $\sigma_w = 30$; (c) TensorDL [18] (PSNR=37.02dB,32.10dB); (d) BM4D [13] (PSNR=37.49dB,32.24dB); (e)**Proposed NLTA-LSM** (PSNR=**40.54**dB,**34.27**dB).

[5] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian. Image denoising by sparse 3-d transform-domain collaborative filtering. *IEEE Trans. Image Process.*, 16(8):2080–2095, Aug. 2007. 1, 2, 6, 7, 8

[6] W. Dong, X. Li, L. Zhang, and G. Shi. Sparsity-based image denoising via dictionary learning and structural clustering. In *Proc. of the IEEE CVPR*, pages 457–464, 2011. 1, 2

[7] W. Dong, G. Shi, and X. Li. Nonlocal image restoration with bilateral variance estimation: a low-rank approach. *IEEE Trans. Image Process.*, 22(2):700–711, Feb. 2013. 1, 2, 5

[8] M. Elad and M. Aharon. Image denoising via sparse and redundant representations over learned dictionaries. *IEEE Trans. Image Process.*, 15(12):3736–3745, Dec. 2006. 1, 2

[9] M. Gavish and D. L. Donoho. The optimal hard threshold for singular values is $4/\sqrt{3}$. *IEEE Trans. Information Theory.*, 60(8):5040–5053, Aug. 2014. 5

[10] S. Gu, L. Zhang, W. Zuo, and X. Feng. Weighted nuclear norm minimization with application to image denoising. In *Proc. of the IEEE CVPR*, 2014. 1, 2, 5

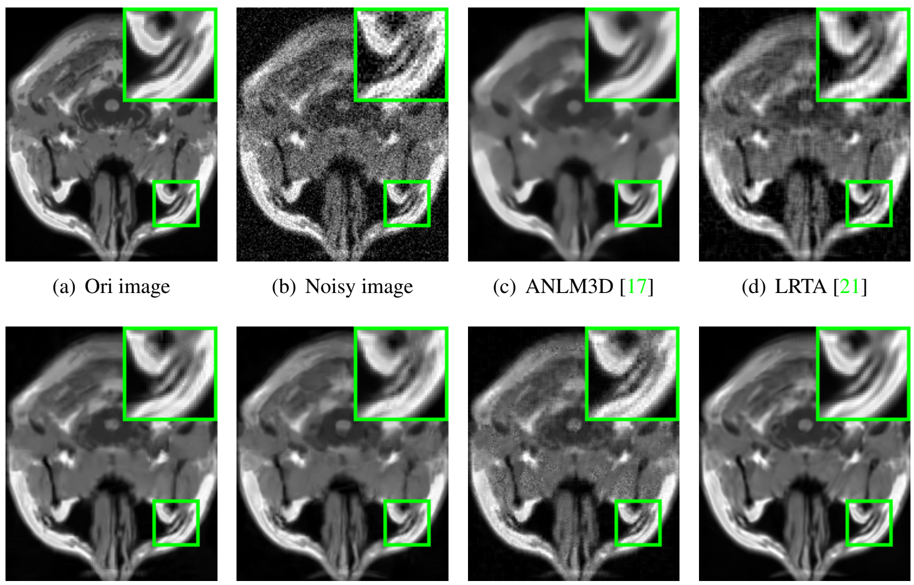
[11] L. D. Lathauwer, B. D. Moor, and J. Vandewalle. A multilinear singular value decomposition. *SIAM J. Matrix Anal. Applicat.*, 21(4):1253–1278, 2000. 3

[12] X. Liu, S. Bourennane, and C. Fossati. Denoising of hyperspectral images using the parafac model and statistical performance analysis. *IEEE Trans. Geoscience and Remote Sensing*, 50(10):3717C–3724, Oct. 2012. 2, 6, 7

[13] M. Maggioni, V. Katkovnik, K. Egiazarian, and A. Foi. Nonlocal transform-domain filter for volumetric data denoising

Table 2. The PSNR results of the test methods for additive Gaussian noise on the 3D MRIs.

method	Dynamic MRI sequence				
	$\sigma_w = 10$	$\sigma_w = 20$	$\sigma_w = 30$	$\sigma_w = 50$	$\sigma_w = 100$
ANLM3D [17]	32.85	29.11	27.30	25.49	23.59
LRTA [21]	32.79	28.72	27.08	24.50	21.59
BwBM3D [5]	35.42	31.81	29.78	27.33	23.94
BM4D [13]	36.53	33.03	31.06	28.58	25.28
TensorDL [18]	33.63	30.03	28.48	26.18	23.66
Proposed NLTA-LSM	**37.06**	**33.62**	**31.83**	**29.53**	**26.41**



(a) Ori image (b) Noisy image (c) ANLM3D [17] (d) LRTA [21]

(e) BwBM3D [5] (f) BM4D [13] (g) TensorDL [18] (h) **NLTA-LSM**

Figure 2. (a) The original MRI (the 3^{rd} slice); (b) The noisy MRI ($\sigma_w = 30$, PSNR=18.58dB); denoised MRI by (c) ANLM3D [17] (PSNR=27.30dB); (d) LRTA [21] (PSNR=27.08dB); (e) BwBM3D [5] (PSNR=29.78dB); (f) BM4D [13] (PSNR=31.06dB); (g) TensorDL [18] (PSNR=28.48dB); (h) **Proposed NLTA-LSM** (PSNR= **31.83** dB).

and reconstruction. *IEEE Trans. Image Process.*, 22(1):119–133, Jan. 2013. 2, 6, 7, 8

[14] J. Mairal, F. Bach, J. Ponce, G. Sapiro, and A. Zisserman. Non-local sparse models for image restoration. In *Proc. of the IEEE ICCV*, Tokyo, Japan, 2009. 1, 2

[15] J. Mairal, M. Elad, and G. Sapiro. Sparse representation for color image restoration. *IEEE Trans. on Image Processing*, 17(1):53–69, Jan. 2008. 1, 2

[16] M. Makitalo and A. Foi. Optimal inversion of the generalized anscombe transformation for poisson-gaussian noise. *IEEE Trans. Image Process.*, 22(1):91–103, 2013. 6

[17] J. V. Manjon, P. Coupe, L. Marti-Bonmati, L. Collins, and M. Robles. Adaptive non-local means denoising of mr images with spatially varying noise levels. *Journal of Magnetic Resonance Imaging*, 31(1):192–203, 2010. 6, 7, 8

[18] Y. Peng, D. Meng, Z. Xu, C. Gao, Y. Yang, and B. Zhang. Decomposable nonlocal tensor dictionary learning for multispectral image denoising. In *Proc. of the IEEE CVPR*, pages 4321–4328, 2014. 2, 6, 7, 8

[19] J. Portilla, V. Strela, M. Wainwright, and E. Simoncelli. Image denoising using scale mixtures of gaussians in the wavelet domain. *IEEE Trans. Signal Process.*, 12(11):1338C–1351, Nov. 2003. 1, 4

[20] A. Rajwade, A. Rangarajan, and A. Banerjee. Image denoising using the higher order singular value decomposition. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 35(4):849C–862, 2013. 2, 3

[21] N. Renard, S. Bourennane, and J. Blanc-Talon. Denoising and dimensionality reduction using multilinear tools for hyperspectral images. *IEEE Geoscience and Remote Sensing Letters*, 5(2):138–142, 2008. 2, 6, 7, 8

[22] M. Wax and T. Kailath. Detection of signals by information theoretic criteria. *IEEE Trans. Acoustics, Speech and Signal Process.*, 33(2):387–392, 1985. 2, 3

[23] F. Yasuma, T. Mitsunaga, D. Iso, and S. K. Nayar. Generalized assorted pixel camera: postcapture control of resolution, dynamic range, and spectrum. *IEEE Trans. Image Process.*, 19(9):2241–2253, 2010. 6, 7

[24] L. Zhang, W. Dong, D. Zhang, and G. Shi. Two-stage image denoising by principal component analysis with local pixel grouping. *Pattern Recognition*, 43:1531–1549, Apr. 2010. 1, 2