

# Hyperspectral image segmentation using 3D regularized subspace clustering model

Carlos Hinojosa<sup>✉</sup>,\* Fernando Rojas, Sergio Castillo, and Henry Arguello  
Universidad Industrial de Santander, Department of Computer Science, Bucaramanga, Colombia

**Abstract.** The accurate segmentation of remotely sensed hyperspectral images has widespread attention in the Earth observation and remote sensing communities. In the past decade, most of the efforts focus on the development of different supervised methods for hyperspectral image classification. Recently, the computer vision community is developing unsupervised methods that can adapt to new conditions without leveraging expensive supervision. In general, among unsupervised classification methods, sparse subspace clustering (SSC) is a popular tool that achieves good clustering results on experiments with real data. However, for the specific case of hyperspectral clustering, the SSC model does not take into account the spatial information of such images, which limits its discrimination capability and hampering the spatial homogeneity of the clustering results. As a solution, we propose to incorporate a regularization term to the SSC model, which takes into account the neighboring spatial information of spectral pixels in the scene. Specifically, the proposed method uses a three-dimensional (3D) Gaussian filter to perform a 3D convolution on the sparse coefficients, obtaining a piecewise-smooth representation matrix that enforces an averaging constraint in the SSC optimization program. Extensive simulations demonstrate the effectiveness of the proposed method, achieving an overall accuracy of up to 99% in the selected hyperspectral remote sensing datasets. © 2021 Society of Photo-Optical Instrumentation Engineers (SPIE) [DOI: [10.1117/1.JRS.15.016508](https://doi.org/10.1117/1.JRS.15.016508)]

**Keywords:** hyperspectral image segmentation; subspace clustering; sparse subspace clustering; optimization algorithms.

Paper 200539 received Jul. 18, 2020; accepted for publication Jan. 7, 2021; published online Jan. 28, 2021.

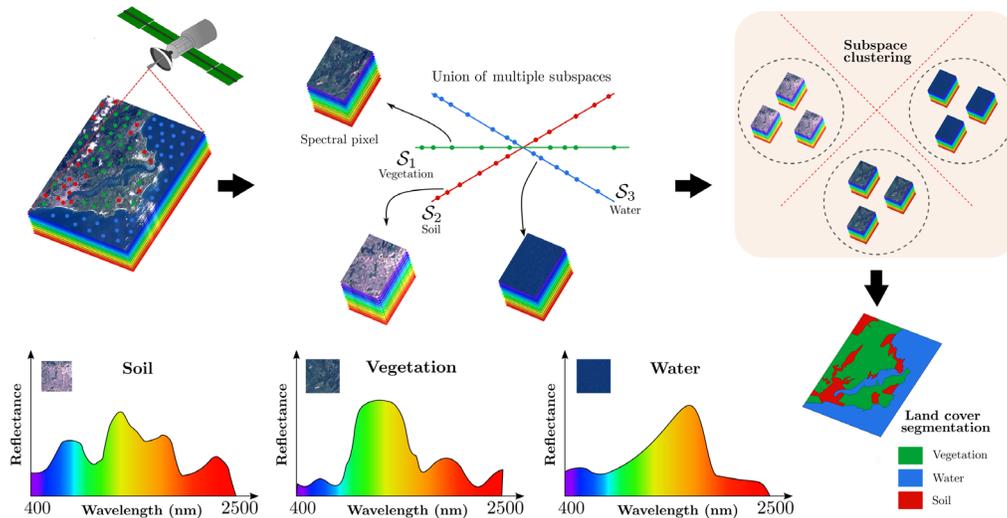
## 1 Introduction

Hyperspectral imaging sensors collect spectral information at every spatial location of a scene. The acquired high dimensional data are commonly regarded as three-dimensional (3D) images, where two of the coordinates correspond to the spatial domain  $(x, y)$  and the third one represents the spectral wavelengths  $(\lambda)$ .<sup>1,2</sup> Such 3D images are known as hyperspectral images, which we also will refer to them as hyperspectral images (HSI) through the paper. Every spatial location in a hyperspectral image is represented by a vector, also known as spectral signature of the pixel or spectral pixel, whose values correspond to the intensity at different spectral bands, as shown in Fig. 1. Since materials reflect different electromagnetic energy at specific wavelengths,<sup>1,3</sup> the information provided by the spectral signatures allows for better characterization and discrimination of the objects within the scene. In remote sensing, the classification of HSI is an important computer vision task for many practical applications, such as precision agriculture,<sup>4</sup> vegetation classification,<sup>5-7</sup> monitoring and management of the environment,<sup>8,9</sup> as well as security and defense.<sup>10,11</sup> Due to the cost of both data labeling and algorithm supervision, the development of unsupervised methods that can adapt to new conditions has drawn the attention of Earth observation and remote sensing communities.

In general, the accurate unsupervised classification of the spectral pixels is challenging due to the high-dimensional feature space. In the literature, many different clustering methods for HSI land-cover segmentation have been proposed. The fuzzy c-mean<sup>12</sup> (FCM) is one the early and most used method for image segmentation, and its success relies on the introduction of fuzziness

---

\*Address all correspondence to Carlos Hinojosa, [carlos.hinojosa@saber.uis.edu.co](mailto:carlos.hinojosa@saber.uis.edu.co)



**Fig. 1** Hyperspectral image land cover segmentation via subspace clustering theory.

for the belongingness of each image pixels. Even though FCM is efficient for images with simple texture and background, it fails to segment images with complex structures, e.g., HSI, or images corrupted by noise since it does not take advantage of the spatial information. In this respect, authors in Ref. 13 proposed the spatial fuzzy c-mean (FCM\_S1), which employ average filtering and median filtering to obtain the spatial neighborhood information in advance.

Among recent works on HSI unsupervised classification, the robust manifold matrix factorization (RMMF) is based on unified low-rank matrix factorization and performs dimensionality reduction in conjunction with data clustering.<sup>14</sup> The authors also designed an augmented Lagrangian method-based procedure to find the optimal local solution of the proposed optimization. Moreover, in Ref. 15, the authors presented the 3D-CAE, an end-to-end approach to segment HSI and offer high-quality segmentation in an unsupervised way. In particular, the paper introduces a deep learning technique that uses a 3-D convolutional autoencoder to learn embedded features with later undergo clustering, which is performed during the network training with a clustering-oriented loss.

On the other hand, during the last decade, some of the most successful unsupervised learning methods first find low-dimensional subspaces that represent the high-dimensional features before performing any further processing. For instance, subspace clustering (SC) refers to the task of separating data according to their underlying subspaces. Among SC algorithms, the sparse subspace clustering (SSC) model is one of the most successful and has been used in applications such as image representation, and motion and image segmentation.<sup>16,17</sup> SSC captures the global geometric relationship among all data points by expressing each one of them as a linear combination of all others. Then, the set of solutions is restricted to be sparse by minimizing the  $\ell_1$  norm of the representation coefficient matrix.<sup>18</sup> Finally, an affinity matrix is built using the obtained sparse coefficients, and the normalized spectral clustering algorithm<sup>19</sup> is applied to achieve the final segmentation. Instead of learning the underlying subspaces by expressing each point as a linear combination of all other points, some recent SC algorithms proposed to select the most representative data points before built the affinity matrix. For instance, the scalable and robust SSC (SR-SSC)<sup>20</sup> selects a few sets of anchor points using a randomized hierarchical clustering method. Then, within each set of anchor points, it solves the LASSO<sup>21</sup> problem for each data point, only allowing anchor points to have non-zero weights. Similarly, the exemplar-based subspace clustering (ESC-FFS)<sup>22</sup> proposes to select a small subset that represents all data points using the farthest first search (FFS)<sup>22</sup> algorithm, which is a modified version of the farthest-first traversal algorithm.<sup>23</sup> However, these general-purpose methods do not fully exploit the complex structure of remotely sensed hyperspectral images. Furthermore, they do not consider the rich spatial information of the spectral images, which could boost the accuracy of these algorithms on HSI imagery.

In general, the spectral signatures corresponding to a specific land-cover material would lie in the same nonlinear manifold. However, the manifold clustering represents a more complex problem than subspace (flat manifolds) clustering.<sup>24</sup> On the other hand, different previous works have been successfully modeling the hyperspectral image segmentation problem using the SC theory.<sup>25,26</sup> Those works relax the problem by assuming that the spectral variability of the signatures is approximately linear; hence, the spectral signatures with similar spectral characteristics have a high probability of lying in the same affine low-dimensional subspace. For instance, the SSC algorithm has been successfully used for land cover segmentation.<sup>27–31</sup> However, the traditional SSC model only captures the relationship between pixels by analyzing the spectral features and does not take advantage of the spatial information. Considering that spectral pixels belonging to the same land cover material are arranged in common regions within the HSI, the sparse representation matrix of the SSC model should also account for the spatial relationship between neighboring pixels. Indeed, it is expected that the obtained representation coefficients of two adjacent pixels would be very similar since they capture the geometric relationship among pixels.

Recently, few works have proposed to take into account the neighboring spatial information by incorporating a regularization term in the SSC optimization problem, which enforces a local averaging constraint on the sparse coefficient matrix.<sup>25,26,29</sup> However, the regularization term proposed in such works assigns the same sparse basis to adjacent pixels by performing independent two-dimensional (2D) smoothing operations over one column vector of the SSC representation matrix without taking advantage of the information among columns. In this work, we propose to efficiently integrate the spatial-contextual information of the HSI in the SSC model by performing a 3D convolution operation with a 3D Gaussian filter over the representation coefficient matrix. This procedure provides a more accurate coefficient matrix by taking into account the information among adjacent representation coefficient column vectors of the SSC matrix to enforce an averaging constraint, which preserves the similarity and improves the clustering accuracy. We incorporate the 3D convolution into the SSC model by adding a regularization term to the optimization problem. Experimental results demonstrate that the proposed approach significantly improves the clustering performance, in both the visual and quantitative evaluations.

We organize the paper as follows: Section 2 briefly introduces the HSI land cover segmentation via SC model. In Sec. 3, we present the mathematical formulation of the proposed 3D regularized subspace clustering model (3DS-SSC) for HSI segmentation. Besides, we give a solution to the proposed optimization problem using the alternating direction method of multipliers (ADMM) algorithm. Section 4 presents the experimental results, and Sec. 5 gives the conclusions of this work.

## 2 HSI Land Cover Segmentation via Subspace Clustering Model

In recent years, different advances in processing complex high-dimensional data rely on the observation that their intrinsic dimension is often much smaller than the dimension of the ambient space.<sup>18</sup> This led to the development of different techniques for finding a low-dimensional representation, or subspace, of high-dimensional data sets.<sup>29,32</sup> In practice, all the high-dimensional data points can be modeled as being drawn from a union of multiple linear or affine subspaces where the membership of the data points to each subspace might be unknown. Then, SC refers to the problem of separating data according to their underlying subspaces. The subspaces can be linear or affine, which could be seen as a linear subspace with a constraint that the sum of all the coordinates is equal to one. One of the most popular SC algorithms is the SSC.<sup>32</sup> The key idea behind SSC is that among the infinitely many possible representations of a data point in terms of the others, a sparse representation corresponds to selecting a few points from the same subspace.

Assuming that HSI's pixels are drawn from a union of multiple subspaces and spectral pixels with a similar spectrum belong to the same land cover material, SSC can be effectively applied to perform unsupervised land cover segmentation. Denote a spectral pixel  $\mathbf{x}_j$  as a  $D$ -dimensional vector after lexicographically reordering the  $N_r \times N_c \times D$  3D data cube into a 2D matrix as

$\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_j, \dots, \mathbf{x}_N] \in \mathbb{R}^D$ , where  $N = N_r N_c$ , and  $D$  refers to the number of spectral bands.<sup>29</sup> Note that each column of  $\mathbf{X}$  now corresponds to a spectral pixel (signature), see Fig. 1. Under the context of SC, each column of  $\mathbf{X}$  belongs to the union of  $k$  subspaces  $\{\mathcal{S}\}_{i=1}^k$ . By taking advantage of the subspace structure, the data points in  $\mathbf{X}$  obey the so called self-expressiveness property,<sup>32</sup> i.e., each data point in a union of subspaces can be well represented by a linear combination of other points in the dataset. This can be mathematically expressed as

$$\mathbf{X} = \mathbf{X}\mathbf{Z}, \quad (1)$$

where  $\mathbf{Z} \in \mathbb{R}^{N \times N}$  is the representation coefficient matrix. The choice of  $\mathbf{Z}$  is usually not unique, and the goal is to find a certain  $\mathbf{Z}$  that be discriminative for subspace clustering. The problem in Eq. (1) is that it may have many feasible solutions, and thus the regularization is necessary to produce the solution. Motivated by the observation that the block-diagonal solution is sparse, SSC finds a sparse  $\mathbf{Z}$  by  $\ell_1$ -norm minimizing. Here, the  $\ell_1$  norm is used as the convex surrogate of the  $\ell_0$  norm, which is not convex and, otherwise, would turn the problem NP-hard.<sup>32</sup> Then, the sparse representation model can be built as follows:

$$\min_{\mathbf{Z}, \mathbf{R}} \|\mathbf{Z}\|_1 + \lambda \|\mathbf{R}\|_F^2 \quad \text{s.t. } \mathbf{X} = \mathbf{X}\mathbf{Z} + \mathbf{R}, \quad \text{diag}(\mathbf{Z}) = 0, \quad \mathbf{Z}^T \mathbf{1} = \mathbf{1}, \quad (2)$$

where  $\mathbf{R}$  stands for the representation error, and the parameter  $\lambda$  is used to trade-off the relative contribution between the sparsity of the coefficients and the magnitude of the noise. Here,  $\text{diag}(\mathbf{Z}) \in \mathbb{R}^N$  is the vector of the diagonal elements of  $\mathbf{Z}$ , and the constraint  $\text{diag}(\mathbf{Z}) = 0$  is used to eliminate the trivial solution of writing a point as an affine combination of itself.<sup>32</sup> In addition, the constraint  $\mathbf{Z}^T \mathbf{1} = \mathbf{1}$  ensures that it is a case of an affine subspace.

The optimization problem in Eq. (2) can be solved by the ADMM, and its implementation can be found in Ref. 32. The solution obtained by Eq. (2) corresponds to subspace-sparse representation of the data points, which is used by spectral clustering to infer the clustering of the data. Specifically, the clustering result is obtained by applying the spectral clustering to the Laplacian matrix induced by the similarity matrix  $\mathbf{W} \in \mathbb{R}^{N \times N}$ , defined as  $\mathbf{W} = |\mathbf{Z}| + |\mathbf{Z}|^T$ , where the operator  $|\cdot|$  stands for the absolute value. This definition of  $\mathbf{W}$  is adopted to ensure that the matrix is both  $\geq 0$  and symmetric.<sup>18</sup> The complete SSC method is summarized in Algorithm 1.

As observed in Eq. (2), the SSC model for HSI only captures the relationship of pixels by analyzing the spectral features without considering the spatial information. In the following section, we present the proposed subspace clustering approach, which successfully improves the clustering results on HSI by considering the spatial information in the optimization problem.

#### Algorithm 1 SSC algorithm<sup>32</sup>

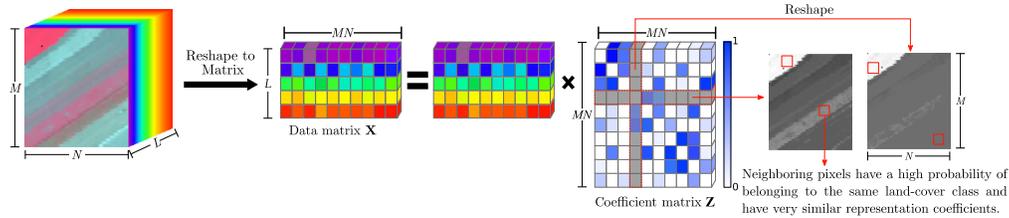
---

**Input:** HSI data  $\mathbf{X} \in \mathbb{R}^{D \times N}$ , containing a set of spectral pixels  $\{\mathbf{x}_j\}_{j=1}^N$ , in a union of  $k$  affine subspaces  $\{\mathcal{S}_i\}_{i=1}^k$ ; the number of subspaces  $k$  and the regularization parameter  $\lambda$

**Output:** Matrix  $\hat{\mathbf{X}}$  with the obtained clustering labels of  $\mathbf{X}$

**Function** SSC( $\mathbf{X}, k, \lambda$ )

- 1 Solve the sparse optimization problem in Eq. (2).
  - 2 Normalize the columns of  $\mathbf{Z}$  as  $\mathbf{z}_j \leftarrow \frac{\mathbf{z}_j}{\|\mathbf{z}_j\|_\infty}$ .
  - 3 Form a similarity matrix as  $\mathbf{W} = |\mathbf{Z}| + |\mathbf{Z}|^T$ , representing the relationship (weights) among data points.
  - 4 Apply spectral clustering<sup>19</sup> to the similarity matrix.
  - 5 return Segmentation of the data:  $\hat{\mathbf{X}}_1, \dots, \hat{\mathbf{X}}_k$ .
-



**Fig. 2** Graphical interpretation of each column and row of the sparse coefficient matrix  $\mathbf{Z}$ .

### 3 Sparse Subspace Clustering with 3D Spatial Regularization

In the SSC model, the  $j$ 'th column of the sparse representation matrix  $\mathbf{Z}$  contains the representation coefficient distribution of the whole image with respect to a single atom. Alternatively, using the spectral unmixing jargon, this can be thought of as a specific fractional abundance concerning an endmember in the unmixing domain. Similarly, the  $j$ 'th row of  $\mathbf{Z}$  should present a similar coefficient distribution due to the symmetric nature of  $\mathbf{Z}$ . The meaning of each row or column of  $\mathbf{Z}$  is better appreciated by reshaping the corresponding  $MN$  vector into a  $M \times N$  matrix, as shown in Fig. 2.

Notice that the spectral pixels belonging to the same land cover material are arranged in common regions, i.e., two spatially neighboring pixels in an HSI usually have a high probability of belonging to the same class. Hence their representation coefficients should also be very close, concerning the same sparse basis, according to the SSC mechanism. Then, in general, the obtained  $\mathbf{Z}$  should be piecewise smooth since there is a spatial relationship between the sparse representation vector of every pixel and its neighbors. This observation suggests that it is reasonable to apply a spatial constraint to  $\mathbf{Z}$  in the optimization problem. The aim of this work is to improve the representation coefficient matrix by enforcing a 3D spatial regularization, hence incorporating the contextual dependence of the pixels.

#### 3.1 Problem Formulation

As previously discussed, the spatial neighborhood information of an HSI can be incorporated into the SSC model by constraining  $\mathbf{Z}$  to be piecewise-smooth. One approach to achieve it is using a smoothing convolution that assigns similar sparse coefficients to adjacent pixels. Since it is desired that the sparse coefficient matrix obtained by the SSC to be symmetric, and hence rows and columns of  $\mathbf{Z}$  present a similar coefficient distribution, we propose to use a 3D convolution to perform the smoothing on  $\mathbf{Z}$ .

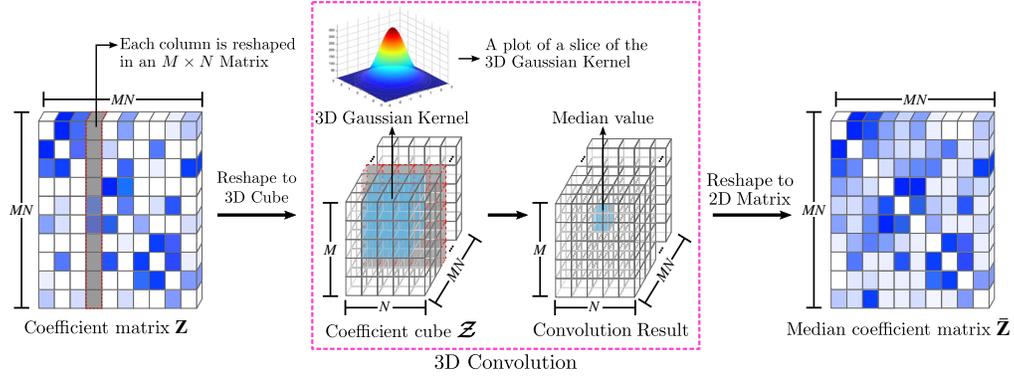
Specifically, we first rearrange the 2D coefficient matrix  $\mathbf{Z}$  to a 3-D cube  $\hat{\mathbf{Z}} \in \mathbb{R}^{N_r \times N_c \times N}$ , such that each column of  $\mathbf{Z}$  corresponds to a slice of  $\hat{\mathbf{Z}}$ . Then, we perform a 3D smoothness convolution by opening a 3D window to each spatial location of  $\hat{\mathbf{Z}}$ . The aim of using a 3D Gaussian kernel to perform the smoothness operation is to simultaneously take advantage of the information among adjacent columns/rows of  $\mathbf{Z}$ , obtaining a more accurate sparse coefficient matrix. The filtering is performed using the isotropic 3D Gaussian kernel given by

$$G_{i,j,k} = -\frac{1}{\sqrt{2\pi\sigma}} e^{-\frac{i^2+j^2+k^2}{2\sigma^2}}, \quad (3)$$

with a specific window size  $h$  and standard deviation  $\sigma$ . After the filtering process, the cube  $\hat{\mathbf{Z}}$  is rearranged into the filtered coefficient matrix  $\bar{\mathbf{Z}}$ . Figure 3 shows the proposed methodology described above.

Instead of directly imposing a smoothness constraint over  $\mathbf{Z}$ , the enhanced coefficient matrix  $\bar{\mathbf{Z}}$  can be used to regularize the solution of the SSC optimization problem described in Eq. (2). Therefore, the proposed subspace clustering model for HSI segmentation is expressed as

$$\min_{\mathbf{Z}, \mathbf{R}} \|\mathbf{Z}\|_1 + \frac{\lambda}{2} \|\mathbf{R}\|_F^2 + f_{3DS}(\mathbf{Z}) \quad \text{s.t.} \quad \mathbf{X} = \mathbf{XZ} + \mathbf{R}, \quad \text{diag}(\mathbf{Z}) = 0, \quad \mathbf{Z}^T \mathbf{1} = \mathbf{1}, \quad (4)$$



**Fig. 3** The proposed methodology to enhance the sparse representation coefficients matrix using 3D convolution.

where

$$f_{3DS}(\mathbf{Z}) = \frac{\alpha}{2} \|\mathbf{Z} - \bar{\mathbf{Z}}\|_F^2, \quad (5)$$

corresponds to the 3D spatial regularization term.

### 3.2 Optimization

To solve the optimization problem in Eq. (4), we develop an algorithm based on the ADMM scheme.<sup>33</sup> First, an auxiliary matrix  $\mathbf{U} \in \mathbb{R}^{MN \times MN}$  with the same size as the sparse coefficient matrix  $\mathbf{Z}$  is defined to separate the variables. In this way, we only need to solve the following optimization problem:

$$\min_{\mathbf{Z}, \mathbf{U}, \bar{\mathbf{Z}}} \|\mathbf{Z}\|_1 + \frac{\lambda}{2} \|\mathbf{X} - \mathbf{XU}\|_F^2 + \frac{\alpha}{2} \|\bar{\mathbf{Z}} - \mathbf{U}\|_F^2 \quad \text{s.t. } \mathbf{U}^T \mathbf{1} = \mathbf{1}, \quad \mathbf{U} = \mathbf{Z} - \text{diag}(\mathbf{Z}), \quad (6)$$

where we directly replace the regularization term  $f_{3DS}(\mathbf{Z})$  of Eq. (5) in Eq. (4). Note that, for convenience, in this section, we will denote by  $\text{diag}(\mathbf{Z})$  both a vector whose elements are the diagonal entries of  $\mathbf{Z}$  and a diagonal matrix whose diagonal elements are the diagonal entries of  $\mathbf{Z}$ .

Following the ADMM scheme, and using a parameter  $\rho > 0$ , we add to the objective function of Eq. (6) two penalty terms corresponding to the constraints  $\mathbf{U}^T \mathbf{1} = \mathbf{1}$  and  $\mathbf{U} = \mathbf{Z} - \text{diag}(\mathbf{Z})$ . Then, we obtain the following optimization problem:

$$\min_{\mathbf{Z}, \mathbf{U}, \bar{\mathbf{Z}}} \|\mathbf{Z}\|_1 + \frac{\lambda}{2} \|\mathbf{X} - \mathbf{XU}\|_F^2 + \frac{\alpha}{2} \|\bar{\mathbf{Z}} - \mathbf{U}\|_F^2 + \frac{\rho}{2} \|\mathbf{U}^T \mathbf{1} - \mathbf{1}\|_2^2 + \frac{\rho}{2} \|\mathbf{U} - [\mathbf{Z} - \text{diag}(\mathbf{Z})]\|_F^2$$

$$\text{s.t. } \mathbf{U}^T \mathbf{1} = \mathbf{1}, \quad \mathbf{U} = \mathbf{Z} - \text{diag}(\mathbf{Z}). \quad (7)$$

It can be easily proved that adding the penalty terms to Eq. (6) does not change its optimal solution, then the solutions to Eqs. (4) and (6) coincide with that of Eq. (7).<sup>33</sup> Next, we introduce a vector  $\bar{\boldsymbol{\delta}} \in \mathbb{R}^{MN}$  and a matrix  $\bar{\Delta} \in \mathbb{R}^{MN \times MN}$  as Lagrange multipliers for the two equality constraints in Eq. (7) to obtain the Lagrange function

$$\mathcal{L}(\mathbf{Z}, \mathbf{U}, \bar{\mathbf{Z}}, \bar{\boldsymbol{\delta}}, \bar{\Delta}) = \|\mathbf{Z}\|_1 + \frac{\lambda}{2} \|\mathbf{X} - \mathbf{XU}\|_F^2 + \frac{\alpha}{2} \|\bar{\mathbf{Z}} - \mathbf{U}\|_F^2 + \frac{\rho}{2} \|\mathbf{U}^T \mathbf{1} - \mathbf{1}\|_2^2$$

$$+ \frac{\rho}{2} \|\mathbf{U} - [\mathbf{Z} - \text{diag}(\mathbf{Z})]\|_F^2 + \bar{\boldsymbol{\delta}}^T (\mathbf{U}^T \mathbf{1} - \mathbf{1}) + \text{tr}\{\bar{\Delta}^T [\mathbf{U} - \mathbf{Z} + \text{diag}(\mathbf{Z})]\}, \quad (8)$$

where  $\text{tr}(\cdot)$  denotes the trace operator of a given matrix.

The optimization problem in Eq. (8) can then be divided into four subproblems:

1. **U update:**  $\mathbf{U}^{(t+1)}$  is obtained by minimizing  $\mathcal{L}$  with respect to  $\mathbf{U}$  while  $(\mathbf{Z}^{(t)}, \bar{\mathbf{Z}}^{(t)}, \bar{\boldsymbol{\delta}}^{(t)}, \bar{\Delta}^{(t)})$  are fixed. We calculate the derivative of  $\mathcal{L}$  with respect to  $\mathbf{U}$  and set it to zero to obtain the calculation formula of  $\mathbf{U}$  as follows:

$$(\lambda \mathbf{X}^T \mathbf{X} + \alpha \mathbf{I} + \rho \mathbf{1}\mathbf{1}^T + \rho \mathbf{I}) \mathbf{U}^{(t+1)} = \lambda \mathbf{X}^T \mathbf{X} + \alpha \bar{\mathbf{Z}}^{(t)} + \rho (\mathbf{1}\mathbf{1}^T + \mathbf{Z}^{(t)}) - \mathbf{1} \bar{\boldsymbol{\delta}}^{(t)T} - \bar{\Delta}^{(t)}. \quad (9)$$

2. **Z update:**  $\mathbf{Z}^{(t+1)}$  is computed by minimizing  $\mathcal{L}$  with respect to  $\mathbf{Z}$  while  $(\mathbf{U}^{(t+1)}, \bar{\mathbf{Z}}^{(t)}, \bar{\boldsymbol{\delta}}^{(t)}, \bar{\Delta}^{(t)})$  are fixed, obtaining

$$\mathbf{Z}^{(t+1)} = \mathbf{J} - \text{diag}(\mathbf{J}), \quad \mathbf{J} \triangleq \Gamma_{\frac{1}{\rho}} \left( \mathbf{U}^{(t+1)} + \frac{\bar{\Delta}^{(t)}}{\rho} \right), \quad (10)$$

where  $\Gamma_{1/\rho}(\cdot)$  is a shrinkage-thresholding operator,  $\Gamma_{1/\rho}(v) = [|u| - (1/\rho)]_+ \text{sgn}(v)$ , and the operator  $(\cdot)_+$  returns its arguments if it is nonnegative and zero otherwise.

3. **Enhanced coefficient matrix  $\bar{\mathbf{Z}}$  update:** Once  $\mathbf{Z}^{(t+1)}$  is obtained in the previous step, we then use the actual matrix  $\mathbf{Z}$  and follow the procedure described in Fig. 3 to update  $\bar{\mathbf{Z}}^{(t+1)}$ . We use the Gaussian kernel given by Eq. (3).
4. **The Lagrange multipliers update:**  $\bar{\boldsymbol{\delta}}^{(t+1)}$  and  $\bar{\Delta}^{(t+1)}$  are obtained through a gradient ascent update with step size  $\rho = 300$ , as follows:

$$\bar{\boldsymbol{\delta}}^{(t+1)} = \bar{\boldsymbol{\delta}}^{(t)} + \rho [\mathbf{Z}^{(t+1)} \mathbf{1} - \mathbf{1}] \quad \bar{\Delta}^{(t+1)} = \bar{\Delta}^{(t)} + \rho [\mathbf{U}^{(t+1)} - \mathbf{Z}^{(t+1)}]. \quad (11)$$

These four steps are repeated until convergence is achieved or the number of iterations exceeds the maximum fixed. Specifically, the iteration is terminated when we have  $\|\mathbf{U}^{(t)T} \mathbf{1} - \mathbf{1}\|_{\infty} \leq \bar{\epsilon}$ ,  $\|\mathbf{U}^{(t)} - \mathbf{Z}^{(t)}\|_{\infty} \leq \bar{\epsilon}$ ,  $\|\mathbf{U}^{(t)} - \mathbf{U}^{t-1}\|_{\infty} \leq \bar{\epsilon}$ , where  $\bar{\epsilon}$  denotes the error tolerance for the primal and dual residuals.

Once the sparse coefficient matrix  $\mathbf{Z}$  is obtained by solving the optimization problem in Eq. (4), the segmentation of the data points is attained by applying spectral clustering to the Laplacian matrix induced by the similarity matrix  $\mathbf{W} \in \mathbb{R}^{N \times N}$ , which is defined as  $\mathbf{W} = |\mathbf{Z}| + |\mathbf{Z}|^T$ .<sup>32</sup> The complete SSC with 3D regularization (3DS-SSC) is summarized in Algorithm 2.

## 4 Experimental Evaluation

This section shows the performance of 3DS-SSC for land cover segmentation. (A MatLab implementation of Algorithm 2 can be found in Ref. 34.) The ADMM algorithm for solving the sparse optimization problem in Eq. (4) was implemented in MATLAB. We used the implementation of SSC, described in Algorithm 1, provided by authors in Ref. 32. All the experiments were executed on an Intel Core i7 CPU (2.60 GHz, 6 cores), with 32 GB of RAM.

### 4.1 Experiment Setup

#### 4.1.1 Databases

The proposed subspace clustering approach (3DS-SSC) was tested on three well-known hyperspectral images (The datasets can be downloaded from Ref. 35.), with different imaging environments: Indian Pines, Salinas, and University of Pavia dataset. The Indian Pines hyperspectral data set has  $145 \times 145$  pixels and 200 spectral bands in the range 0.4 to 2.5  $\mu\text{m}$ . The second scene, Salinas, has  $512 \times 217$  pixels and 204 spectral bands in the range 0.24 to 2.40  $\mu\text{m}$ . The third scene, University of Pavia, comprises  $610 \times 340$  pixels, and has 103 spectral bands with spectral coverage ranging from 0.43 to 0.84  $\mu\text{m}$ . In the first experiments we selected, for each image, one frequently used region of interest (ROI) in hyperspectral image clustering, as shown in Fig. 4. The Indian Pines ROI has a size of  $70 \times 70$  pixels, which includes four main land-cover classes: corn-no-till, grass, soybeans-no-till, and soybeans-min-till. The Salinas ROI comprises

**Algorithm 2** Sparse subspace clustering with 3D regularization (3DS-SSC).

---

**Input:** HSI data  $\mathbf{X} \in \mathbb{R}^{D \times N}$ , containing a set of spectral pixels  $\{\mathbf{x}_j\}_{j=1}^N$ , in a union of  $k$  affine subspaces  $\{\mathcal{S}_i\}_{i=1}^k$ ; the number of subspaces  $k$  and the regularization parameter  $\lambda$ ; the window size  $h$  and standard deviation  $\sigma$  of the 3D Gaussian kernel; ADMM maximum number of iteration  $MAXITER$

**Output:** Matrix  $\hat{\mathbf{X}}$  with the obtained clustering labels of  $\mathbf{X}$

**Function** 3DS – SSC( $\mathbf{X}, k, \lambda$ )

```

1  ▷ Solving the optimization problem in Eq. (4)
2   $\mathbf{U}^{(0)} = \mathbf{0}, \mathbf{Z}^{(0)} = \mathbf{0}, \bar{\mathbf{Z}}^{(0)} = \mathbf{0}, \bar{\delta}^{(0)} = \mathbf{0}, \bar{\Delta}^{(0)} = \mathbf{0}, \rho = 300$ 
3  for  $t = 0, \dots, MAXITER - 1$  do
4     $\mathbf{U}^{(t+1)} \leftarrow \arg \min_{\mathbf{U}} \mathcal{L}(\mathbf{Z}^{(t)}, \mathbf{U}, \bar{\mathbf{Z}}^{(t)}, \bar{\delta}^{(t)}, \bar{\Delta}^{(t)})$ 
5     $\mathbf{Z}^{(t+1)} \leftarrow \arg \min_{\mathbf{Z}} \mathcal{L}(\mathbf{Z}, \mathbf{U}^{(t+1)}, \bar{\mathbf{Z}}^{(t)}, \bar{\delta}^{(t)}, \bar{\Delta}^{(t)})$ 
6     $\bar{\mathbf{Z}}^{(t+1)} \leftarrow \text{conv3D}(\mathbf{Z}^{(t+1)}, h, \sigma)$ 
7     $\bar{\delta}^{(t+1)} \leftarrow \bar{\delta}^{(t)} + \rho(\mathbf{Z}^{(t+1)} \mathbf{1} - \mathbf{1})$ 
8     $\bar{\Delta}^{(t+1)} \leftarrow \bar{\Delta}^{(t)} + \rho(\mathbf{U}^{(t+1)} - \mathbf{Z}^{(t+1)})$ 
9  ▷ Obtaining the segmentation using spectral clustering
10 Normalize the columns of  $\mathbf{Z}$  as  $\mathbf{z}_j \leftarrow \frac{\mathbf{z}_j}{\|\mathbf{z}_j\|_{\infty}}$ .
11 Form a similarity matrix as  $\mathbf{W} = |\mathbf{Z}| + |\mathbf{Z}|^T$ , representing the relationship (weights) among data points.
12 Apply spectral clustering19 to the similarity matrix.
13 return segmentation of the data:  $\hat{\mathbf{X}}_1, \dots, \hat{\mathbf{X}}_k$ .
```

**Function** conv3D( $\mathbf{Z}, h, \sigma$ )

```

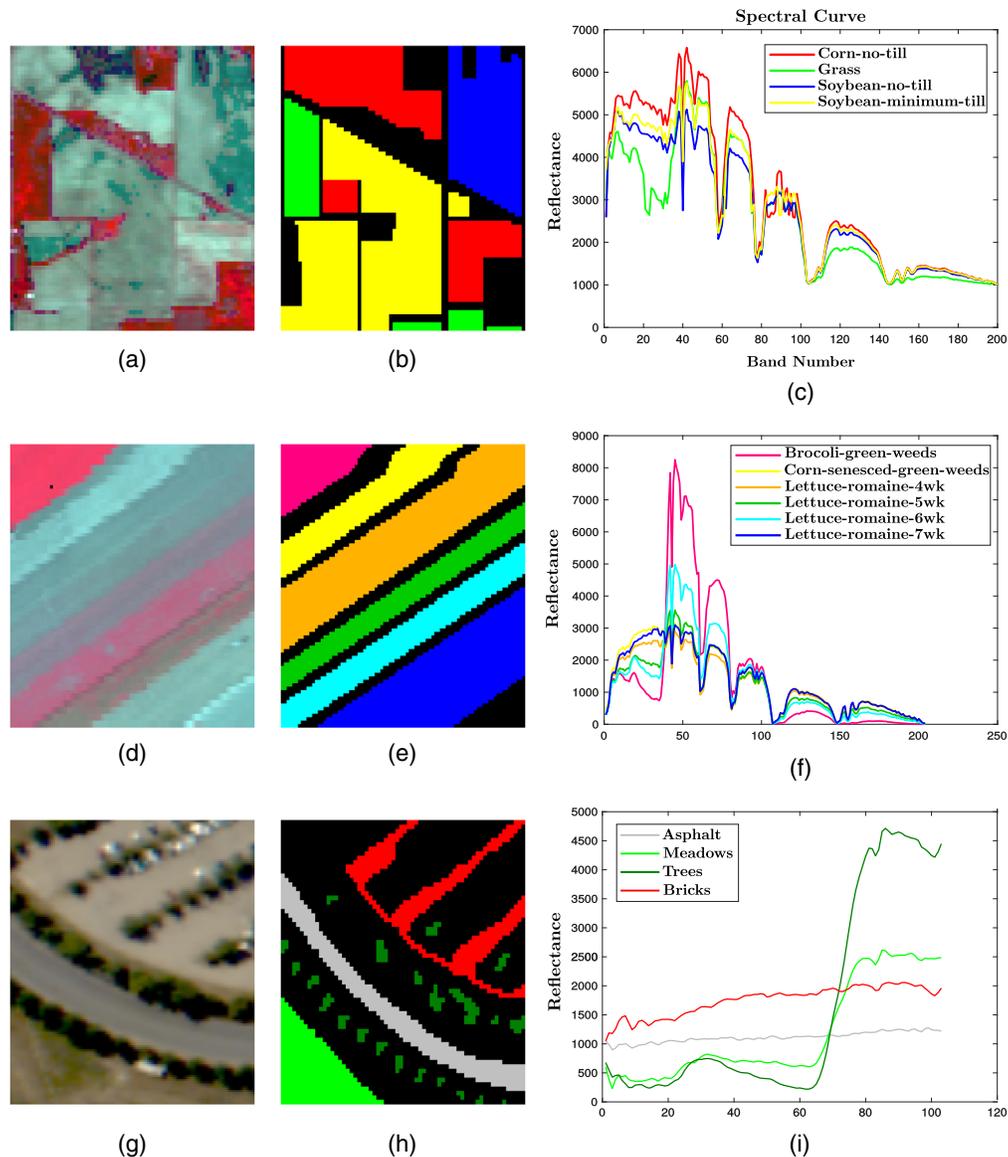
14 ▷ Perform 3D convolution (see Fig. 3)
15  $\hat{\mathbf{Z}} \leftarrow \text{reshape}(\mathbf{Z})$  ▷ Reshapes each column of  $\mathbf{Z}$  into a matrix using columnwise order
16  $\mathbf{G}_{i,j,k} = -\frac{1}{\sqrt{2\pi\sigma}} e^{-\frac{i^2 + j^2 + k^2}{2\sigma^2}}$  ▷ 3D Gaussian kernel  $\mathbf{G} \in \mathbb{R}^{h \times h \times h}$ 
17  $\hat{\mathbf{Z}} \leftarrow \hat{\mathbf{Z}} * \mathbf{G}$  ▷ 3D convolution
18  $\bar{\mathbf{Z}} \leftarrow \text{reshape}(\hat{\mathbf{Z}})$  ▷ Reshapes the cube back to a matrix
19 return  $\bar{\mathbf{Z}}$ 
```

---

83 × 83 pixels and includes six classes: broccoli-1, corn-senesced, lettuce-4wk, lettuce-5wk, lettuce-6wk, and lettuce-7wk. The University of Pavia ROI is composed of 64 × 64 pixels and includes four classes: asphalt, meadows, trees, and bricks. Figure 4 presents a false-color image, the ground truth image, and the spectral signatures of each land cover class within (a)–(c) the Indian Pines, (d)–(f) Salinas, and (g)–(i) University of Pavia selected ROIs, respectively. In addition, we also evaluate the performance of the proposed method using the full hyperspectral images in Sec. 4.2.4.

#### 4.1.2 Baselines and evaluation metrics

We compare our approach with three non-SSC-based algorithms for hyperspectral land cover segmentation: the FCM\_S1,<sup>13</sup> the RMMF,<sup>14</sup> and the 3D-CAE,<sup>15</sup> which use 3D convolutional autoencoders to perform the clustering; with two SSC-based algorithms for hyperspectral land-cover segmentation: the TV-KSSC-SMP<sup>36</sup> and S-SSC;<sup>25</sup> and with two state-of-the-art general-purpose subspace clustering algorithms: ESC-FFS<sup>22</sup> and SR-SSC.<sup>20</sup> Besides, we show the results obtained with SSC<sup>32</sup> as a reference. For the sake of fairness, we performed 10 experiments



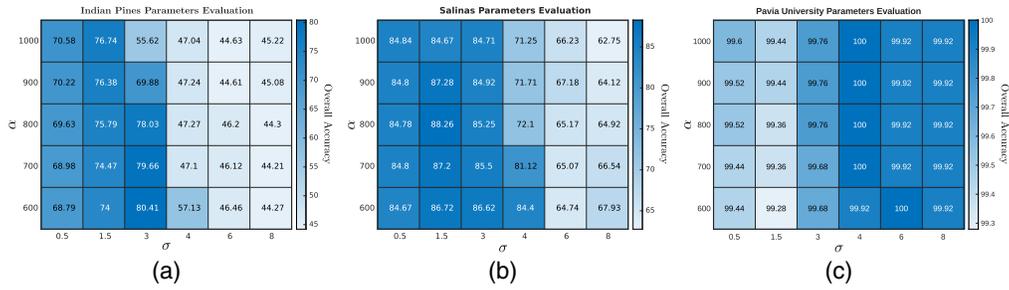
**Fig. 4** False-color images, ground truth images and spectral curves of each land-cover classes for (a)–(c) Indian Pines, (d)–(f) Salinas, and (g)–(i) University of Pavia ROIs, respectively.

for the ESC-FFS method, which relies on random initialization. Therefore, the presented results in the tables are the average of all run trials, and we selected the best results to show the land cover maps.

To compare the clustering performance of our model, we rely on six standard metrics: producer’s accuracy (PA), user’s accuracy (UA), average accuracy (AA), overall accuracy (OA), Kappa coefficient, and normalized mutual information (NMI).<sup>37,38</sup> PA and UA represent the clustering accuracy of each class. UA, AA, and OA values are presented in percentage, whereas Kappa coefficients and NMI values range from 0 to 1, where 1 stands for perfect classification.

#### 4.1.3 Parameters setting

In the experiments, the number of clusters was manually set for all the subspace clustering algorithms. The regularization parameter  $\lambda$ , in Eq. (4), which acts as the trade-off between the sparsity of the coefficient matrix and the magnitude of the noise, was set using the following formulation:<sup>32</sup>



**Fig. 5** Analysis of the parameters  $\sigma$  and  $\alpha$  in terms of OA for (a) Indian Pines, (b) Salinas, and (c) University of Pavia datasets.

$$\lambda = \frac{\beta}{\gamma}, \quad \gamma = \min_j \max_{j \neq j'} |\mathbf{x}_j^T \mathbf{x}_{j'}|, \quad (12)$$

where  $\beta$  is the adjustment coefficient,  $\gamma$  is a parameter related to the data set, which can be explicitly determined, and  $\mathbf{x}_j, \mathbf{x}_{j'}$  are columns of  $\mathbf{X}$ . The regularization parameter  $\alpha$  in Eq. (5) denotes the weight of the spatial information in 3DS-SSC. Furthermore, the window size  $h$  of the Gaussian kernel and the standard deviation  $\sigma$  in Eq. (3) also contribute in the incorporation of spatial information to the subspace clustering model. In this work, the window size  $h$  is defined as a function of the standard deviation  $\sigma$  as follows:

$$h = 2\lceil 2\sigma \rceil + 1. \quad (13)$$

We conducted different experiments varying  $\alpha$  and  $\sigma$  parameters to analyze the performance of the proposed method with each hyperspectral image. The change in the OA of the 3DS-SSC algorithm, corresponding to  $\alpha$  values ranging from 600 to 1000 and  $\sigma$  values ranging from 0.5 to 8, is shown in Fig. 5. We observe that the clustering precision can significantly change when using different values of  $\alpha$  and  $\sigma$ , which suggests that spatial information plays a very important role in the clustering process. In order to make a fair comparison, we also performed simulations with the other methods to select the best parameters in their configurations.

## 4.2 Visual and Quantitative Results

In this section, we present the visual and quantitative results obtained for the three hyperspectral images described in Sec. 4.1. The clustering was a challenging task because the spectral signatures of the land-cover classes in the selected ROIs are very similar and some of the spectral curves are mixed, as shown in Figs. 4(c), 4(f), and 4(i). Table 1 summarizes the selected parameters for each SSC-based clustering method and hyperspectral image. For the other methods (FCM\_S1, RMMF, and 3D-CAE), we follow the parameter settings reported by the authors in the corresponding literature. Note that, in the next sections, the quantitative evaluations shown in Tables 2–4 present the best results in bold font and the second-best are italicized.

### 4.2.1 Indian Pines ROI

The clustering performance of each method with the Indian Pines ROI is visually and quantitative reported in Fig. 6 and Table 2, respectively. From the results, it is evident that the two methods that perform better in this HSI are the TV-KSSC-SMP and our proposed method 3DS-SSC. Specifically, the 3DS-SSC achieved an OA of 80.41% and the TV-KSSC-SMP obtained an OA of 76.72%. As observed in the quantitative evaluations, the most difficult classes to correctly classify were the soybeans-no-till and soybeans-min-till. However, our method achieves the best PA with the soybeans-no-till and the best UA with soybeans-min-till. From the visual results, it is possible to observe that the TV-KSSC-SMP assigns more pixels to the wrong classes, hence it is more penalized by the used accuracy metrics. On the other hand, it is observed that the proposed

**Table 1** Selected parameters for each testing hyperspectral images.

Parameters			
SC-based method	Indian Pines	Salinas	University of Pavia
SSC	$\lambda = 7.76 \times 10^{-7}$	$\lambda = 1.17 \times 10^{-6}$	$\lambda = 3.90 \times 10^{-6}$
S-SSC	$\lambda = 7.76 \times 10^{-7}$ , $\alpha = 2.1 \times 10^5$	$\lambda = 1.17 \times 10^{-6}$ , $\alpha = 1.12 \times 10^6$	$\lambda = 3.90 \times 10^{-6}$ , $\alpha = 8.8 \times 10^5$
TV-KSSC-SMP	$\lambda = 5.4$ , $nv = 3$ , $nf = 150$	$\lambda = 7.4$ , $nv = 6$ , $nf = 50$	$\lambda = 3.6$ , $nv = 8$ , $nf = 15$
ESC-FFS	$\lambda = 10$ , $k = 700$ , $t = 10$	$\lambda = 20$ , $k = 700$ , $t = 10$	$\lambda = 10$ , $k = 1000$ , $t = 20$
SR-SSC	$\tau = 200$ , $nGraph = 5$ , $Ns = 10$	$\tau = 100$ , $nGraph = 8$ , $Ns = 10$	$\tau = 700$ , $nGraph = 15$ , $Ns = 10$
3DS-SSC	$\lambda = 7.76 \times 10^{-7}$ , $\alpha = 1.18 \times 10^5$ , $\sigma = 3$	$\lambda = 1.17 \times 10^{-6}$ , $\alpha = 8.4 \times 10^5$ , $\sigma = 1.5$	$\lambda = 3.90 \times 10^{-6}$ , $\alpha = 6.6 \times 10^5$ , $\sigma = 6$

**Table 2** Clustering performance of the different algorithms for the Indian Pines ROI.

Metric		FCM_S1	RMMF	SSC	S-SSC	TV-KSSC-	ESC-FFS	SR-SSC	3DS-SSC
						SMP			
PA	Corn-no-till	<b>63.28</b>	35.02	48.96	63.18	45.37	51.08	62.09	59.90
	Grass	93.29	<b>100.00</b>	98.60	<b>100.00</b>	<b>100.00</b>	99.92	94.69	<b>100.00</b>
	Soybean-no-till	46.99	45.22	70.63	62.30	68.44	49.00	63.11	<b>77.05</b>
	Soybeans-min-till	77.20	94.17	59.23	75.78	<b>96.54</b>	84.12	75.44	91.32
UA	Corn-no-till	80.30	<b>100.00</b>	97.43	83.99	84.29	93.98	68.87	73.24
	Grass	80.87	<b>97.02</b>	89.14	87.32	67.80	86.98	91.13	83.64
	Soybean-no-till	53.42	64.39	41.36	53.84	<b>97.85</b>	55.15	58.19	73.73
	Soybeans-min-till	66.16	59.47	61.57	71.79	71.58	63.26	74.28	<b>86.62</b>
AA		70.19	68.60	69.35	75.31	77.59	71.03	73.83	<b>82.07</b>
OA		68.70	68.06	62.62	71.90	76.72	69.20	71.08	<b>80.41</b>
Kappa		0.55	0.52	0.48	0.60	0.66	0.55	0.59	<b>0.72</b>
NMI		0.37	0.45	0.39	0.42	0.53	0.42	0.38	<b>0.57</b>

method can take advantage of the spatial contextual information better, thus providing a smoother result in comparison with the SSC and S-SSC methods.

#### 4.2.2 Salinas ROI

Figure 7 shows the obtained land cover maps on the Salinas ROI. The quantitative evaluations corresponding to the PA, UA, AA, OA, Kappa, and NMI with all the clustering methods are reported in Table 3. As observed, the RMMF algorithm achieves the best performance with an OA of 98.20%. The second-best result was achieved by our proposed method 3DS-SSC with an OA of 88.26%. In general, the most difficult land-cover class to classify was the corn-senesced according to the PA and the Lettuce-4wk according to the UA. Although our method did not obtain the best performance on these land-cover classes, the obtained results were not the worst

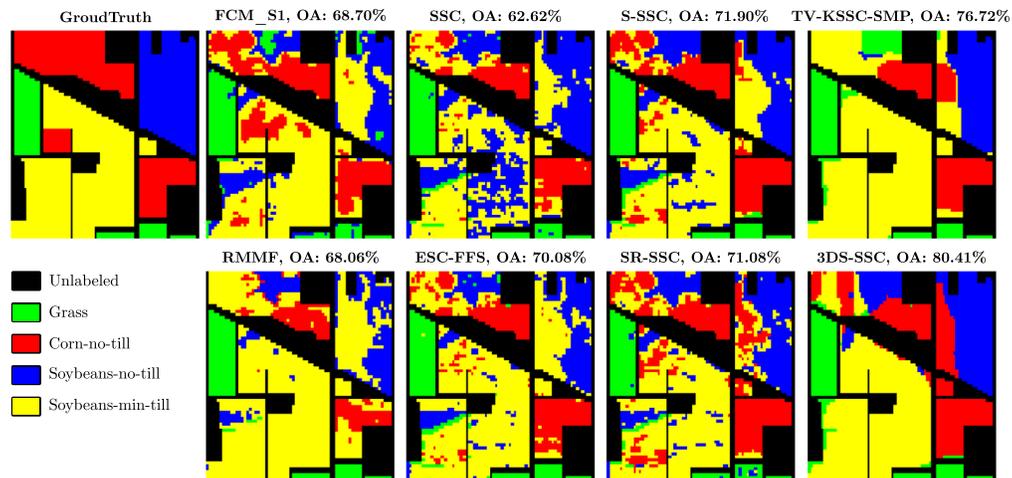
**Table 3** Clustering performance of the different algorithms for the Salinas ROI.

Metric		FCM_S1	RMMF	SSC	S-SSC	TV-KSSC-SMP	ESC-FFS	SR-SSC	3DS-SSC
PA	Brocoli-1	89.00	98.72	99.49	99.74	<b>100.00</b>	99.51	99.74	<b>100.00</b>
	Corn-senesced	47.15	<b>97.31</b>	55.62	42.01	62.82	55.09	71.28	58.31
	Lettuce-4wk	91.23	95.45	93.51	95.29	<b>100.00</b>	90.99	0.00	<b>100.00</b>
	Lettuce-5wk	88.36	99.53	99.53	91.46	36.79	<b>99.80</b>	56.29	99.06
	Lettuce-6wk	94.62	98.92	0.00	99.85	<b>100.00</b>	29.48	99.85	99.08
	Lettuce-7wk	78.47	98.44	<b>99.87</b>	97.41	90.27	98.20	98.31	92.09
UA	Brocoli-1	35.44	97.23	<b>100.00</b>	<b>100.00</b>	<b>100.00</b>	99.62	<b>100.00</b>	87.28
	Corn-senesced	85.63	<b>100.00</b>	<b>100.00</b>	96.37	<b>100.00</b>	95.90	<b>100.00</b>	99.46
	Lettuce-4wk	<b>99.65</b>	96.39	95.21	44.57	62.22	94.97	0.00	53.24
	Lettuce-5wk	95.91	97.69	68.23	97.63	<b>100.00</b>	76.39	55.10	<b>100.00</b>
	Lettuce-6wk	74.73	97.87	0.00	83.63	50.35	16.17	49.77	<b>100.00</b>
	Lettuce-7wk	81.32	98.57	99.23	<b>99.87</b>	59.69	98.64	<b>99.87</b>	99.16
AA		81.47	<b>98.06</b>	74.67	87.63	81.65	78.85	70.91	91.42
OA		78.01	<b>98.20</b>	75.65	82.41	71.33	78.76	68.26	88.26
Kappa		0.73	<b>0.98</b>	0.69	0.79	0.66	0.73	0.61	0.86
NMI		0.68	<b>0.94</b>	0.83	0.83	0.70	0.80	0.75	0.87

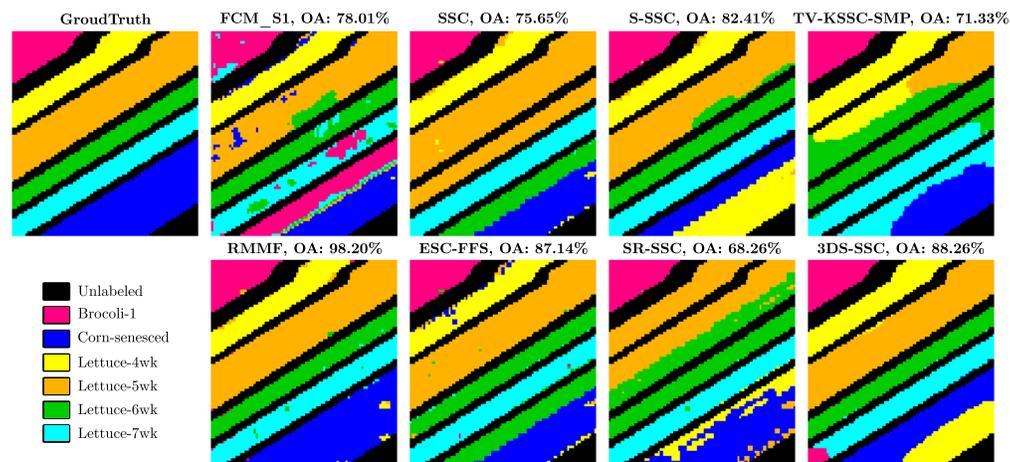
**Table 4** Clustering performance of the different algorithms for the University of Pavia ROI.

Metric		FCM_S1	RMMF	SSC	S-SSC	TV-KSSC-SMP	ESC-FFS	SR-SSC	3DS-SSC
PA	Asphalt	<b>100.00</b>	<b>100.00</b>	<b>100.00</b>	<b>100.00</b>	50.26	99.02	72.49	<b>100.00</b>
	Meadows	80.37	87.85	97.82	96.26	99.38	67.73	93.46	<b>99.82</b>
	Trees	77.71	79.62	77.71	98.09	68.15	99.11	98.73	<b>99.25</b>
	Bricks	97.46	<b>100.00</b>	99.75	<b>100.00</b>	<b>100.00</b>	98.98	99.49	<b>100.00</b>
UA	Asphalt	91.53	<b>100.00</b>	<b>100.00</b>	<b>100.00</b>	98.96	<b>100.00</b>	<b>100.00</b>	<b>100.00</b>
	Meadows	91.81	91.56	89.71	99.36	<b>99.38</b>	96.69	98.68	99.12
	Trees	73.49	<b>100.00</b>	94.57	92.77	36.27	62.72	88.07	99.93
	Bricks	98.46	89.73	<b>100.00</b>	99.75	89.12	<b>100.00</b>	78.99	<b>100.00</b>
AA		88.88	91.87	93.82	98.59	79.45	91.21	91.04	<b>99.76</b>
OA		91.35	94.32	96.56	98.80	80.78	90.98	89.67	<b>99.52</b>
Kappa		0.88	0.92	0.95	0.97	0.74	0.88	0.86	<b>0.99</b>
NMI		0.78	0.86	0.92	0.96	0.77	0.87	0.80	<b>0.99</b>

reported in Table 3. In addition note that, although TV-KSSC-SMP achieves the best performance in three different land-cover classes, its performance on the other classes is worst in comparison with our method. Conversely, although the RMMF algorithm only obtains the best performance on one land cover class (corn-senesced), it achieves high and similar performance



**Fig. 6** Land cover maps on Indian Pines ROI. The proposed method (3DS-SSC) is compared with the methods FCM\_S1, SSC, S-SSC, TV-KSSC-SMP, RMMF, ESC-FFS, and SR-SSC.

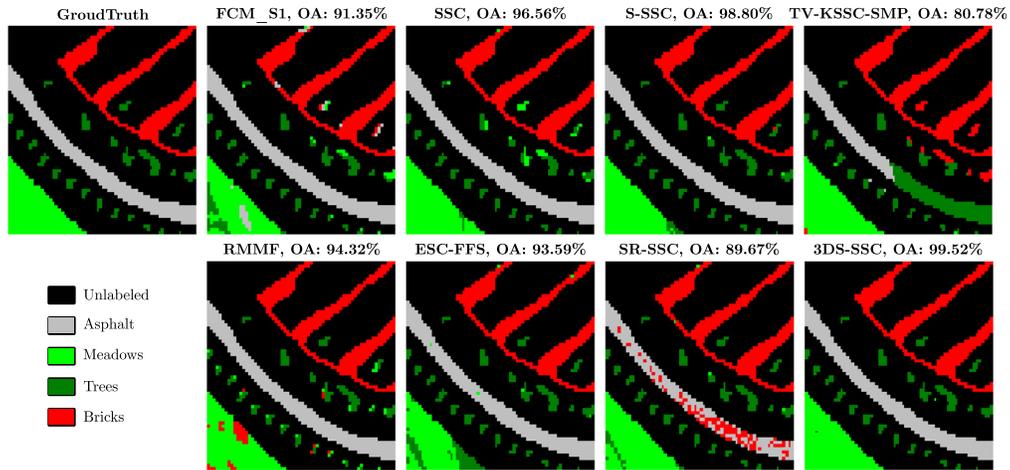


**Fig. 7** Land cover maps on Salinas ROI. The proposed method (3DS-SSC) is compared with the methods FCM\_S1, RMMF, SSC, S-SSC, TV-KSSC-SMP, ESC-FFS, and SR-SSC.

on the other classes hence obtains the best OA. From the visual results, it is possible to observe that the SR-SSC assigns more pixels to the wrong classes, hence it is more penalized by the used accuracy metrics. On the other hand, it is observed that the visual results provided by our method are smooth, which suggests that 3DS-SSC takes advantage of the spatial contextual information in comparison with the other methods

#### 4.2.3 University of Pavia ROI

Finally, Fig. 8 presents the obtained land cover maps on the University of Pavia ROI, and the quantitative evaluations corresponding to the PA, UA, AA, OA, Kappa, and NMI with all the clustering methods are reported in Table 4. As observed, our proposed method (3DS-SSC) obtains an OA of 99.52%, which corresponds to the highest classification score of the land-cover classes within the hyperspectral scene. The second-best result was achieved by the S-SSC method with an OA of 98.80%. It is important to notice that this image was the easiest to cluster since its spectral signatures are not critically mixed, as in the other hyperspectral scenes, see Fig. 4(i). From the results, it is observed that the TV-KSSC-SMP achieves the worst performance in this particular image. Although it also provides smooth results, TV-KSSC-SMP assigns more pixels from the class tree to the other classes, hence it is penalized by the used accuracy metrics.

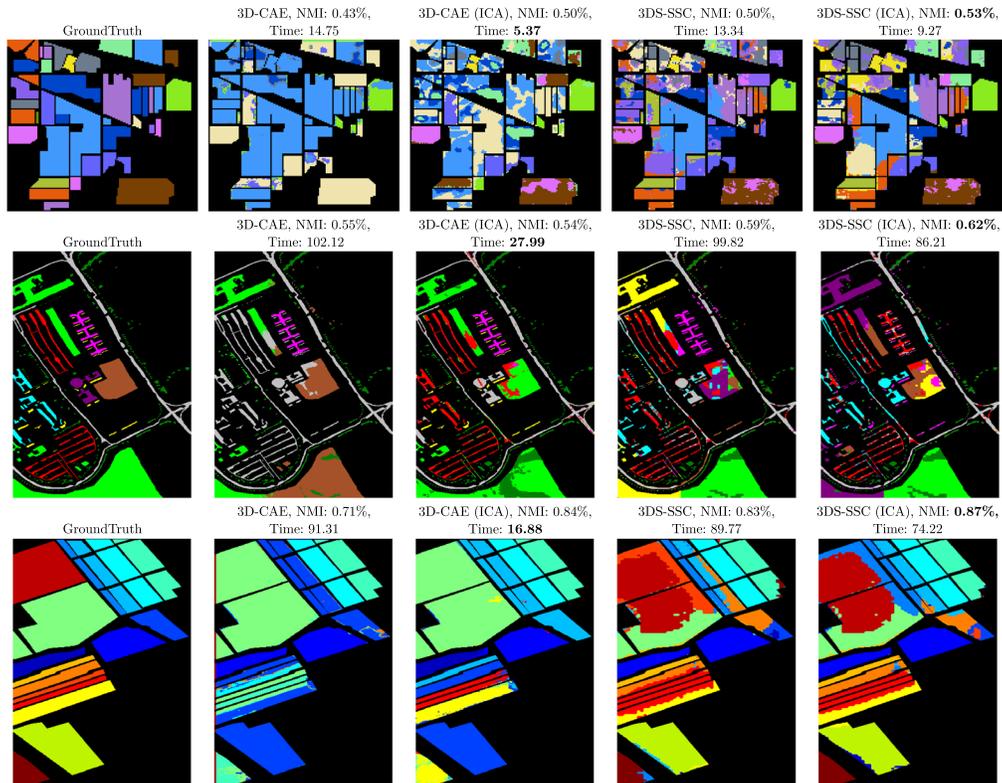


**Fig. 8** Land cover maps on University of Pavia ROI. The proposed method (3DS-SSC) is compared with the methods FCM\_S1, RMMF, SSC, S-SSC, TV-KSSC-SMP, ESC-FFS, and SR-SSC.

On the other hand, it can be observed that the proposed method better takes advantage of the spatial contextual information, providing a smoother result in comparison with RMMF, SSC, and S-SSC, which assign some pixels to the wrong classes.

#### 4.2.4 Full hyperspectral images comparison

For the sake of completeness, we compare the proposed method with a recent deep learning-based approach (3D-CAE)<sup>15</sup> that relies on 3D convolutional autoencoders to perform the



**Fig. 9** Land cover maps on the Indian Pines, University of Pavia, and Salinas full images. The proposed method (3DS-SSC) is compared with the 3D-CAE. The best results for NMI and time are shown in bold font.

clustering. We run our proposed method on the full hyperspectral images described in Sec. 4.1 and present the results in Fig. 9. The presented land cover maps of the 3D-CAE method, as well as the given execution time (in minutes) and NMI score, were obtained from its original manuscript. The land-cover maps are provided by the authors in the project repository available at: <https://gitlab.com/jnalepa/3d-cae>. For comparison purposes, we retrieve the original land-cover maps but change the colormap to compare it with our results. In the figure, we provide results for the full image after applying dimensionality reduction via independent component analysis (ICA) and without any preprocessing. Besides, the shown execution time only reflects the segmentation time and does not consider the preprocessing time. As observed, the proposed method (3DS-SSC) achieves the best performance over the test images. Moreover, the execution time of the proposed method 3DS-SSC is similar to 3D-CAE only when no applying dimensionality reduction. However, if ICA is applied to the HSI, the 3D-CAE obtains the clustering results faster in comparison with 3DS-SSC. Such behavior is expected since the theoretical complexity analysis of the SSC-based method is  $O(DM^3N^3)$ ; hence the computational burden is dominated by the number of pixels ( $MN$ ), which remains equal after ICA preprocessing.<sup>32</sup>

## 5 Conclusion

In this paper, we proposed a subspace clustering method for hyperspectral image land-cover segmentation. Precisely, our approach consists of using a 3D Gaussian filter to enhance the sparse coefficients matrix, and then we obtain the segmentation by spectral clustering. The proposed method provides a significant improvement when compared with other subspace clustering methods used for hyperspectral image land cover segmentation. Specifically, an OA of 80.41%, 88.26%, and 99.52% was achieved when performing the proposed 3DS-SSC method on a selected ROI of the Indian Pines, Salinas, and University of Pavia hyperspectral images.

## Acknowledgments

The authors gratefully acknowledge the Vicerrectoría de Investigación y Extensión of Universidad Industrial de Santander for supporting this work with the Projects “Algoritmo computacional de extracción de características basado en el modelo de mezcla lineal y en el análisis de variabilidad espectral para la identificación de palma de aceite en imágenes hiperespectrales” (VIE code 2519), and “Sistema óptico-computacional compresivo para la adquisición y reconstrucción de información espacial, espectral y de profundidad de una escena, mediante modulaciones del frente de onda utilizando espejos deformables y aperturas codificadas” (VIE code 2521).

## References

1. N. A. Hagen and M. W. Kudenov, “Review of snapshot spectral imaging technologies,” *Opt. Eng.* **52**(9), 090901 (2013).
2. C. V. Correa et al., “Multiple snapshot colored compressive spectral imager,” *Opt. Eng.* **56**(4), 041309 (2016).
3. N. Diaz, C. Hinojosa, and H. Arguello, “Adaptive grayscale compressive spectral imaging using optimal blue noise coding patterns,” *Opt. Laser Technol.* **117**, 147–157 (2019).
4. D. Caballero, R. Salvini, and J. M. Amigo, “Chapter 3.3–Hyperspectral imaging in crop fields: precision agriculture,” in *Data Handling in Science and Technology*, J. M. Amigo, Ed., Vol. 32, pp. 453–473, Elsevier, Amsterdam, Netherlands (2020).
5. K. Sanchez, C. Hinojosa, and H. Arguello, “Supervised spatio-spectral classification of fused images using superpixels,” *Appl. Opt.* **58**(7), B9–B18 (2019).
6. C. Hinojosa, J. M. Ramirez, and H. Arguello, “Spectral-spatial classification from multi-sensor compressive measurements using superpixels,” in *IEEE Int. Conf. Image Process.*, IEEE, pp. 3143–3147 (2019).

7. C. Hinojosa et al., "Single-pixel camera sensing matrix design for hierarchical compressed spectral clustering," in *IEEE 29th Int. Workshop Mach. Learn. Signal Process.*, IEEE, pp. 1–6 (2019).
8. B. Gessesse, W. Bewket, and A. Bräuning, "Model-based characterization and monitoring of runoff and soil erosion in response to land use/land cover changes in the Modjo Watershed, Ethiopia," *Land Degrad. Dev.* **26**(7), 711–724 (2015).
9. M. Volpi and V. Ferrari, "Semantic segmentation of urban scenes by learning local class interactions," in *Proc. IEEE Conf. Comput. Vision and Pattern Recognit. Workshops*, pp. 1–9 (2015).
10. X. Briottet et al., "Military applications of hyperspectral imagery," *Proc. SPIE* **6239**, 62390B (2006).
11. G. Melillos et al., "Integrated use of field spectroscopy and satellite remote sensing for defence and security applications in cyprus," *Proc. SPIE* **9688**, 96880F (2016).
12. J. C. Bezdek, "Pattern recognition with fuzzy objective function algorithms," Springer, Berlin, Germany (1981).
13. S. Chen and D. Zhang, "Robust image segmentation using FCM with spatial constraints based on new kernel-induced distance measure," *IEEE Trans. Syst. Man Cybern. Part B* **34**(4), 1907–1916 (2004).
14. L. Zhang et al., "Hyperspectral image unsupervised classification by robust manifold matrix factorization," *Inf. Sci.* **485**, 154–169 (2019).
15. J. Nalepa et al., "Unsupervised segmentation of hyperspectral images using 3-D convolutional autoencoders," *IEEE Geosci. Remote Sens. Lett.* **17**, 1948–1952 (2020).
16. C. Lu et al., "Subspace clustering by block diagonal representation," *IEEE Trans. Pattern Anal. Mach. Intell.* **41**(2), 487–501 (2019).
17. J. Zhang et al., "Self-supervised convolutional subspace clustering network," in *Proc. IEEE Conf. Comput. Vision and Pattern Recognit.*, pp. 5473–5482 (2019).
18. E. Elhamifar and R. Vidal, "Sparse subspace clustering," in *IEEE Conf. Comput. Vision and Pattern Recognit.*, IEEE, pp. 2790–2797 (2009).
19. U. Von Luxburg, "A tutorial on spectral clustering," *Stat. Comput.* **17**, 395–416 (2007).
20. M. Abdolali, N. Gillis, and M. Rahmati, "Scalable and robust sparse subspace clustering using randomized clustering and multilayer graphs," *Signal Process.* **163**, 166–180 (2019).
21. R. Tibshirani, "Regression shrinkage and selection via the lasso," *J. R. Stat. Soc. Ser. B* **58**(1), 267–288 (1996).
22. C. You et al., "A scalable exemplar-based subspace clustering algorithm for class-imbalanced data," in *Lect. Notes Comput. Sci.*, V. Ferrari et al., Eds., Vol. 11213, pp. 68–85, Springer, Cham, Germany (2018).
23. D. P. Williamson and D. B. Shmoys, "The design of approximation algorithms," Cambridge University Press, Cambridge (2011).
24. E. Elhamifar and R. Vidal, "Sparse manifold clustering and embedding," in *Adv. Neural Inf. Process. Syst.*, pp. 55–63 (2011).
25. H. Zhang et al., "Spectral–spatial sparse subspace clustering for hyperspectral remote sensing images," *IEEE Trans. Geosci. Remote Sens.* **54**(6), 3672–3684 (2016).
26. H. Zhai et al., "A new sparse subspace clustering algorithm for hyperspectral remote sensing imagery," *IEEE Geosci. Remote Sens. Lett.* **14**(1), 43–47 (2017).
27. Y. Guo, J. Gao, and F. Li, "Spatial subspace clustering for drill hole spectral data," *J. Appl. Remote Sens.* **8**(1), 083644 (2014).
28. A. Mehta and O. Dikshit, "Segmentation-based clustering of hyperspectral images using local band selection," *J. Appl. Remote Sens.* **11**(1), 015028 (2017).
29. C. Hinojosa, J. Bacca, and H. Arguello, "Coded aperture design for compressive spectral subspace clustering," *IEEE J. Sel. Top. Signal Process.* **12**(6), 1589–1600 (2018).
30. Y. Liang, P. P. Markopoulos, and E. Saber, "Spatial–spectral segmentation of hyperspectral images for subpixel target detection," *J. Appl. Remote Sens.* **13**(3), 036502 (2019).
31. S. Huang, H. Zhang, and A. Pižurica, "Semisupervised sparse subspace clustering method with a joint sparsity constraint for hyperspectral remote sensing images," *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **12**(3), 989–999 (2019).

32. E. Elhamifar and R. Vidal, "Sparse subspace clustering: algorithm, theory, and applications," *IEEE Trans. Pattern Anal. Mach. Intell.* **35**(11), 2765–2781 (2013).
33. S. Boyd et al., "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Found. Trends Mach. Learn.* **3**(1), 1–122 (2011).
34. C. Hinojosa, "3DS-SSC code repository," <https://link.carloshinojosa.me/3DS-SSC> (accessed 17 January 2021).
35. "Grupo de Inteligencia Computacional," <http://www.ehu.es/ccwintco/index.php> (accessed 17 January 2021).
36. J. Bacca, C. A. Hinojosa, and H. Arguello, "Kernel sparse subspace clustering with total variation denoising for hyperspectral remote sensing images," in *Math. Imaging*, MTu4C–5, Optical Society of America (2017).
37. T. Lillesand, R. W. Kiefer, and J. Chipman, "Remote Sensing and Image Interpretation," John Wiley & Sons (2015).
38. A. Strehl and J. Ghosh, "Cluster ensembles—a knowledge reuse framework for combining multiple partitions," *J. Mach. Learn. Res.* **3**, 583–617 (2003).

**Carlos Hinojosa** received his BSc Eng. and MSc degrees in systems engineering and informatics from Universidad Industrial de Santander, Colombia, in 2015 and 2018, respectively. Currently, he is pursuing his PhD in computer science at Universidad Industrial de Santander. His research interests include computational imaging, high-dimensional image processing, unsupervised learning, optical code design, compressive imaging, computer vision, and sparse representation.

**Fernando Rojas** received his BSc Eng. degree in systems engineering and informatics from the Universidad Industrial de Santander, Colombia, and his MSc degree in computational science from Instituto Tecnológico de Monterrey. Currently, he is an associate professor with the Department of Systems Engineering, Universidad Industrial de Santander. His main research areas are image processing and software engineering.

**Sergio Castillo** received his BSc Eng. degree in systems engineering and informatics from the Universidad Industrial de Santander, Colombia, MSc degree in informatics from Université Laval, MSc degree in complex systems from Universidad De Aix-Marseille, and his PhD in Telematic Systems Engineering from Universidad Politecnica de Madrid. Currently, he is an associate professor with the Department of Systems Engineering, Universidad Industrial de Santander. His main research interests include image processing and software development.

**Henry Arguello** received his BSc Eng. degree in electrical engineering, MSc degree in electrical power from the Universidad Industrial de Santander, Bucaramanga, Colombia, in 2000 and 2003, respectively, and his PhD in electrical engineering from the University of Delaware, USA, in 2013. Currently, he is an associate professor with the Department of Systems Engineering, Universidad Industrial de Santander. His research interests include high-dimensional signal processing, optical imaging, compressed sensing, and computational imaging.