Hyperspectral Image Denoising Using 3-D Geometrical Kernel With Local Similarity Prior

Feng Zhang, Kai Zhang ⊕, Jiande Sun ⊕, Zhixi Feng ⊕, Wenbo Wan ⊕, and Quanyuan Wu

Abstract—The majority of existing hyperspectral (HS) image denoising methods exploit local similarity in HS images by rearranging them into the matrix or vector forms. As the typical 3-D data, the inherent spatial and spectral properties in HS images should be simultaneously explored for denoising. Therefore, a 3-D geometrical kernel (3DGK) is developed in this article to describe the local structure. The proposed method assumes that the pixel can be represented by other pixels within a 3-D block efficiently owing to the local similarity with adjacent positions. Then, the HS image is modeled by the 3-D kernel regression with $L_1$-norm constraint, in which the local similarity is captured by the proposed 3DGK. To efficiently compute the parameters in 3DGK, geometrical structures, such as scale, shape, and orientation in the 3-D block, are estimated from the gradient information approximately. Finally, the noises are effectively removed while preserving the structures in HS images. Moreover, experimental results on simulated and real datasets demonstrate that the performance of 3DGK is better than those of the methods based on local similarity prior.

Index Terms—3-D geometrical kernel (3DGK), geometrical structure, hyperspectral (HS) image denoising, kernel regression, local similarity (LS).

I. INTRODUCTION

With the rapid development of imaging sensors, hyperspectral (HS) remote sensing technology has been attracting much attention and HS images have also been applied in many scene interpretation tasks, such as land-use classification [1], target detection [2], and unmixing [3]–[6], due to containing abundant spectral information. However, during acquiring or transmitting HS images are always corrupted by some undesired noises [7], which have some impacts on the performance of the above-mentioned tasks. Therefore, it is vital to efficiently remove the noises and recover the HS image accurately.

Over the past two decades, many methods are proposed to achieve the denoising of HS images by adopting different tools. For example, Li et al. [8] proposed an HS image denoising method based on distributed sparse representation [9], which exploited the intraband and interband information for dictionary learning. Qian and Ye [10] introduced the spatial and spectral structure into sparse representation for HS denoising, in which similar patches are collected to ensure the sparsity. Fu et al. [11] learned an adaptive spatial-spectral dictionary from the noisy HS images for noise removal. Subsequently, Zhuang and Bioucas-Dias [12] combined low-rank and sparse representation to capture the spatial and spectral correlation in HS images. In [13], Xiong et al. embedded the low-rank prior into tensor to model the correlations in both spectral and spatial domains, which is regularized by $L_0$ gradient term. Besides, to capture the nonlocal self-similarity [14], Zhang et al. [15] proposed a total variation (TV) regularized nonlocal low-rank tensor decomposition model to restore the clean HS images. For accurate weight estimation, Hu et al. [16] proposed an adaptive nonlocal means denoising method for the denoising of HS images. Xue et al. [17] further explored the global correlation across the spectrum and nonlocal similarity in HS images to regularize the low-rank tensor model. Besides, Xie et al. [18] proposed an intrinsic tensor sparsity measure to encode the low-rank prior in the Tucker decomposition model, which also considered the nonlocal similarity. Chen et al. [19] utilized the tensor-ring decomposition model to construct high-order tensors, which improves the ability for noisy HS images. Moreover, other priors, such as nonnegativity [20], are also incorporated with the tensor model for HS image denoising.

Besides, a large number of methods are also specifically developed for HS image denoising by integrating local information priors, which has been widely focused and extensively applied to HS image denoising. Local prior mainly captures the inherent information of HS image locally, because it is difficult to describe the global prior in the whole HS images directly. As a popular model, TV [21] is proposed to explore the local gradient information in images, which is extended for HS images by considering the spectral property. For example, a spectral–spatial adaptive hyperspectral TV (SSAHTV) is proposed in [22] to further exploit the noise intensity differences and spatial property differences. Further, the anisotropic spatial and spectral TV is designed to deal with the piecewise structures in [23] and low-rank tensor decomposition is considered to capture the global spatial and spectral correlation of HS images. Meanwhile, He et al. [24] also employed

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/
TV to regularize the local low-rank decomposition model for denoising of HS images. Fan et al. [25] presented a spatial and spectral TV model to deal with spectral structure loss led by the bandwise TV. In [26], 1-D TV along the spectral dimension is introduced to preserve the spectral signatures. Besides, local similarity (LS) [27] is also an efficient prior commonly used for HS image denoising. It is assumed that the pixel values in a local region are similar and they can represent each other due to the close location among them. For example, Zhao and Yang [28] utilized sparse and low-rank constraints to capture the local redundancy and correlation in HS images, which can well preserve the spatial and spectral details in the denoised HS images. Lu et al. [29] divided all bands in HS images into several groups according to spectral correlation and then spatial adaptive LS is considered to find related pixels. In [30], the local similar pixels are defined by superpixel segmentation, which avoids dividing HS images into square patches. Recently, some HS image denoising methods based on deep neural networks [31]–[34] are also proposed and acquire satisfactory results. These methods hierarchically extracted image features from local regions, which are also inspired by local prior. For example, the local structures in HS images are extracted in convolutional neural networks level by level [35].

As a powerful tool to model local relationships, kernel-based methods are usually employed to describe the LS prior among adjacent pixels and have been generally used for natural image denoising. For example, Takeda et al. [36] analyzed the local structures in images by kernel regression and obtained better denoising results. In [37], Bouboulis et al. adopted the semiparametric formulation to establish the adaptive kernel for image denoising. Subsequently, the local geometrical structures are further exploited in [38] and the sparsity of neighbors is cast on the steerable kernel. Then, the LS prior captured by the kernel is also applied to the denoising of HS images. For example, Zhong and Wang [39] considered the conditional random field to handle the edges and textures in a local region, in which a filter bank is used to represent the structures. Moreover, kernel regression is also utilized in [40] and a structure tensor is defined to explore the local gradient orientation information. However, the spectral information is represented by a fixed vector in [40], which cannot adaptively estimate the features in the spectral direction. These methods obtain better denoised HS images, the inner structures in 3-D blocks of HS images are not analyzed carefully and deeply. The LS between different pixels is not only related to the distance in position but also decided by the geometrical structures, such as edges and textures. As shown in Fig. 1(a), the pixel \( p \) locating in the sharp edge is not similar to pixel \( p_1 \) although the distance in position between them is close, where the pixels are labeled by a circle. Inversely, \( p \) looks very similar to the pixel \( p_2 \) in intensity because they are in the same edge structure. Moreover, we compute the correlation coefficients (CC) of the adjacent bands of the HS image in Fig. 1(a) and the CC curve is displayed in Fig. 1(b). From Fig. 1(b), one can see that the similarity in adjacent bands is high owing to the high spectral resolution. Thus, the similarity in the spectral domain should also be considered.

Therefore, to simultaneously exploit the local geometrical structure and spectral correlation in HS images, we proposed a new 3-D geometrical kernel (3DGK) to remove the noises in HS images from the perspective of kernel regression in this article. It is assumed that the other pixels in a 3-D block centered by the pixel to be denoised are similar to the center pixel. Then, the similarity weights between the center pixel and the other neighboring pixels are measured by the 3DGK. In the proposed method, the inner structures within a local 3-D block of HS images can be approximated adaptively by the 3DGK, where the scale, shape, and orientation of the kernel are deformable and can be adjusted by elongation, rotation, and scaling. The estimated kernel not only considers the distance information but also makes full use of the spatial and spectral structures. In 3DGK, the similarity weights can be inferred more reasonably. Through the geometrical structure prior, the noises in HS images can be reduced efficiently while preserving the spectral and spatial details.

The contributions of the proposed method can be summarized as follows.

1) We present a novel 3DGK regression model for HS images denoising, which adopts the geometrical structures in 3-D blocks to capture the LS in spatial and spectral domains. The geometrical structures are reflected by the weights assigned to each pixel within the 3-D blocks.

2) We analyze the geometrical structures in the 3-D blocks deeply and infer the scaling, elongation, and rotation factors from the local gradient information in the 3-D blocks to estimate more accurate weights.

The rest of this article is organized as follows. In Section II, the 3DGK is constructed and its denoising process is described. In Section III, experimental results on different datasets and noise distributions are given. Finally, Section IV concludes this article.

II. 3DGK DENOISING

In this section, according to the LS assumption, the pixel to be denoised in a 3-D block is estimated by the neighboring pixels under the 3-D kernel regression framework. Then, the inner geometrical structures are adaptively analyzed by elongation, rotation, and scaling. Finally, the denoised HS images are produced by the proposed 3DGK.
A. Kernel Regression in HS Images

Similar to the model in [36], the denoising model can be written as (1) for the noisy HS image according to regression theory [41]

\[ p_i = f(x_i) + \varepsilon_i, \quad i = 1, 2, \ldots, N \]

where \( p_i \) is the noisy pixel value. \( x_i = [x_i^h, x_i^v, x_i^t]^T \) denotes the position of the pixel. \( f(\cdot) \) is a regression function. Here, the two spatial dimensions correspond to horizontal and longitudinal axes, respectively. The spectral dimension is viewed as the vertical axis. So, \( x_i^h, x_i^v, \) and \( x_i^t \) are coordinate values of \( p_i, \varepsilon_i \) is the noise. \( N \) equals to the number of estimated pixels and \( N = b^3 \) for a 3-D block with size \( b \times b \times b \). In a local 3-D block of HS image, the center pixel to be denoised is similar to the neighboring pixels. Then, the center pixel can be well represented. Thus, in the local 3-D block, the Taylor expansion of (1) is calculated as

\[ f(x_i) = f(x) + \nabla f(x)^T (x_i - x) + \frac{1}{2} (x_i - x)^T \mathcal{H} f(x) (x_i - x) \]

where \( \nabla \) and \( \mathcal{H} \) stand for Gradient and Hessian operators, respectively. Due to the symmetry property of the Hessian matrix, (2) can be rewritten as

\[ f(x_i) = \beta_0 + \beta_1^T (x_i - x) + \beta_2^T \text{ltr} \{ (x_i - x) (x_i - x)^T \} \]

where \( \text{ltr} \) lexicographically rearranges all elements in the lower triangular part of a matrix into a column vector. \( \beta_0 \) is equivalent to \( f(x) \), which is the pixel value to be estimated. \( \beta_1 \) and \( \beta_2 \) are computed by (4) and (5), respectively

\[ \beta_1 = \begin{bmatrix} \frac{\partial f(x)}{\partial x^h}, \frac{\partial f(x)}{\partial x^v}, \frac{\partial f(x)}{\partial x^t} \end{bmatrix}^T \]

\[ \beta_2 = \frac{1}{2} \begin{bmatrix} \frac{\partial^2 f(x)}{\partial (x^h)^2}, 2 \frac{\partial^2 f(x)}{\partial x^h \partial x^v}, 2 \frac{\partial^2 f(x)}{\partial x^h \partial x^t}, \ldots, \frac{\partial^2 f(x)}{\partial (x^v)^2}, 2 \frac{\partial^2 f(x)}{\partial x^v \partial x^t}, \frac{\partial^2 f(x)}{\partial (x^t)^2} \end{bmatrix}^T . \]

Then, \( \beta_0, \beta_1, \) and \( \beta_2 \) can be obtained by minimizing (6). Besides, to produce clearer spatial details, we adopt \( L_1 \)-norm in (6) to replace the \( L_2 \)-norm used in [36]; then, the optimization of (6) becomes the minimization of the weighted \( L_1 \)-norm

\[ \hat{b} = \arg \min_b \| p - S_x b \|_1, w_x \]

\[ = \arg \min_b \| (p - S_x b) \circ w_x \|_1 \]

where \( \circ \) denotes elementwise multiplication. The related symbols in (6) are defined as follows:

\[ p = [p_1, p_2, \ldots, p_N]^T \]

\[ b = [\beta_0, \beta_1^T, \beta_2^T]^T \]

\[ w_x = [K_H (x_1 - x), K_H (x_2 - x), \ldots, K_H (x_N - x)]^T \]

\[ S_x = \begin{bmatrix} 1(x_1 - x)^T ltr^T \{(x_1 - x) (x_1 - x)^T\} \\ 1(x_2 - x)^T ltr^T \{(x_2 - x) (x_2 - x)^T\} \\ \ldots \\ 1(x_N - x)^T ltr^T \{(x_N - x) (x_N - x)^T\} \end{bmatrix} . \]

In (9), \( K_H(x) = \frac{1}{\det(H)} K(H^{-1}x) \) is a kernel determined by a smoothing factor \( H \) whose size is \( 3 \times 3 \). To estimate the denoised pixel values, an auxiliary variable is introduced for efficient optimization of (6), which is formulated as

\[ \min_{b, q} \| q \|_1 \text{s.t.} q = (p - S_x b) \circ w_x. \]

Then, the augmented Lagrangian multiplier method (ALM) [42] is considered to solve (11). So, the augmented Lagrange function of (11) is written as

\[ \min_{b, q} \| q \|_1 + \frac{\rho}{2} \| q - (p - S_x b) \circ w_x \|_F^2 \]

\[ + \langle y, q - (p - S_x b) \circ w_x \rangle. \]

where \( \rho \) is the penalty parameter and \( y \) is the Lagrangian multiplier. According to the iterative scheme, \( q \) is obtained by solving the subproblem

\[ F_q = \| q \|_1 + \frac{\rho}{2} \| q - (p - S_x b) \circ w_x - \frac{y}{\rho} \|_F^2 . \]

Equation (13) can be optimized efficiently by the soft-threshold operator in [43].

For \( b \), its subproblem is

\[ F_b = \frac{\rho}{2} \| q - (p - S_x b) \circ w_x \|_F^2 \]

\[ + \langle y, q - (p - S_x b) \circ w_x \rangle. \]

By replacing the elementwise multiplication with the matrix multiplication and simplification, (14) can be reformulated as

\[ F_b = \frac{\rho}{2} \| q - (p - S_x b) W_x + \frac{y}{\rho} \|_F^2 \]

where \( W_x = \text{diag}(w_x) \). \text{diag}(\cdot) transforms a vector into a diagonal matrix. This is a least-squares optimization problem, whose closed-form solution can be obtained by setting the derivate of (15) w.r.t. \( b \) as 0. The derivate of (15) w.r.t. \( b \) is

\[ \frac{\partial F_b}{\partial b} = \rho S_x^T W_x^T W_x S_x b + \rho S_x^T W_x q \]

\[ - \rho S_x^T W_x^T W_x p + S_x^T W_x y. \]

Finally, the multiplier \( y \) is updated by

\[ y = y + \rho (q - (p - S_x b) \circ w_x). \]

Taking the computation complexity into consideration, the iteration number is set as 20 due to the efficiency and rapid convergence of ALM. It is obvious that the value of the estimated pixel is \( \beta_0 \), which is the first element in \( b \). Naturally, if the weights in \( W_x \) are given, the denoised pixel value can be computed easily.
B. 3-D Geometrical Kernel

It can be found that the denoising performance of (6) depends upon the weight matrix \( W \), which is derived from the kernel \( K_H(x) \). So, the choice of the kernel is critical. A typical kernel is a Gaussian function, which uses the information of location distance between the center pixel and the neighboring pixels. For example, the Gaussian kernel is employed in [44] to find the smooth or texture components of HS images. It is obvious that the similarity may be low for the close pixels in location distance if they have different spatial or spectral appearances. In addition to the distance information, the pixels are very similar in a 3-D block of HS images, which not only have similar intensity in the spatial domain but also are similar in spectral response due to the high spectral resolution of HS images. Besides, there exist abundant geometrical structures, such as edges or textures in the 3-D block, which also should be considered for HS image denoising. As shown in Fig. 1(a), the pixels along the edges have a more similar intensity and the pixel values are closer in the smooth regions. For spectral domain, the structure similarity also can be found.

Although the geometrical structure prior is utilized in [36] for image denoising, only spatial information is calculated to estimate the weights that cannot take full advantage of the spectral information in HS images. For \( K_H(x) \), its property is packaged inside \( H \). Therefore, \( H \) should carry the location distance, intensity, and structure information in a 3-D block of HS images. Considering the requirement, 3DGK is constructed to simultaneously capture the geometrical information of HS images in the spatial and spectral domains, which is formulated as

\[
K_{H_i}(x_i - x) = \sqrt{\text{det}(C_i)} \exp\left\{ -\frac{(x_i - x)^T C_i (x_i - x)}{2h^2 \mu_i^2} \right\}
\]

where \( H_i = h \mu_i / \sqrt{C_i} \). \( h \) is a smoothing parameter. \( \mu_i \) denotes the local density. \( C_i \) contains the geometrical information, such as the orientation and magnitude of structures in a 3-D block, which can be reflected roughly by the covariance matrix of gradients in spatial and spectral domains. Thus, \( C_i \) is defined by (19) shown at the bottom of this page, where \( g_h(x_i) \), \( g_l(x_i) \), and \( g_v(x_i) \) are the gradients of the pixel in \( x_i \) along horizontal, longitudinal, and vertical directions, respectively. \( B_i \) is the 3-D block whose central point is the pixel \( p_i \). However, it is time-consuming to compute \( C_i \) directly. Fortunately, \( C_i \) can be decomposed approximately by (20) owing to the symmetry in \( C_i \):

\[
C_i \approx \gamma_i U_i \Lambda_i U_i^T
\]

(20)

where \( \gamma_i \) is the scaling factor of the 3-D kernel. According to the elongation matrix \( \Lambda_i \), the principal axes of the 3-D kernel are elongated. \( t_i, v_i, \) and \( w_i \) are the elongation factors on the principal axes of the 3-D kernel, respectively. The rotation of the 3-D kernel is denoted by \( U_i \), which can be modeled as (22) shown at the bottom of the next page, where \( R_{x_1}(\theta_i), R_{y_1}(\phi_i), \) and \( R_{z_1}(\psi_i) \) are the rotation matrices along with different directions. Thus, the rotation of principal axes is achieved by \( U_i \). \( \theta_i, \phi_i, \) and \( \psi_i \) are the rotation angles on horizontal, longitudinal, and vertical axes, which can be efficiently from \( U_i \) by dcm2angle in MATLAB.

For an intuitive understanding, the scaling, elongation, and rotation of a 3-D kernel are shown in Fig. 2, which gradually matches the spatial and spectral structures in the 3-D block. The 3-D ball kernel is first resized by \( \gamma_i \). Then, the lengths of the three axes are adjusted by \( \Lambda_i \). Finally, a triaxial ellipsoid kernel is rotated along with different directions through \( U_i \) and the 3DGK is obtained. Through the transformation of the above operations, the structures in the 3-D block can be efficiently represented.

According to the image local analysis in [45], the orientation of the structure in the 3-D block can be derived from its gradients. \( G_i = [g_h, g_l, g_v] \) contains the gradient information in the 3-D block:

\[
C_i = \begin{bmatrix}
\sum_{x_j \in B_i} g_h(x_j) g_h(x_j) \\
\sum_{x_j \in B_i} g_l(x_j) g_l(x_j) \\
\sum_{x_j \in B_i} g_v(x_j) g_v(x_j)
\end{bmatrix}
\]

(19)
block. \(g_a, g_b, \) and \(g_c\) are column vectors, which are comprised of the gradients of the 3-D block in different directions. The relationship between \(G_i\) and \(C_i\) can be written as
\[
C_i = G_i^T G_i. \tag{23}
\]

Then, \(U_i\) and \(\Lambda_i\) can be calculated from the singular value decomposition (SVD) of \(G_i\), which is
\[
G_i = V_i \Sigma_i U_i^T \tag{24}
\]
where \(V_i, \Sigma_i, \) and \(U_i\) are the corresponding decomposed results of \(G_i\) after SVD. \(U_i\) in (20) is the same as that \(U_i\) in (24) and \(\Sigma_i = \text{diag}\{[s_{i,1}, s_{i,2}, s_{i,3}]\}\). So, the parameters in (20) are efficiently inferred from (24) due to the numerical stability and efficiency of SVD. For example, \(\Lambda_i\) can be regarded as the amplitude of the principal axes of the 3-D kernel, whose elements are calculated by
\[
t_i = \frac{s_{i,1} + \alpha}{s_{i,2}s_{i,3} + \alpha}, \quad w_i = \frac{s_{i,2} + \alpha}{s_{i,1}s_{i,3} + \alpha}, \quad v_i = \frac{s_{i,3} + \alpha}{s_{i,1}s_{i,2} + \alpha} \tag{25}
\]
where \(\alpha\) is a regularization parameter to avoid meaningless values. By (25), the elongation factors can be normalized to avoid the large variation caused by pixel values. The shape of the 3-D kernel is controlled by \(\Lambda_i\). In homogeneous or smooth regions, the 3-D kernel is close to a ball in shape. In some heterogeneous or edges regions, the 3-D kernel is adjusted to capture the structure information. Therefore, the geometrical structure is embedded into the 3-D kernel naturally.

Besides, the scaling factor \(\gamma_i\) is defined as
\[
\gamma_i = \sqrt{\frac{s_{i,1}s_{i,2}s_{i,3} + \beta}{M}} \tag{26}
\]
where \(M\) is the number of pixels in the 3-D blocks. \(\gamma_i\) is also related to the intrinsic structure in the 3-D block. For smooth areas, the values of \(s_{i,1}, s_{i,2}, \) and \(s_{i,3}\) will be close because the pixel values are similar in smooth areas. So, the kernel size should be large, which is decided by \(\gamma_i\). Through a large kernel, more pixels can be used to reconstruct the central pixel in the 3-D block. In edge or texture areas, the pixel values in the 3-D block will have large differences and \(s_{i,1}, s_{i,2}, \) and \(s_{i,3}\) in different directions are also quite different. Then, the kernel will be small.

Fig. 3 displays the estimated 3DGK from some typical geometrical structures in the Pavia University (PaviaU) dataset, such as smoothness, edge, and texture, for visual analysis. It can be observed that the structure of the estimated kernel is consistent with that in the corresponding 3-D block in Fig. 3. Because the weights within the 3-D kernel are invisible in 3-D space, we project the estimated 3-D kernels into 2-D space along with three straightforward directions. The horizontal and longitudinal directions are along the spatial dimensions of the 3-D blocks, respectively. The vertical direction is consistent with the spectral dimension. For example, we can see that larger weights mainly concentrate on the center. In smooth blocks, most pixel values are similar. Thus, the weights are mainly dependent upon distance in position, which means the weights are larger for the pixels closer to the center. For the edge block, the projections on horizontal and longitudinal directions reflect that the neighboring pixels are more similar to the center pixel along the vertical direction, namely spectral dimension. Because the similarity is high among adjacent bands. In the vertical projection, the structure looks like the spatial information in the edge block. For the texture block, larger weights are obtained along spectral dimension due to the complex structures in the spatial domain. We can see that the structures in 3-D blocks are dominated by the edge for edge blocks. For texture blocks, their structures are complex and there are no obvious principal directions. Naturally, the kernel area in the vertical projection of the texture blocks is small.

Through 3DGK, the structure in 3-D blocks can be well exploited. Therefore, \(\beta_i\) that is the desired pixel value can be obtained through (6) once the 3-D kernel is referred from the local structures of HS images. Finally, HS image denoising is realized by the proposed 3DGK.

C. Complexity Analysis

The proposed denoising method is composed of two parts: the estimation of 3DGK for each pixel and the reconstruction of the denoised image. In the first part, gradient calculation and the approximation of covariance matrix are involved. The complexity is \(O(3WHB)\) for computing the gradients of the image with size \(W \times H \times B\). \(W\) and \(H\) are the width and height of HS images, respectively. \(B\) denotes the number of bands in HS images. When estimating the covariance matrix, SVD is utilized whose complexity is \(O(18NWHB)\) for total pixels. \(N\) stands for the number of pixels in the 3-D block. Thus, the computational complexity is \(O((3 + 18N)WHB)\). For the second part, the complexity is dominated by the update of \(q\) and matrix multiplication in (15). The complexity of the update of \(q\) is \(O(NWB)\) for the involved elementwise operation. For \(b\), the complexity is \(O((10N^2 + 100N)WHB)\) to solve (15). Therefore, it can be found that the second part of 3DGK dominates most of the computing time, and the computational complexity increases with the increasing of the 3-D block size for a given HS image. Fortunately, the method can be achieved in parallel for acceleration because the pixels in the HS image can be estimated independently.

\[
U_i = R_x(\theta_i) R_y(\phi_i) R_z(\psi_i) \tag{22}
\]

\[
= \begin{bmatrix}
1 & 0 & 0 \\
0 & \cos \theta_i & \sin \theta_i \\
0 & -\sin \theta_i & \cos \theta_i \\
\end{bmatrix} \begin{bmatrix}
\cos \phi_i & 0 & -\sin \phi_i \\
0 & 1 & 0 \\
\sin \phi_i & 0 & \cos \phi_i \\
\end{bmatrix} \begin{bmatrix}
\cos \psi_i & \sin \psi_i & 0 \\
-\sin \psi_i & \cos \psi_i & 0 \\
0 & 0 & 1 \\
\end{bmatrix}
\]
III. EXPERIMENTAL RESULTS AND COMPARISONS

In this section, experiments on simulated and real datasets are conducted to verify the effectiveness of the proposed method. The denoised results of the proposed methods are compared with those of some kernel-based denoising methods, including SK [36], SDGNLM [38], and AKR [40]. Besides, some methods based on local or global similarity are also considered for comparisons, such as SSAHTV [22], 3DNL [46], LRMR [47], and E3DTV [48]. The denoised results of all methods are analyzed in visual quality and then some numerical indicators are employed to evaluate the denoised images objectively. The results of simulated datasets are assessed by three indicators [49], i.e., mean peak signal-to-noise ratio (MPSNR), mean structural similarity (MSSIM), mean feature similarity (MFSIM), and spectral angle mapper (SAM). These indicators are the average of PSNR [50], SSIM [51], FSIM [52], and SAM on all bands, which are defined as

\[
\text{MPSNR} = \frac{1}{C} \sum_{c=1}^{C} \text{PSNR}_c \quad (27)
\]

\[
\text{MSSIM} = \frac{1}{C} \sum_{c=1}^{C} \text{SSIM}_c \quad (28)
\]

\[
\text{MFSIM} = \frac{1}{C} \sum_{c=1}^{C} \text{FSIM}_c \quad (29)
\]

\[
\text{SAM} = \arccos \left( \frac{\langle \mathbf{i}_i, \mathbf{j}_i \rangle}{\| \mathbf{i}_i \| \| \mathbf{j}_i \|} \right) \quad (30)
\]

where \( C \) equals to the number of bands in HS images, \( \mathbf{i}_i \) and \( \mathbf{j}_i \) are the spectral vectors of the \( i \)th pixel in the reference HS images and the denoised HS images, respectively. For the three indicators, the larger the values, the better the denoised results. CEIQ [53] and blind score (BS) [54] are considered to assess the denoised results of all methods on the real dataset. BS extracts the statistics features in HS images for assessment and has been used in many fields [55], [56]. The denoised images will be better if CEIQ is larger. However, smaller values mean better quality for BS.

In the following experiments, \( h \) in 3DGK has an important influence on denoising results, which is related to the noise level. According to the formulation in [38], we find a linear function between \( h \) and noise standard deviation \( \sigma \), expressed as:

\[
h = 4\sigma + 0.175, \quad \mu_i \text{ about kernel is set as 1. The regularization parameters } \alpha \text{ and } \beta \text{ are set as 0.01. The size of the 3-D block is } 9 \times 9 \times 9 \text{ in 3DGK. For a comprehensive analysis, the influences of some key parameters on denoised images are given in Section III-D for 3DGK. For real datasets, the 3-D block size is set as } 11 \times 11 \times 11. \text{ Because the noise level is unknown in real datasets, } h \text{ is empirically set as 0.4 and 0.25 for Indian Pines (IP) data and Urban data, respectively.}

A. Simulated Data Experiments

In the simulated data experiments, PaviaU and Washington DC Mall (WDCM) datasets are used to evaluate the performance of the proposed methods visually and quantitatively. PaviaU dataset is collected by the reflective optics system imaging spectrometer sensor, whose spatial resolution is 1.3 m. The spatial resolution of the WDCM dataset is 2.8 m, which is acquired by the hyperspectral digital imagery collection experiment (HYDICE) sensor in 1995. Owing to space limitation, a subimage with the size 256 \( \times \) 256 \( \times \) 191 is cropped from WDCM data as the reference image. For the PaviaU data, some bands are corrupted by heavy noises, which cannot be viewed as reference images. So, these bands in PaviaU data are removed and the size of the selected subimage is 256 \( \times \) 256 \( \times \) 98. The two datasets are shown in Fig. 4. For the simulated data, the pixel values of all bands are normalized into \([0, 1]\).
data experiments, Gaussian noises are added into HS images according to the following three cases.

Case 1: The noise distribution is the same in all bands of WDCM and PaviaU datasets, which obeys the zero-mean Gaussian distribution with fixed variance. Different noise levels are considered and the standard deviations of the Gaussian noises are 0.05, 0.1, 0.15, and 0.2.

Case 2: Different noises with zero-mean Gaussian distribution are added into different bands of WDCM and PaviaU datasets. The variance for each band is selected randomly from the range $[0.05, 0.2]$.

Case 3: In this case, multiplicative Poisson noise is introduced into all bands and the variance is 4. Besides, Gaussian noise is also added, whose mean and standard deviation are 0 and 0.05, respectively.

For the PaviaU dataset, the numerical evaluation values of the denoised results from different methods are given in Table I, in which the best values are labeled in bold. Because the signal-to-noise ratio of noise images decreases with the increase of the noise level, the values of MPSNR, MSSIM, MFSIM, and SAM gradually decrease for all methods in Table I when the noise level increases from 0.05 to 0.2. For different noise levels, the proposed method 3DGK can provide a better performance in the indicators on the whole. For the noise HS image in Case 2, the denoised results have a better performance than the noise HS images with noise levels 0.15 and 0.2 for almost all methods. Besides, Fig. 5 also presents the PSNR values of all bands of the denoised results from all methods. From Fig. 5, one can see that the proposed method has better performance for the first 70 bands. LRMR [47] behaves well on the last several bands. Besides, the denoised results for the HS image with noise level 0.05 are shown in Fig. 6. False-color composites are displayed in Fig. 6 for a more direct visual analysis, which are composed of three bands (661, 633, and 475 nm) from the denoised HS images. Besides, Fig. 6 also reports the error maps between the

---

**TABLE I**

| Noise case | Evaluation index | SSAHTV [22] | SK [36] | AKR [40] | SDGNLM [38] | 3DNLN [46] | LRMR [47] | 3DRTV [48] | 3DGK |
|------------|------------------|-------------|---------|---------|-------------|-----------|---------|-----------|-------|
| 0.05       | MPSNR            | 33.6725     | 31.5101 | 31.4176 | 32.1640     | 31.9832   | 33.5004 | 33.7577   | 34.2295|
|            | MSSIM            | 0.7675      | 0.7072  | 0.7416  | 0.6911      | 0.6726    | 0.7544  | 0.8018    | 0.8045 |
|            | MFSIM            | 0.8788      | 0.8393  | 0.8416  | 0.8436      | 0.8586    | 0.8848  | 0.8579    | 0.8871 |
|            | SAM              | 5.4107      | 5.7939  | 6.4293  | 6.7265      | 7.4715    | 6.9214  | 4.1178    | 5.4586 |
| 0.1        | MPSNR            | 29.3561     | 27.9395 | 28.1689 | 28.4688     | 25.0571   | 29.9492 | 31.0274   | 31.1218|
|            | MSSIM            | 0.5993      | 0.5481  | 0.5685  | 0.4986      | 0.4189    | 0.6389  | 0.6852    | 0.6965 |
|            | MFSIM            | 0.7986      | 0.7605  | 0.7703  | 0.7760      | 0.7370    | 0.8347  | 0.7799    | 0.8354 |
|            | SAM              | 7.88324     | 11.2004 | 9.9468  | 10.3056     | 16.6787   | 9.6053  | 5.4130    | 6.2209 |
| 0.15       | MPSNR            | 26.6273     | 24.7365 | 26.0735 | 26.0486     | 24.5043   | 27.9645 | 29.2208   | 29.5701|
|            | MSSIM            | 0.4940      | 0.4181  | 0.4625  | 0.3644      | 0.3804    | 0.5639  | 0.6060    | 0.6261 |
|            | MFSIM            | 0.7648      | 0.6815  | 0.7223  | 0.7240      | 0.7251    | 0.8012  | 0.7521    | 0.8027 |
|            | SAM              | 12.1648     | 16.6666 | 13.1466 | 10.8432     | 17.3134   | 11.6406 | 6.3923    | 7.2314 |
| 0.2        | MPSNR            | 24.1300     | 22.3278 | 24.5129 | 24.3475     | 20.3892   | 26.5248 | 28.0857   | 28.5604|
|            | MSSIM            | 0.4000      | 0.3259  | 0.3883  | 0.2747      | 0.2416    | 0.5042  | 0.5238    | 0.5735 |
|            | MFSIM            | 0.7272      | 0.6181  | 0.6867  | 0.6847      | 0.6270    | 0.7728  | 0.6708    | 0.7777 |
|            | SAM              | 15.3350     | 21.7221 | 15.9429 | 14.2127     | 26.4707   | 13.3258 | 7.1762    | 8.0320 |
| [0.05, 0.2]| MPSNR            | 26.6560     | 25.6899 | 26.9524 | 26.6557     | 25.7579   | 28.6680 | 30.5948   | 30.0146|
|            | MSSIM            | 0.4920      | 0.4659  | 0.5115  | 0.3840      | 0.4699    | 0.5895  | 0.6679    | 0.6464 |
|            | MFSIM            | 0.7656      | 0.7009  | 0.7415  | 0.6977      | 0.7616    | 0.8066  | 0.7664    | 0.8123 |
|            | SAM              | 12.8683     | 17.1400 | 12.4676 | 11.4792     | 13.8124   | 12.7970 | 5.8974    | 7.4059 |
| Gaussian   | MPSNR            | 20.9305     | 20.8533 | 20.6531 | 20.9144     | 21.0016   | 21.0001 | 20.0410   | 21.0433|
| Poisson    | MSSIM            | 0.6084      | 0.6296  | 0.6086  | 0.6176      | 0.6513    | 0.6785  | 0.7114    | 0.7187 |
|            | MFSIM            | 0.8266      | 0.8375  | 0.7946  | 0.8418      | 0.8275    | 0.8856  | 0.8521    | 0.8886 |
|            | SAM              | 8.9055      | 8.7773  | 8.4685  | 8.8983      | 8.7382    | 8.7974  | 8.6888    | 8.2761 |

Fig. 4. Simulated datasets. (a) PaviaU dataset. (b) WDCM dataset.

Fig. 5. PSNR values of each band of PaviaU data with noise level 0.05.
Fig. 6. Denoised results and error maps of different methods on PaviaU dataset. The error maps in the second row and fourth row correspond the images in the first row and third row, respectively. (a) Reference image. (b) Noise image. (c) SSAHTV [22]. (d) SK [36]. (e) AKR [40]. (f) SDGNLM [38]. (g) 3DNLM [46]. (h) LRMR [47]. (i) E3DTV [48]. (j) 3DGK.

reference HS image and the denoised HS images from different methods on the 661-nm band. The reference image and noise image are displayed in Fig. 6(a) and (b), respectively. The denoised images for all methods are shown in Fig. 6(c)–(j). From Fig. 6, we can see that the noises in the HS image are removed visibly for all methods when compared with the noise image in Fig. 6(b). For SSAHTV [22], the noises are suppressed but some artificial effects are introduced, especially in the smoothing areas. For Fig. 6(d), the blurring of the image can be observed and the noises in Fig. 6(e) are not eliminated well. For the result of SDGNLM [38], although the noises are efficiently removed, the spatial details and textures are also oversmoothed. For the denoised images in Fig. 6(g) and (h), some noises can be still found. The result of the proposed method has a better visual performance. Besides, a region containing buildings and vegetation is selected for a comparison, which is magnified and put on the bottom right corner of the image. From the enlarged region, we can see that the details in Fig. 6(d) and (e) are more blurry than those in Fig. 6(g) and (h). For the region in Fig. 6(j), the spatial information is blurred and the noises are suppressed well. Besides, we can see that the reconstruction errors of the result from the proposed method are relatively small from the error maps in Fig. 6.

Moreover, some pixels are chosen and their spectral residual curves are plotted in Fig. 7. From Fig. 7, it can be found that the spectral differences are considerable between the reference curve and that of SDGNLM [38]. The curve of 3DNLM [46] also has an obvious difference from the reference curve. Compared with other difference curves, the spectral residual of 3DGK is relatively smaller, which can preserve the spectral features more efficiently.

For the WDCM dataset, the numerical indexes are reported in Table II. Similar results can be seen in Table II, where the denoising performance degrades with the increase of noise level. Compared with other methods, the proposed method provides better performance in the three indicators on the whole. The
The denoised images of all methods are compared and one band is selected and exhibited in Fig. 9, in which the false-color composites of HS images are made up of three bands (2320, 1910, 590 nm). Moreover, Fig. 9 also presents the error maps between the reference HS image and the denoised HS images from different methods on the 2320-nm band. The reference band and the noised band are placed in Fig. 9(a) and (b). In Fig. 9(d) and (e), the blurring effects are visible, which may be caused by improper spatial and spectral structure estimation. Because the spectral structures of HS images are ignored in SK [36] and AKR [40]. For the result of SDGNLM [38], some textures in flat areas are obviously suppressed. In Fig. 9(g) and (h), the noises in some homogenous regions are not eliminated adequately. For the proposed method, the spatial structures are maintained well. A homogeneous area is chosen and enlarged for visual analysis in Fig. 9 and the enlarged area is put on the bottom right corner of the denoised image. For the local area in Fig. 9(c), some spatial information is distorted. The results in Fig. 9(d) and (e) are also blurred. It can be observed that many details are lost in the selected area of Fig. 9(f) although the noises are removed. Besides,

PSNR curves from different methods are also displayed in Fig. 8. In Fig. 8, it can be seen that the PSNR values of AKR [40] are lower than those of other methods. The proposed method can produce better performance.
Fig. 9. Denoised results and error maps of different methods on WDCM dataset. The error maps in the second row and fourth row correspond the images in the first row and third row, respectively. (a) Reference image. (b) Noise image. (c) SSAHTV [22]. (d) SK [36]. (e) AKR [40]. (f) SDGNLM [38]. (g) 3DLNM [46]. (h) LRMR [47]. (i) E3DTV [48]. (j) 3DGK.

Fig. 10. Spectral residuals of different pixels in WDCM dataset of different methods before and after denoising. (a) Pixel (80,10). (b) Pixel (215,10). (c) Pixel (205,80). (d) Pixel (215,110).

Some noises can be seen in the area of Fig. 9(h). For the proposed method, some blur effects can be found in the magnified region. From Fig. 9, it can be observed that smaller reconstruction errors are achieved for the proposed method. Fig. 10 displays the residual curves on different locations compared with the reference spectral signatures. From Fig. 10, we can see that although the noises are removed for SDGNLM [38], the spectral features are also damaged. For most methods, the consistency is better when the band number is larger than 100 except SDGNLM [38]. For the proposed method, the residual
The denoising results of the Urban dataset are shown in Fig. 13 and three bands (1470, 1550, and 1650 nm) are provided for comparison. There exist some striping noises and mixed noises in the Urban dataset. From Fig. 13, we can see that the striping noises are suppressed in the results of different methods. But the spatial details are oversmoothed in the results of SDGNLM [38] and 3DNLM [46]. In Fig. 13(i), the stripes can be still viewed from the denoised result in the first row. For the results of AKR [40], LRMR [47], and 3DGK behave well in visual performance. But the region in the last row of Fig. 12, the results of AKR [40], LRMR [47], and 3DNLM [46], are still blurry severely. For the results in Fig. 12(c) and (d), the noises are removed but some blurring effects arise. Fig. 12(g) provides better denoising results, which may be benefited from the constraint of low rank. For the results of the proposed method, spatial information is preserved but some noises can be still found from the image in the middle row. Besides, different regions in different bands are considered, which are enlarged for a direct comparison. In the selected regions of the images in the first row, we can see some spatial details or weak targets are oversmoothed in Fig. 12(f)–(i). For the images with heavy noises in the second row, the spatial information is enhanced well in Fig. 12(f)–(h), but the hue of Fig. 12(f) has an obvious difference from that of Fig. 12(a). In the last row of Fig. 12, the results of AKR [40], LRMR [47], and 3DGK behave well in visual performance. But the region in Fig. 12(d) is more blurry.

Similarly, some interesting regions are selected in the denoised results for comparison. From different regions, we can see a similar performance with the results in Fig. 12. In the second row of Fig. 13, some subtle textures are smoothed in Fig. 13(e) and (f) when removing the strips. The results of LRMR [47] have a better performance in spatial details but the noises are not suppressed well. The proposed method has a better performance in noise removal.

Besides, Table III lists the numerical results of these methods on IP and Urban datasets. Larger CEIQ represents better denoising results. For BS, the higher score means lower denoising quality. For the IP dataset, the proposed method provides better numerical values in CEIQ and BS, which means the denoised result of the proposed method behaves well in general. In the Urban dataset, the best CEIQ is from the proposed method, but LRMR [47] provides the best BS.

### C. Robustness of 3DGK

To verify the robustness of the kernel estimation on different noise levels, we use the simulated PaviaU and WDCM datasets and then calculate the difference values of the rotation angles $\theta_i$, $\phi_i$, and $\psi_i$ of the kernel for noise pixels with the corresponding angles of the kernel for clear pixels. Then, the means of all relative errors for different noise levels are computed and plotted in Fig. 14.

Fig. 14(a) demonstrates the estimation errors of the PaviaU dataset on rotation angles $\theta_i$, $\phi_i$, and $\psi_i$ when compared with its clear HS image. It can be observed that the relative errors vary with the increase of noise level. However, the relative errors of different angles are less than 0.07, which indicates the estimated kernel is robust to the noises. The relative errors of the WDCM dataset on rotation angles are shown in Fig. 14(b). From Fig. 14(b), we also can see that the relative errors are smaller. Although, the relative error of $\phi$ is greater than 0.1 for noise level with 0.2, which is affected by the heavy noises, the relative errors of $\theta$ and $\psi$ are still less than 0.1. Compared with the PaviaU dataset, the relative errors of the WDCM dataset are greater, which may be caused by the complex structures of different objects in the WDCM dataset.

### D. Parameter Analysis

In the proposed method, the smoothing parameter $h$ and the size of the 3-D block have significant effects on the denoised results. In this part, different parameter settings are analyzed and the numerical variation results of the PaviaU dataset are shown in Fig. 15, in which the dataset contains Gaussian noise with a standard deviation 0.05. In Fig. 15, $h$ increases from 0.025 to 1 with step size 0.025 and the size of 3-D block increases from $3 \times 3 \times 3$ to $11 \times 11 \times 11$ with step size 2 on each dimension. From Fig. 15, we can observe that the numerical values vary with the increase of $h$ and the size of the 3-D block. With the increasing of $h$, MPSNR, MSSIM, and MFSIM increase first and then decrease. The best values are achieved when $h$ is equal to 0.375. For the size of the 3-D block, the variations of numerical values are flat. However, there are some differences in index.
values for the size of the 3-D block. Finally, the size of the 3-D block is set as $9 \times 9 \times 9$. Thus, $h$ and block size are set as 0.375 and $9 \times 9 \times 9$ for the PaviaU dataset corrupted by Gaussian noise with a standard deviation 0.05.

### E. Running Time

In Table IV, we report the running time of all methods on different datasets for a comprehensive analysis. All methods are implemented on the same computer with Intel Core i7-6700 processor, 3.4 GHz, and 16 GB memory by MATLAB R2017a. From Table IV, it can be observed that regression-based methods, such as SK [36] and AKR [40], spend more time. Besides, 3DNLM [46] is also time-consuming because of the neighbor search in local areas. SSAHTV [22] and LRMR [47] are faster when compared with the proposed method. Compared with SDGNLM [38] and 3DNLM [46], 3DGK has low computational complexity.
Fig. 14. Variations of rotation angles with the increasing noise level. (a) PaviaU dataset. (b) WDCM dataset.

Fig. 15. Influences of smoothing parameter and the size of 3-D block on the denoised result of PaviaU dataset. (a) MPSNR. (b) MSSIM. (c) MFSIM.

### TABLE IV

*TIME COMPARISON OF ALL METHODS ON DIFFERENT DATASET (S)*

| Method   | Reduced-Scale Dataset | Full-Scale Dataset |
|----------|-----------------------|--------------------|
|          | PaviaU | WDCM | IP | Urban |
| SSAHTV[22] | 84.06   | 172.95 | 42.64 | 256.19 |
| SK[36] | 461.58  | 897.51 | 342.76 | 1434.26 |
| AKR[40] | 488.41  | 921.71 | 351.94 | 2261.63 |
| SDGNI.M[38] | 4191.46 | 8096.49 | 2889.04 | 12898.83 |
| 3DNL.M[46] | 4508.61 | 5670.37 | 2220.2 | 7461.69 |
| LSUM[47] | 42.64   | 100.45 | 25.54 | 104.37 |
| E3D TV[48] | 54.27   | 113.24 | 40.55 | 186.33 |
| 3DGK | 518.07  | 954.09 | 381.86 | 2570.39 |

### IV. CONCLUSION

In this article, an HS image denoising method is developed by extending the kernel regression model to the 3-D formulation, which makes effective use of the spatial and spectral geometrical structures in HS images. In the proposed method, 3DGK is established by estimating the structure and orientation in a 3-D block, whose information is inferred from its gradients. Then, 3DGK is adjusted adaptively by scaling, elongation, and rotation to approximate the inner structure of the 3-D block. So, more proper weights can be assigned to the adjacent pixels in the 3-D block. Finally, the denoised pixels are obtained by efficiently minimizing the weighted $L_1$-norm constraint through ALM. Experiments are conducted on simulated and real datasets. Compared with some methods based on LS prior, the proposed method has a better performance in qualitative and quantitative evaluations. In future work, the geometrical structure captured by 3DGK will be further explored and combined with other efficient priors, such as TV.

### REFERENCES

[1] J. Liu, Z. Wu, J. Li, A. Plaza, and Y. Yuan, “Probabilistic-kernel collaborative representation for spatial-spectral hyperspectral image classification,” *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 4, pp. 2371–2384, Apr. 2016.

[2] S. Yang, Z. Shi, and W. Tang, “Robust hyperspectral image target detection using an inequality constraint,” *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 6, pp. 3389–3404, Jun. 2015.

[3] L. Tong, J. Zhou, B. Qian, J. Yu, and C. Xiao, “Adaptive graph regularized multilayer nonnegative matrix factorization for hyperspectral unmixing,” *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 13, pp. 434–447, Jan. 2020.

[4] F. Khajehrayeni and H. Ghassemian, “Hyperspectral unmixing using deep convolutional autoencoders in a supervised scenario,” *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 13, pp. 567–576, Feb. 2020.

[5] K. Qu and W. Bao, “Multiple-priors ensemble constrained nonnegative matrix factorization for spectral unmixing,” *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 13, pp. 963–975, Feb. 2020.

[6] L. Sun, F. Wu, T. Zhan, W. Liu, J. Wang, and B. Jeon, “Weighted nonlocal low-rank tensor decomposition method for sparse unmixing of hyperspectral images,” *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 13, pp. 1174–1188, Mar. 2020.

[7] M. Ye and Y. Qian, “Mixed poisson-Gaussian noise model based sparse denoising for hyperspectral imagery,” in *Proc. 4th Workshop Hyperspectral Image Signal Process.*, Evol. Remote Sens., Jun. 2012, pp. 1–4.

[8] J. Li, Q. Yuan, H. Shen, and L. Zhang, “Noise removal from hyperspectral image with joint spectral-spatial distributed sparse representation,” *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 9, pp. 5425–5439, Sep. 2016.

[9] M. Elad and M. Aharon, “Image denoising via sparse and redundant representations over learned dictionaries,” *IEEE Trans. Image Process.*, vol. 15, no. 12, pp. 3736–3745, Dec. 2006.

[10] Y. Qian and M. Ye, “Hyperspectral imagery restoration using nonlocal spectral-spatial structured sparse representation with noise estimation,” *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 6, no. 2, pp. 499–515, Apr. 2013.
[33] W. Dong, H. Wang, F. Wu, G. Shi, and X. Li, “Deep spatial-spectral representation learning for hyperspectral image denoising,” IEEE Trans. Comput. Imag., vol. 5, no. 4, pp. 635–648, Dec. 2019.

[34] K. Wei, Y. Fu, and H. Huang, “3-D quasi-recurrent neural network for hyperspectral image denoising,” IEEE Trans. Neural Netw. Learn. Syst., vol. 32, no. 1, pp. 363–375, Jan. 2021.

[35] A. Maffei, J. Haut, M. Paolletti, J. Plaza, L. Bruzzone, and A. Plaza, “A single model CNN for hyperspectral image denoising,” IEEE Trans. Geosci. Remote Sens., vol. 58, no. 4, pp. 2516–2529, Apr. 2020.

[36] H. Takeda, S. Farsiu, and P. Milanfar, “Kernel regression for image processing and reconstruction,” IEEE Trans. Image Process., vol. 16, no. 2, pp. 349–366, Jan. 2007.

[37] P. Bouboulis, K. Slavakis, and S. Theodoridis, “Adaptive kernel-based image denoising employing semi-parametric regularization,” IEEE Trans. Image Process., vol. 19, no. 6, pp. 1465–1479, Jun. 2010.

[38] K. Zhang et al., “Sparse double-geometric nonlocal mean image recovery via steerable kernel,” Appl. Soft Comput., vol. 33, pp. 77–85, Aug. 2015.

[39] P. Zhong and R. Wang, “Jointly learning the hybrid CRF and MLR model for simultaneous denoising and classification of hyperspectral imagery,” IEEE Trans. Neural Netw. Learn. Syst., vol. 25, no. 7, pp. 1319–1334, Jul. 2014.

[40] H. Liu, Z. Zhang, L. Xiao, and Z. Wei, “Adaptive tensor matrix based kernel regression for hyperspectral image denoising,” in Proc. IEEE Geosci. Remote Sens. Symp., Quebec City, QC, Canada, Jul. 2014, pp. 4616–4619.

[41] W. Hardle, Applied Nonparametric Regression. Cambridge, U.K.: Cambridge Univ. Press, 1990.

[42] Z. Lin, M. Chen, L. Wu, and Y. Ma, “The augmented Lagrange multiplier method for exact recovery of corrupted low-rank matrices,” Univ. Illinois Urbana-Champaign, Champaign, IL, USA, Tech. Rep. UILU-ENG-09-2215, 2009.

[43] J.-F. Cai, E. J. Candès, and Z. Shen, “A singular value thresholding algorithm for matrix completion,” SIAM J. Optim., vol. 20, no. 4, pp. 1956–1982, 2010.

[44] Q. Wang, L. Zhang, Q. Tong, and F. Zhang, “Hyperspectral imagery denoising based on oblique subspace projection,” IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens., vol. 7, no. 6, pp. 2468–2480, Jun. 2014.

[45] X. Feng and P. Milanfar, “Multiscale principal components analysis for image local orientation estimation,” in Proc. 36th Asilomar Conf. Signals, Syst., Comput., Pacific Grove, CA, USA, Nov. 2002, pp. 478–482.

[46] Y. Qian, Y. Shen, M. Ye, and Q. Wang, “3D nonlocal means filter with noise estimation for hyperspectral imagery denoising,” in Proc. IEEE Int. Geosci. Remote Sens. Symp., Munich, Germany, Jul. 2012, pp. 1345–1348.

[47] H. Zhang, W. He, L. Zhang, H. Shen, and Q. Yuan, “Hyperspectral image restoration using low-rank matrix recovery,” IEEE Trans. Geosci. Remote Sens., vol. 52, no. 8, pp. 4729–4743, Aug. 2014.

[48] J. Peng, Q. Xie, Q. Zhao, Y. Wang, L. Yee, and D. Meng, “Enhanced 3DTV denoising and its applications on HSI denoising and compressed sensing,” IEEE Trans. Image Process., vol. 29, pp. 7890–7903, Jul. 2020.

[49] W. He, H. Zhang, L. Zhang, and H. Shen, “Total-variation-regularized low-rank matrix factorization for hyperspectral image restoration,” IEEE Trans. Geosci. Remote Sens., vol. 54, no. 1, pp. 176–188, Jan. 2016.

[50] Q. Huynh-Thu and M. Ghanbari, “Scope of validity of PSNR in image/video quality assessment,” Electron. Lett., vol. 44, no. 13, pp. 800–801, Jun. 2008.

[51] Z. Wang, A. C. Bovik, and H. R. Sheikh, “Image quality assessment: From error visibility to structural similarity,” IEEE Trans. Image Process., vol. 13, no. 4, pp. 600–612, Apr. 2004.

[52] L. Zhang, L. Zhang, X. Mou, and D. Zhang, “FSIM: A feature similarity index for image quality assessment,” IEEE Trans. Image Process., vol. 20, no. 8, pp. 2378–2386, Aug. 2011.

[53] M. Xu and Z. Wang, “No-reference quality assessment of contrast-distorted images,” in Proc. IEEE Int. Conf. Signal Process. Image, Beijing, China, Aug. 2016, pp. 362–367.

[54] J. Yang, Y. Zhao, C. Yi, and J. C.-W. Chan, “No-reference hyperspectral image quality assessment via quality-sensitive features learning,” Remote Sens. Lett., vol. 8, no. 4, pp. 308–316, 2017.

[55] K. Zhang, M. Wang, S. Yang, and L. Jiao, “Spatial-spectral-graph-regularized low-rank tensor decomposition for multispectral and hyperspectral image fusion,” IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens., vol. 11, no. 4, pp. 1030–1040, Apr. 2018.

[56] H. Fan, Y. Chen, Y. Guo, H. Zhang, and G. Kuang, “Hyperspectral image restoration using low-rank tensor recovery,” IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens., vol. 10, no. 10, pp. 4589–4604, Oct. 2017.
Feng Zhang received the B.S. degree in electronic information engineering from Shandong Normal University, Jinan, China, in 2012, and the M.Sc. degree in electronic information engineering from Xidian University, Xi’an, China, in 2016. She is currently working toward the Ph.D. degree in computer science and technology with the School of Information Science and Engineering, Shandong Normal University. Her research interests include deep learning and image processing.

Kai Zhang was born in Shanxi, China, in 1992. He received the B.S. degree in electrical engineering and automation from North University of China, Taiyuan, China, in 2013, and the Ph.D. degree in circuit and system from Xidian University, Xi’an, China, in 2018.

He is currently a Lecturer with the School of Information Science and Engineering, Shandong Normal University, Jinan, China. His research interests include multisource remote sensing image fusion, matrix factorization, and deep learning.

Jiande Sun received the Ph.D. degree in communication and information system from Shandong University, Jinan, China, in 2000 and 2005, respectively.

From 2008 to 2009, he was a Visiting Researcher with the Institute of Telecommunications System, Technical University of Berlin, Berlin, Germany. From 2010 to 2012, he was a Postdoctoral Researcher with the Institute of Digital Media, Peking University, Beijing, China, and also with the State Key Laboratory of Digital-Media Technology, Hisense Group, Qingdao, China. From 2014 to 2015, he was a DAAD Visiting Researcher with the Technical University of Berlin and the University of Konstanz, Konstanz, Germany. From 2015 to 2016, he was a Visiting Researcher with the School of Computer Science, Language Technology Institute, Carnegie Mellon University, Pittsburgh, PA, USA. He is currently a Professor with the School of Information Science and Engineering, Shandong Normal University, Jinan, China. He has authored/coauthored more than 60 journal and conference papers. He is the coauthor of two books. His research interests include multimedia content analysis, video hashing, gaze tracking, image/video watermarking, and 2-D to 3-D conversion.

Zhixi Feng received the B.S. degree in automation from Lanzhou University of Technology, Lanzhou, China, in 2012, and the Ph.D. degree in intelligent information processing from Xidian University, Xi’an, China, in 2018.

He is currently an Associate Professor with School of Artificial Intelligence, Xidian University. His research interests include hyperspectral image processing and machine learning.

Wenbo Wan received the Ph.D. degree in information and communication engineering from Shandong University, Jinan, China, in 2015, supervised by Prof. J. Liu.

From June 2019 to October 2019, he was a Visiting Researcher with the Department of Computer Science, City University of Hong Kong, Hong Kong. He is currently an Associate Professor with the School of Information Science and Engineering, Shandong Normal University, Jinan, China. His research interests include multimedia security, multimedia quality assessment, and image/video watermarking.

Quanyuan Wu was born in 1959. He received the Ph.D. degree in geodesy and surveying engineering from Shandong University of Science and Technology, Qingdao, China, in 2007.

Since 2003, he has been a Professor with the Department of Geography, College of Geography and Environment, Shandong Normal University, Jinan, China. His research interests include the development of remote sensing and geographic information techniques.

Dr. Wu is a member of the Education and Science Popularization Committee of the China GIS Association, and the Executive Director of the Shandong Remote Sensing Society. He presided over or participated in three Nation Natural Science Foundations of China.