

Article

# Classification of Hyperspectral Images Using Kernel Fully Constrained Least Squares <sup>†</sup>

Jianjun Liu <sup>1,\*</sup> , Zebin Wu <sup>2</sup> , Zhiyong Xiao <sup>1</sup> and Jinlong Yang <sup>1</sup>

<sup>1</sup> The Key Laboratory of Advanced Process Control for Light Industry (Ministry of Education), School of Internet of Things Engineering, Jiangnan University, Wuxi 214122, China; zhiyong.xiao@jiangnan.edu.cn (Z.X.); yjlgedeng@163.com (J.Y.)

<sup>2</sup> School of Computer Science, Nanjing University of Science and Technology, Nanjing 210094, China; zebin.wu@gmail.com

\* Correspondence: liuofficial@163.com

<sup>†</sup> This paper is an extended version of our paper published in 2017 IEEE International Geoscience and Remote Sensing Symposium.

Received: 19 September 2017; Accepted: 3 November 2017; Published: 6 November 2017

**Abstract:** As a widely used classifier, sparse representation classification (SRC) has shown its good performance for hyperspectral image classification. Recent works have highlighted that it is the collaborative representation mechanism under SRC that makes SRC a highly effective technique for classification purposes. If the dimensionality and the discrimination capacity of a test pixel is high, other norms (e.g.,  $\ell_2$ -norm) can be used to regularize the coding coefficients, except for the sparsity  $\ell_1$ -norm. In this paper, we show that in the kernel space the nonnegative constraint can also play the same role, and thus suggest the investigation of kernel fully constrained least squares (KFCLS) for hyperspectral image classification. Furthermore, in order to improve the classification performance of KFCLS by incorporating spatial-spectral information, we investigate two kinds of spatial-spectral methods using two regularization strategies: (1) the coefficient-level regularization strategy, and (2) the class-level regularization strategy. Experimental results conducted on four real hyperspectral images demonstrate the effectiveness of the proposed KFCLS, and show which way to incorporate spatial-spectral information efficiently in the regularization framework.

**Keywords:** hyperspectral; image classification; least squares; collaborative representation; sparse representation; posterior probability; regularization

## 1. Introduction

Sparse representation classification (SRC) has been widely used in many applications, such as pattern recognition [1,2], visual classification [3,4], and hyperspectral image classification [5–11]. Unlike the common classifiers (e.g., support vector machines (SVMs) [12] and multinomial logistic regression [13]), SRC is not a learning-based classifier, which first represents a test sample as the sparse linear combination of all training samples and then directly assigns a class label to the test sample by evaluating which class leads to the minimum reconstruction error. Although the usage of sparsity prior in the literature often leads to robust classification performance, recent works [14,15] have shown that it is the collaborative representation (CR) mechanism under SRC (i.e., representing a test sample collaboratively with training samples from all classes) that makes SRC a highly effective technique for classification purposes. Moreover, if the dimensionality and the discrimination capacity of a sample is high, other regularization terms such as  $\ell_2$ -norm can play the same role as the sparsity  $\ell_1$ -norm. Several approaches have demonstrated the effectiveness of the classification method using  $\ell_2$ -norm in many applications [14,16], including hyperspectral image classification [17,18]. For the sake of

simplicity, the method using  $\ell_2$ -norm is referred to as collaborative representation classification (CRC), and both SRC and CRC are referred to as CR-based classification methods.

Although CR-based classification methods can get good performance, it is difficult to use them to classify data that is not linearly separable. Moreover, for hyperspectral image classification, the discriminability of a pixel is generally low, owing to the presence of redundant spectral bands, although its dimensionality is high. As the pixel-wise classification results reported in [17,18], SRC often produces superior performance compared to CRC. Some approaches have considered using the kernel method that is widely used in SVM classification [12,19] to mitigate these problems [20–22], since in the kernel feature space the dimensionality of a sample is very high, and its discriminability is generally enhanced [23]. In hyperspectral image classification, kernel CR (KCR)-based classification has shown an improvement over CR-based classification [21,24,25], and kernel CRC (KCRC) exhibits competitive advantages in terms of classification accuracy and computational cost when compared with kernel SRC (KSRC) [23].

In the development of CR-based classification methods, more attention is paid to the selection of norms. However, for both SRC or CRC they belong to the regularized least squares. That is to say, the improvement brought by the norms can also be achieved by other regularization terms or constraint terms. Among the numerous terms, the nonnegative constraint is an effective one that is very common in many other techniques and applications, such as nonnegative matrix factorization [26] and spectral unmixing [27]. Moreover, the nonnegative constraint can also bring the sparsity of the coding coefficients [28,29]. Accordingly, we consider exploiting kernel nonnegative constrained least squares (KNLS) for hyperspectral image classification. Since in the kernel feature space the dimensionality and the discrimination capacity of a pixel is high, the nonnegative constraint may play a role of CR. Considering that the nonnegative coding coefficient reflects the similarity between a test pixel and the related training pixel, we suggest to provide the posterior probabilistic outputs by enforcing the summation of the nonnegative coding coefficients of each pixel to be one, and thus investigate kernel fully constrained least squares (KFCLS) [30,31] for hyperspectral image classification.

The investigated KFCLS is a pixel-wise classifier that treats hyperspectral data as an unordered list of feature vectors but not as images. In order to handle the coarse classification maps brought by a pixel-wise classifier, previous methods have considered incorporating spatial-contextual information during the classification process [32]. According to the relationship between the pixel-wise classification process and the fusion of spatial-spectral information, these methods can be roughly divided into three categories:

- (1) The first category can be treated as pre-processing methods. These methods usually extract the spatial features first, and then incorporate both the spatial and spectral features into a pixel-wise classifier. For instance, in [33] a composite kernel framework is proposed to combine the spatial and spectral features first and subsequently embed into SVM for classification purposes. In [34], the authors extract multiple types of spatial features from both linear and nonlinear transformations first, and then integrate them via multinomial logistic regression. In [35], a convolutional neural network is utilized to extract deep features from high levels of the image data and the final classification is done by using SRC.
- (2) The second category can be treated as post-processing methods. These methods usually perform a pixel-wise classifier first, and then refine the pixel-wise results by incorporating spatial information. For instance, in [23,36–38] the class conditional probability density functions are first estimated using a probabilistic pixel-wise classifier, and then refined by using some regularization models to incorporate the spatial information. In [39,40], the original hyperspectral image is first classified per pixel and simultaneously segmented into several adaptive neighborhoods, and then a decision fusion mechanism is undertaken within the pixel-wise classification results of these neighborhoods. In [41], KSRC is first used to get the coding coefficients of the original hyperspectral image, and then the coding coefficients are refined by incorporating the spatial information for the final classification.

- (3) The last category can be treated as co-processing methods, which jointly integrate the pixel-wise classification process and the fusion of spatial-spectral information. For CR-based classification, the related methods usually assign a neighborhood/window to each pixel, and perform the representation of a test pixel jointly by its neighbouring pixels [21,42–45]. In addition, there are other methods that consider incorporating the spatial information by appending a spatial-spectral term to the coding model of CR-based classification [11,25,46].

Notably, regularization is an important technique for CR-based classification, since all CR-based classification methods are built by regularization technique. As a widely used technique in mathematical and image processing problems [47], the regularization technique is very suitable for the integration of different prior knowledge owing to its flexibility and availability. This paper considers incorporating the spatial information into KFCLS using regularization technique. For this issue, we consider both the co-processing and post-processing methods, and propose a weighted H1-norm [48] for the description of spatial information. Furthermore, we investigate two regularization strategies to integrate the spatial and spectral information. One is the coefficient-level regularization strategy that incorporates the spatial information by enforcing or refining the coding coefficients, and the other is the class-level regularization strategy that handles the posterior probabilistic outputs.

The remainder of this paper is organized as follows. Section 2 briefly introduces two instantiations of KCR-based classification methods (i.e., KSRC and KCRC). In Section 3, we first present the proposed KFCLS for hyperspectral image classification, and then introduce the co-processing and post-processing methods for KFCLS using two regularization strategies. The effectiveness of the proposed KFCLS and the suggested way of incorporating spatial-spectral information are demonstrated in Section 4 by conducting experiments on four real hyperspectral images. Finally, Section 5 concludes this paper.

## 2. KCR-Based Classification

In this section, we briefly review the general model of KCR-based classification and subsequently introduce its two instantiations. Given a hyperspectral image, every pixel in it can be interpreted as an  $L$ -dimensional column vector with  $L$  being the number of spectral bands. Suppose the given hyperspectral image includes  $C$  classes, and there exists a feature mapping function  $\phi$  which maps a test pixel  $\mathbf{x} \in \mathbb{R}^L$  and  $J$  training pixels  $\mathbf{A} = [\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_J] \in \mathbb{R}^{L \times J}$  to the high-dimensional feature space:  $\mathbf{x} \rightarrow \phi(\mathbf{x})$ ,  $\mathbf{A} \rightarrow \Phi(\mathbf{A}) = [\phi(\mathbf{a}_1), \phi(\mathbf{a}_2), \dots, \phi(\mathbf{a}_J)]$ . For a mapped pixel  $\phi(\mathbf{x})$ , KCR-based classification supposes that it can be collaboratively represented as the linear combination of all mapped training pixels; i.e.,

$$\phi(\mathbf{x}) \approx \Phi(\mathbf{A})\mathbf{s}, \quad (1)$$

where  $\mathbf{s} \in \mathbb{R}^J$  is an unknown coding coefficient vector of  $\phi(\mathbf{x})$ . To recover a coding coefficient vector  $\mathbf{s}$  from  $\phi(\mathbf{x})$  and  $\Phi(\mathbf{A})$  stably, the regularization method is the best choice, and the corresponding optimization problem can be written as follows:

$$\min_{\mathbf{s}} \frac{1}{2} \|\phi(\mathbf{x}) - \Phi(\mathbf{A})\mathbf{s}\|_2^2 + \lambda \|\mathbf{s}\|_{\ell_q}, \quad (2)$$

where  $\lambda > 0$  is a regularization parameter, and  $\ell_q = 0, 1$ , or  $2$ .

Using different  $\ell_q$  will lead to different instantiations of KCR-based classification. KSRC and KCRC are two instantiations, where  $\ell_q$  is respectively set to 1 and 2. The corresponding optimization problems can be respectively written as follows:

$$\text{KSRC: } \min_{\mathbf{s}} \frac{1}{2} \|\phi(\mathbf{x}) - \Phi(\mathbf{A})\mathbf{s}\|_2^2 + \lambda \|\mathbf{s}\|_1 \quad \text{and} \quad \text{KCRC: } \min_{\mathbf{s}} \frac{1}{2} \|\phi(\mathbf{x}) - \Phi(\mathbf{A})\mathbf{s}\|_2^2 + \frac{\lambda}{2} \|\mathbf{s}\|_2^2. \quad (3)$$

After solving the above optimization problems, the obtained  $\mathbf{s}$  is used for the final classification. For KSRC, the class label  $y$  of  $\mathbf{x}$  is determined via the minimal residual between  $\phi(\mathbf{x})$  and its approximation from the mapped training pixels of each class, and the classification rule can be written as follows:

$$\text{dist: } y = \arg \min_{c=1,2,\dots,C} \|\phi(\mathbf{x}) - \Phi(\mathbf{A})\delta_c(\mathbf{s})\|_2^2, \quad (4)$$

where  $\delta_c(\cdot)$  is the characteristic function that selects coefficients related to the  $c$ th class and makes the rest zero. For KCRC, it considers the discriminative information brought by  $\mathbf{s}$ , and modifies the classification rule as:

$$y = \arg \min_{c=1,2,\dots,C} \|\phi(\mathbf{x}) - \Phi(\mathbf{A})\delta_c(\mathbf{s})\|_2^2 / \|\delta_c(\mathbf{s})\|_2^2. \quad (5)$$

Notably, all  $\phi$  mappings used in kernel methods occur in the form of inner products. For every two pixels  $\mathbf{x}_i$  and  $\mathbf{x}_j$ , we can define a kernel function as:

$$\mathcal{K} = \langle \phi(\mathbf{x}_i), \phi(\mathbf{x}_j) \rangle, \quad (6)$$

where  $\langle \cdot, \cdot \rangle$  represents the inner product. In this paper, only the radial basis function (RBF) kernel ( $\mathcal{K}(\mathbf{x}_i, \mathbf{x}_j) = \exp(-\gamma \|\mathbf{x}_i - \mathbf{x}_j\|_2^2)$ ,  $\gamma > 0$ ) is considered, owing to its simplicity and empirically observed good performance [12,19,21,49]. After defining the kernel function  $\mathcal{K}$ , the optimization problems of KSRC and KCRC can be rewritten as:

$$\text{KSRC: } \min_{\mathbf{s}} \frac{1}{2} \mathbf{s}^T \mathbf{Q} \mathbf{s} - \mathbf{s}^T \mathbf{b} + \lambda \|\mathbf{s}\|_1 \quad \text{and} \quad \text{KCRC: } \min_{\mathbf{s}} \frac{1}{2} \mathbf{s}^T \mathbf{Q} \mathbf{s} - \mathbf{s}^T \mathbf{b} + \frac{\lambda}{2} \|\mathbf{s}\|_2^2, \quad (7)$$

where the constant terms are dropped,  $\mathbf{Q} = \langle \Phi(\mathbf{A}), \Phi(\mathbf{A}) \rangle \in \mathbb{R}^{J \times J}$  is a positive semi-definite matrix with entry  $Q_{ij} = \mathcal{K}(\mathbf{a}_i, \mathbf{a}_j)$  and the mapped test vector  $\mathbf{b} = \langle \Phi(\mathbf{A}), \phi(\mathbf{x}) \rangle = [\mathcal{K}(\mathbf{a}_1, \mathbf{x}), \dots, \mathcal{K}(\mathbf{a}_J, \mathbf{x})] \in \mathbb{R}^J$ . Similarly, the classification rules of KSRC and KCRC can be rewritten as:

$$y = \arg \min_{c=1,2,\dots,C} \delta_c^T(\mathbf{s}) \mathbf{Q} \delta_c(\mathbf{s}) - 2\delta_c^T(\mathbf{s}) \mathbf{b} \quad (8)$$

$$y = \arg \min_{c=1,2,\dots,C} \left( \delta_c^T(\mathbf{s}) \mathbf{Q} \delta_c(\mathbf{s}) - 2\delta_c^T(\mathbf{s}) \mathbf{b} + 1 \right) / \delta_c^T(\mathbf{s}) \delta_c(\mathbf{s}) \quad (9)$$

The optimization problem of KSRC is convex but not smooth. For this type of problem, several algorithms proposed in the sparse representation and compressive sensing community can be adopted to solve it [50–52]. In this paper, an alternating direction method of multipliers (ADMM) algorithm [53] is adopted owing to its flexibility and availability, and the details can be seen in [49]. As for the optimization problem of KCRC, it is convex and smooth, and an analytical solution can be derived (see [23]).

### 3. Proposed Approach

#### 3.1. Pixel-Wise Classification via KFCLS

##### 3.1.1. Problem Formulation

Works in [14,15] point out that if the dimensionality and discriminability of a test sample is high, the estimated coefficient vector will be naturally sparse and concentrate on the training samples whose class labels are the same as that of the test sample, regardless of whether the  $\ell_1$ -norm or  $\ell_2$ -norm is used. Since in the kernel feature space the dimensionality of a test sample is very high and its discriminability is enhanced, KCRC can get the same performance as KSRC [23]. Considering these issues, one may wonder whether other constraint terms can play the same role, except the  $\ell_q$ -norm regularization terms. Notably, in the coefficient vector  $\mathbf{s}$ , each entry can be treated as the similarity between the corresponding training pixel and the test pixel. If the test pixel is similar to some training

pixels, large values will be assigned to the corresponding entries of  $\mathbf{s}$ ; otherwise, small values (may be negative) will be assigned. It is natural to enforce the similarity to be nonnegative. Moreover, the nonnegative constraint can promote the sparsity of the coefficient vector [28,29]. For this reason, we consider the KNLS problem, which is defined as follows:

$$\text{KNLS : } \min_{\mathbf{s}} \frac{1}{2} \mathbf{s}^T \mathbf{Q} \mathbf{s} - \mathbf{s}^T \mathbf{b} \text{ subject to } \mathbf{s} \succeq \mathbf{0}_J, \quad (10)$$

where  $\mathbf{0}_J \in \mathbb{R}^J$  is a zero vector with all entries being 0, and the symbol  $\succeq$  denotes component-wise inequality; i.e.,  $\mathbf{s} \succeq \mathbf{0}_J$  means entry  $s_j \geq 0$  for  $j = 1, 2, \dots, J$ . Since  $\mathbf{s}$  is nonnegative and reflects the similarity, it can be regarded as a probability distribution if we enforce the summation of its entries to be one [54]. Accordingly, we obtain KFCLS and the corresponding problem can be written as follows:

$$\text{KFCLS : } \min_{\mathbf{s}} \frac{1}{2} \mathbf{s}^T \mathbf{Q} \mathbf{s} - \mathbf{s}^T \mathbf{b} \text{ subject to } \mathbf{s} \succeq \mathbf{0}_J, \mathbf{1}_J^T \mathbf{s} = 1, \quad (11)$$

where  $\mathbf{1}_J \in \mathbb{R}^J$  is a vector with all entries being 1. Figure 1 shows a comparison of the coefficient vectors obtained by KNLS, KFCLS, and KSRC. It can be observed that the coefficients of KNLS and KFCLS are almost as sparse as those of KSRC. Although the number of training pixels of each class may be unequal, the summation of the entries of  $\delta_c(\mathbf{s})$  can reflect the similarity between the  $c$ th class and the test pixel [23]. Figure 2 shows the summation of the entries of each  $\delta_c(\mathbf{s})$ . It is apparent that the summation value of the true class label is predominant. Moreover, the outputs of a classifier should be calibrated posterior probabilities to facilitate the subsequent processing, which are very useful in spatial-spectral classification [23,36,37]. With the aforementioned observation and motivation in mind, we have designed a posterior probability in this context as follows:

$$p(y = c|\mathbf{x}) = (\mathbf{T}\mathbf{s})_c, \quad (12)$$

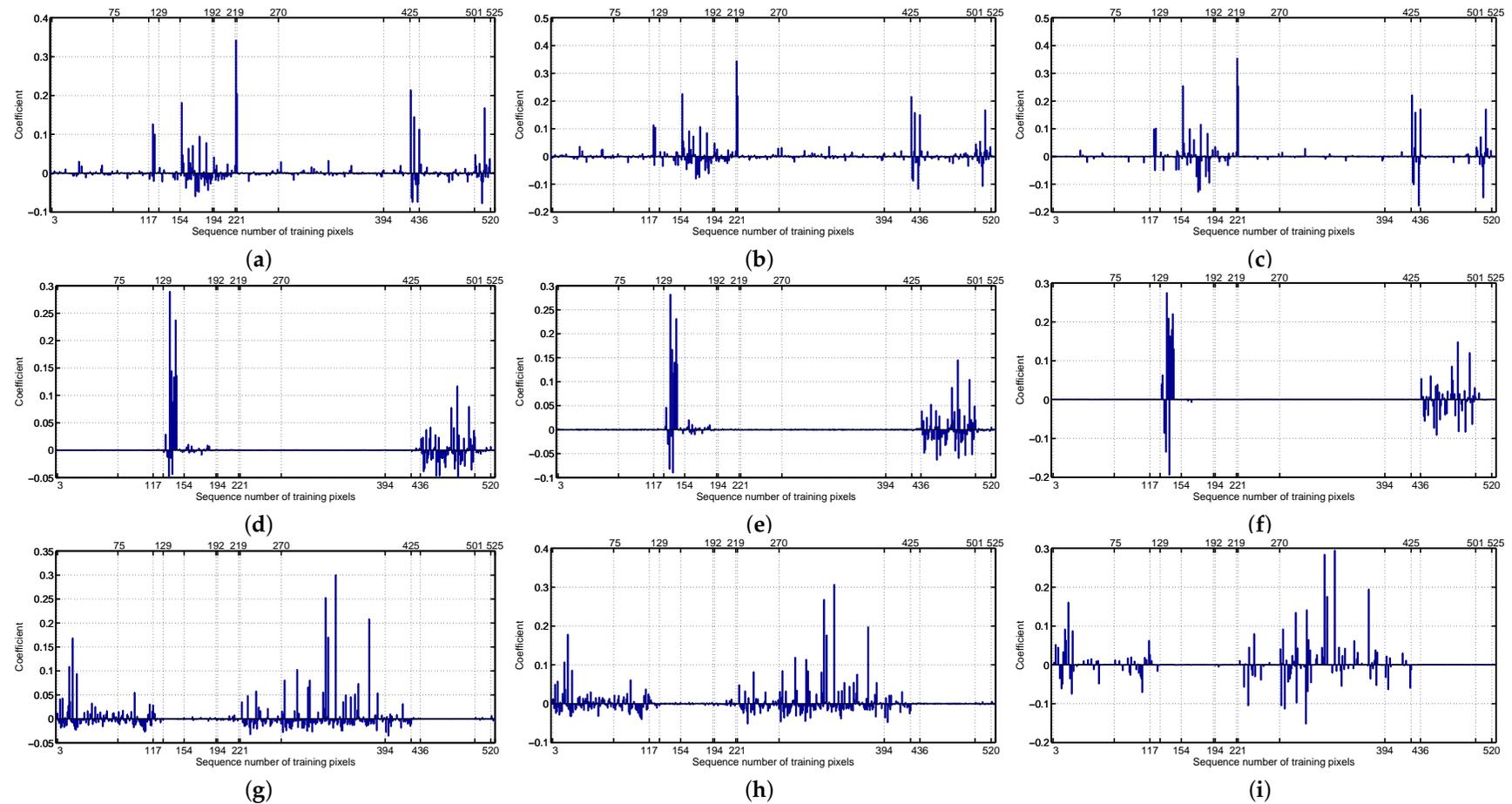
where  $(\cdot)_c$  denotes the  $c$ th entry of a vector, and the summation matrix  $\mathbf{T} \in \mathbb{R}^{C \times J}$  is defined by

$$\mathbf{T}_{cj} = \begin{cases} 1 & \text{if class}(\mathbf{a}_j) = c, \forall c, j. \\ 0 & \text{else} \end{cases} \quad (13)$$

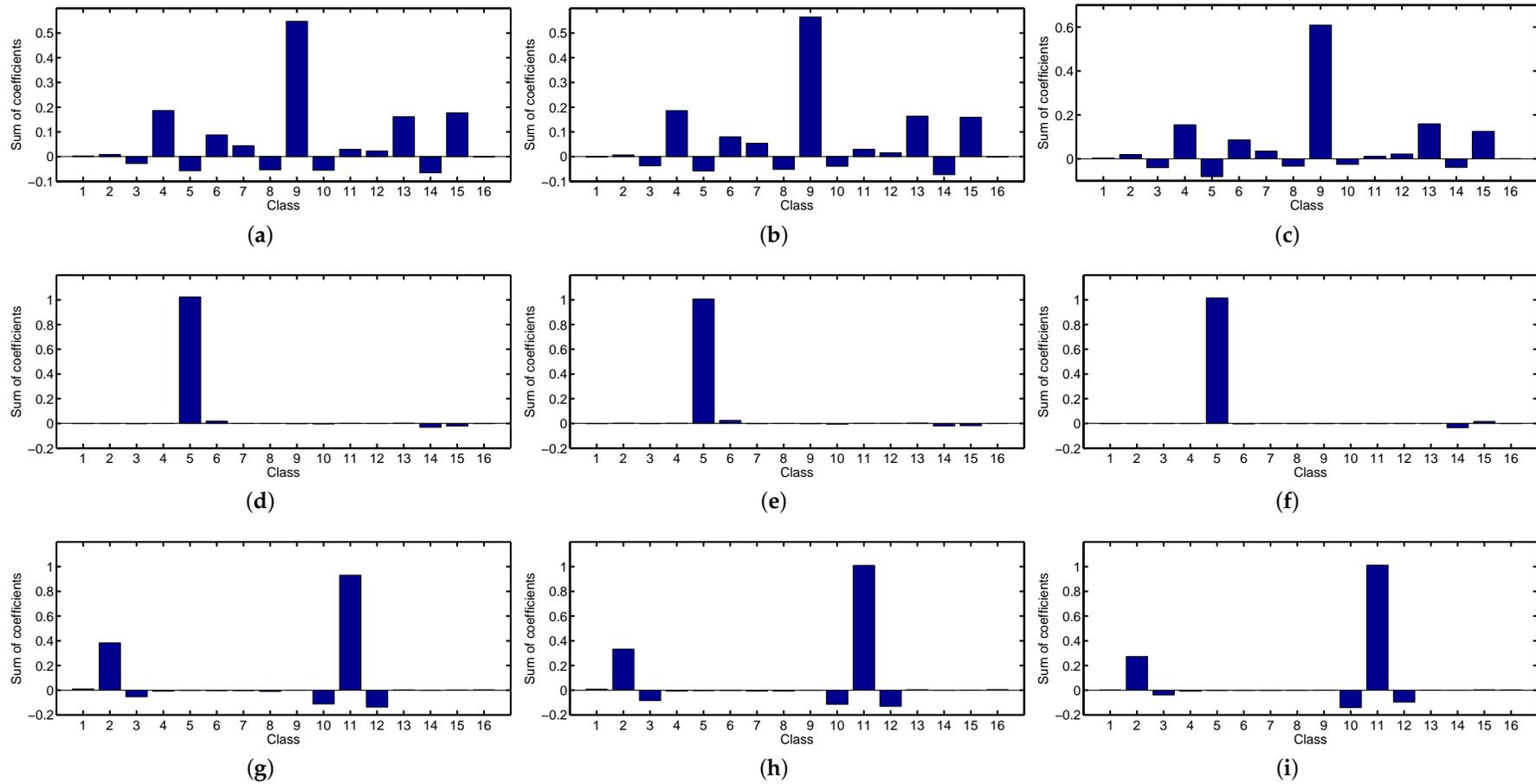
With the definition of the posterior probability, the classification rule of KFCLS can be written as follows:

$$\text{prob: } y = \arg \max_{c=1,2,\dots,C} p(y = c|\mathbf{x}). \quad (14)$$

As for KNLS, we use the classification rule (4) in this paper. Notably, the classification rule (4) is also suitable for KFCLS.



**Figure 1.** Estimated coefficients for the pixels in the Airborne Visible/Infrared Imaging Spectrometer (AVIRIS) Indian Pine dataset (about 5% training pixels are used, see Section 4), the corresponding class labels are included in the parentheses, and all coefficients are ranked in order of the class labels. (a–c): Pixel taken from Class C9, and the coefficients of C9 are in the range (219, 221]. (d–f): Pixel taken from Class C5, and the coefficients of C5 are in the range (129, 154]. (g–i): Pixel taken from Class C11, and the coefficients of C11 are in the range (270, 394]. (a) KNLS (C9). (b) KFCLS (C9). (c) KSRC (C9). (d) KNLS (C5). (e) KFCLS (C5). (f) KSRC (C5). (g) KNLS (C11). (h) KFCLS (C11). (i) KSRC (C11). Notably, alternating direction method of multipliers (ADMM) is used to solve the optimization problems of KNLS and KFCLS, and thus their coefficients are not strictly nonnegative.



**Figure 2.** Sum of the estimated coefficients for the pixels in the Airborne Visible/Infrared Imaging Spectrometer (AVIRIS) Indian Pines dataset (about 5% training pixels are used, see Section 4), the corresponding class labels are included in the parentheses. (a–c): Pixel taken from Class C9. (d–f): Pixel taken from Class C5. (g–i): Pixel taken from Class C11. (a) KNLS (C9). (b) KFCLS (C9). (c) KSRC (C9). (d) KNLS (C5). (e) KFCLS (C5). (f) KSRC (C5). (g) KNLS (C11). (h) KFCLS (C11). (i) KSRC (C11).

### 3.1.2. Optimization Algorithm

In this paper, ADMM is adopted to solve the optimization problems (10) and (11). For KFCLS, the optimization problem (11) can be rewritten as the following equivalent form:

$$\min_{\mathbf{s}} \frac{1}{2} \mathbf{s}^T \mathbf{Q} \mathbf{s} - \mathbf{s}^T \mathbf{b} + \iota_{\mathbb{R}_+^J}(\mathbf{s}) + \iota_1(\mathbf{1}_J^T \mathbf{s}), \quad (15)$$

where  $\iota_S$  is the indicator function of the set  $S$  (i.e.,  $\iota_S(\mathbf{x}) = 0$  if  $\mathbf{x} \in S$  and  $\iota_S(\mathbf{x}) = \infty$  if  $\mathbf{x} \notin S$ ). By introducing a variable  $\mathbf{v} \in \mathbb{R}^J$ , the optimization problem (15) can be rewritten as follows:

$$\min_{\mathbf{s}, \mathbf{v}} \frac{1}{2} \mathbf{s}^T \mathbf{Q} \mathbf{s} - \mathbf{s}^T \mathbf{b} + \iota_{\mathbb{R}_+^J}(\mathbf{v}) + \iota_1(\mathbf{1}_J^T \mathbf{s}) \quad \text{subject to } \mathbf{s} = \mathbf{v}. \quad (16)$$

The augmented Lagrangian function of (16) can be written as follows:

$$\mathcal{L}(\mathbf{s}, \mathbf{v}, \mathbf{d}) = \frac{1}{2} \mathbf{s}^T \mathbf{Q} \mathbf{s} - \mathbf{s}^T \mathbf{b} + \iota_{\mathbb{R}_+^J}(\mathbf{v}) + \iota_1(\mathbf{1}_J^T \mathbf{s}) + \frac{\mu}{2} \|\mathbf{s} - \mathbf{v} - \mathbf{d}\|_2^2, \quad (17)$$

where  $\mu > 0$  is the penalty parameter and  $\mathbf{d} \in \mathbb{R}^J$  is an auxiliary variable. The ADMM iteration procedure can be written as:

$$\begin{cases} \mathbf{s}^t &= \arg \min_{\mathbf{s}} \frac{1}{2} \mathbf{s}^T \mathbf{Q} \mathbf{s} - \mathbf{s}^T \mathbf{b} + \iota_1(\mathbf{1}_J^T \mathbf{s}) + \frac{\mu}{2} \|\mathbf{s} - \mathbf{v}^{t-1} - \mathbf{d}^{t-1}\|_2^2 \\ \mathbf{v}^t &= \arg \min_{\mathbf{v}} \iota_{\mathbb{R}_+^J}(\mathbf{v}) + \frac{\mu}{2} \|\mathbf{s}^t - \mathbf{v} - \mathbf{d}^{t-1}\|_2^2 \\ \mathbf{d}^t &= \mathbf{d}^{t-1} - (\mathbf{s}^t - \mathbf{v}^t) \end{cases} \quad (18)$$

where  $t > 0$  is the iteration number. The first step of (18) is to solve the  $\mathbf{s}$ -subproblem, and the solution can be derived as:

$$\mathbf{s}^t = \mathcal{P}_1(\mathbf{F}^{-1}(\mathbf{b} + \mu(\mathbf{v}^{t-1} + \mathbf{d}^{t-1}))), \quad (19)$$

where  $\mathbf{F} = \mathbf{Q} + \mu \mathbf{I}$  with  $\mathbf{I}$  being the identity matrix, and the projection operator  $\mathcal{P}_1(\mathbf{s}) = \mathbf{s} - \mathbf{F}^{-1} \mathbf{1}_J (\mathbf{1}_J^T \mathbf{F}^{-1} \mathbf{1}_J)^{-1} (\mathbf{1}_J^T \mathbf{s} - 1)$ . The second step of (18) is to solve the  $\mathbf{v}$ -subproblem, which is the well-known proximal operator [55]:

$$\mathbf{v}^t = \max(\mathbf{s}^t - \mathbf{d}^{t-1}, 0), \quad (20)$$

where  $\max(\cdot, 0)$  is used to set the negative components to zero and keep the nonnegative components unchanged. The last step of (18) is used to update the auxiliary variable. The algorithm of KFCLS is detailed as follows.

1. **Input:** A training dictionary  $\mathbf{A} \in \mathbb{R}^{L \times J}$ , and a hyperspectral pixel  $\mathbf{x} \in \mathbb{R}^L$ .
2. Select the parameter  $\gamma$  for the RBF kernel and compute the matrix  $\mathbf{Q}$  and the vector  $\mathbf{b}$ .
3. Set  $t = 1$ , choose  $\mu, \mathbf{s}^1, \mathbf{v}^1, \mathbf{d}^1$ .
4. **Repeat**
5. Compute  $\mathbf{s}^t, \mathbf{v}^t, \mathbf{d}^t$  using (18).
6.  $t = t + 1$ .
7. **Until** some stopping criterion is satisfied.
8. **Output:** The estimated label of  $\mathbf{x}$  using (14) or (4).

As for KNLS, its ADMM iteration procedure is almost as same as that of KFCLS, where we do not need the projection operator  $\mathcal{P}_1$  in (19).

### 3.2. Spatial-Spectral Classification

The suggested KFCLS is just a pixel-wise classifier that does not treat hyperspectral data as images but as an unordered list of pixels. In order to incorporate the spatial-spectral information, several

methods have been proposed as discussed in Section 1. Among these methods, the regularization strategy is an important one for CR-based classification, since CR-based classification is also a group of regularization methods. In this section, we show the incorporation of spatial-spectral information for KFCLS using both the co-processing and post-processing methods, and consider two regularization strategies to combine the spatial-spectral information.

### 3.2.1. Problem Formulation

Suppose that a hyperspectral image is composed of a set of  $I$  pixels  $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_I] \in \mathbb{R}^{L \times I}$ . Correspondingly, we get the coefficient matrix  $\mathbf{S} = [\mathbf{s}_1, \mathbf{s}_2, \dots, \mathbf{s}_I] \in \mathbb{R}^{J \times I}$ , the probability matrix  $\mathbf{P} = [\mathbf{p}_1, \mathbf{p}_2, \dots, \mathbf{p}_I] \in \mathbb{R}^{C \times I}$  with  $(\mathbf{p}_i)_c = p(y = c | \mathbf{x}_i)$ , and the mapped test matrix  $\mathbf{B} = [\mathbf{b}_1, \mathbf{b}_2, \dots, \mathbf{b}_I] \in \mathbb{R}^{J \times I}$ . Then, the unconstrained optimization problem (15) for  $\mathbf{X}$  can be written as:

$$\min_{\mathbf{S}} \frac{1}{2} \text{Tr}(\mathbf{S}^T \mathbf{Q} \mathbf{S}) - \text{Tr}(\mathbf{S}^T \mathbf{B}) + \iota_{\mathbb{R}_+^{J \times I}}(\mathbf{S}) + \iota_{\mathbf{1}_I^T}(\mathbf{1}_I^T \mathbf{S}), \quad (21)$$

where  $\text{Tr}(\cdot)$  denotes the trace of a matrix. In this paper, the spatial relationship between two adjacent pixels  $\mathbf{x}_i$  and  $\mathbf{x}_j$  is modeled by the similarity defined as

$$W_{ij} = \exp(-\beta \|\bar{\mathbf{x}}_i - \bar{\mathbf{x}}_j\|_2) + \epsilon, \beta > 0 \quad (22)$$

where  $\epsilon = 10^{-6}$  is a small positive constant and  $\bar{\mathbf{x}}_i$  and  $\bar{\mathbf{x}}_j$  are the pixels of the first three principle components of the hyperspectral image  $\mathbf{X}$ . For each pixel  $\mathbf{x}_i$ , its neighborhood  $\mathcal{N}_i$  is built by its eight spatially-adjacent neighbors, and  $W_{ij}$  is set to 0 if  $\mathbf{x}_j$  does not belong to  $\mathcal{N}_i$ .

### 3.2.2. Co-Processing Methods

The spatial arrangement of the coefficient matrix  $\mathbf{S}$  is associated with that of the hyperspectral image  $\mathbf{X}$ . That is to say, the spatial relationship between every two pixels  $\mathbf{x}_i$  and  $\mathbf{x}_j$  is also suitable for that of the coefficient vectors  $\mathbf{s}_i$  and  $\mathbf{s}_j$ . It is natural to integrate the spatial information of  $\mathbf{X}$  by enforcing  $\mathbf{S}$ . If  $\mathbf{x}_i$  is similar to  $\mathbf{x}_j$  (i.e.,  $W_{ij}$  is relatively large),  $\mathbf{s}_i$  and  $\mathbf{s}_j$  should be close to each other, and vice versa. In this paper, the weighted H1-norm that is convex and smooth is adopted to describe the aforementioned relationship and the joint regularization model (JRM) can be written as follows:

$$\mathbf{JRM} : \min_{\mathbf{S}} \frac{1}{2} \text{Tr}(\mathbf{S}^T \mathbf{Q} \mathbf{S}) - \text{Tr}(\mathbf{S}^T \mathbf{B}) + \iota_{\mathbb{R}_+^{J \times I}}(\mathbf{S}) + \iota_{\mathbf{1}_I^T}(\mathbf{1}_I^T \mathbf{S}) + \frac{\lambda}{4} \|\nabla_w \mathbf{S}\|_F^2, \quad (23)$$

where  $\lambda > 0$ ,  $\|\cdot\|_F$  denotes the Frobenius norm, and  $\|\nabla_w \mathbf{S}\|_F^2$  is the weighted H1-norm of  $\mathbf{S}$  with  $\nabla_w \mathbf{s}_i = \{\sqrt{W_{ij}}(\mathbf{s}_i - \mathbf{s}_j) | j \in \mathcal{N}_i\}$ . We may note that the spatial arrangement of the probability matrix  $\mathbf{P}$  is also associated with that of the hyperspectral image  $\mathbf{X}$ . That is to say, we can integrate the spatial information of  $\mathbf{X}$  by enforcing  $\mathbf{P}$  (i.e.,  $\mathbf{TS}$ ). In view of this, we propose the following class-level JRM (CJRM):

$$\mathbf{CJRM} : \min_{\mathbf{S}} \frac{1}{2} \text{Tr}(\mathbf{S}^T \mathbf{Q} \mathbf{S}) - \text{Tr}(\mathbf{S}^T \mathbf{B}) + \iota_{\mathbb{R}_+^{J \times I}}(\mathbf{S}) + \iota_{\mathbf{1}_I^T}(\mathbf{1}_I^T \mathbf{S}) + \frac{\lambda}{4} \|\nabla_w \mathbf{P}\|_F^2 \text{ subject to } \mathbf{P} = \mathbf{TS}. \quad (24)$$

Since the objective solutions of JRM and CJRM are the coefficient matrix  $\mathbf{S}$  and the columns of which sum to one, both the classification rules (4) and (14) are suitable for the final classification.

### 3.2.3. Post-Processing Methods

For the post-processing methods, the procedures of the pixel-wise classification and the integration of spatial-spectral information are performed separately. Generally, the incorporation of spatial-spectral information can be done by refining the coefficient matrix  $\mathbf{S}$  [41] or the probability

matrix  $\mathbf{P}$  [23,36,37,56]. For this issue, we propose the corresponding post-processing regularization model (PRM) and class-level PRM (CPRM) for KFCLS, which are defined as follows:

$$\text{PRM : } \min_{\mathbf{V}} \frac{1}{2} \|\mathbf{S} - \mathbf{V}\|_F^2 + \frac{\lambda}{4} \|\nabla_w \mathbf{V}\|_F^2 \quad (25)$$

$$\text{CPRM : } \min_{\mathbf{U}} \frac{1}{2} \|\mathbf{P} - \mathbf{U}\|_F^2 + \frac{\lambda}{4} \|\nabla_w \mathbf{U}\|_F^2 \quad (26)$$

Notably, we can verify that the columns of solutions  $\mathbf{V}$  and  $\mathbf{U}$  sum to one with reference to (34). The objective solution of PRM is the refined coefficient matrix  $\mathbf{V}$ , and thus both the classification rules (4) and (14) are suitable for the final classification; whereas the objective solution of CPRM is the refined probability matrix  $\mathbf{U}$ , and thus only the classification rule (14) is suitable for classification purposes. Furthermore, we can prove that PRM using the classification rule (14) is equivalent to CPRM using the classification rule (14) with reference to (34). Accordingly, PRM using the classification rule (14) is dropped in the experiments.

### 3.2.4. Optimization Algorithm

In this paper, the optimization problems (23) and (24) are solved by ADMM. For CJRM, the optimization problem (24) can be rewritten as the following formulation by introducing a variable  $\mathbf{V} \in \mathbb{R}^{J \times I}$ :

$$\min_{\mathbf{S}, \mathbf{V}, \mathbf{P}} \frac{1}{2} \text{Tr}(\mathbf{S}^T \mathbf{Q} \mathbf{S}) - \text{Tr}(\mathbf{S}^T \mathbf{B}) + \iota_{\mathbb{R}_+^{J \times I}}(\mathbf{V}) + \iota_{\mathbf{1}_I^T}(\mathbf{1}_J^T \mathbf{S}) + \frac{\lambda}{4} \|\nabla_w \mathbf{P}\|_F^2 \quad \text{subject to } \mathbf{P} = \mathbf{T} \mathbf{S}, \mathbf{V} = \mathbf{S}. \quad (27)$$

The optimization problem (27) accords with the framework of ADMM, and the corresponding augmented Lagrangian function of (27) can be written as:

$$\begin{aligned} \mathcal{L}(\mathbf{S}, \mathbf{V}, \mathbf{D}, \mathbf{P}, \mathbf{R}) = & \frac{1}{2} \text{Tr}(\mathbf{S}^T \mathbf{Q} \mathbf{S}) - \text{Tr}(\mathbf{S}^T \mathbf{B}) + \iota_{\mathbb{R}_+^{J \times I}}(\mathbf{V}) + \iota_{\mathbf{1}_I^T}(\mathbf{1}_J^T \mathbf{S}) + \frac{\lambda}{4} \|\nabla_w \mathbf{P}\|_F^2 \\ & + \frac{\mu}{2} \|\mathbf{S} - \mathbf{V} - \mathbf{D}\|_F^2 + \frac{\mu}{2} \|\mathbf{T} \mathbf{S} - \mathbf{P} - \mathbf{R}\|_F^2, \end{aligned} \quad (28)$$

where  $\mathbf{D} \in \mathbb{R}^{J \times I}$  and  $\mathbf{R} \in \mathbb{R}^{C \times I}$  are two auxiliary variables. Then, the optimization problem (27) can be solved by the following ADMM iterations:

$$\begin{cases} \mathbf{S}^t = \arg \min_{\mathbf{S}} \mathcal{L}(\mathbf{S}, \mathbf{V}^{t-1}, \mathbf{D}^{t-1}, \mathbf{P}^{t-1}, \mathbf{R}^{t-1}) \\ \mathbf{V}^t = \arg \min_{\mathbf{V}} \mathcal{L}(\mathbf{S}^t, \mathbf{V}, \mathbf{D}^{t-1}, \mathbf{P}^{t-1}, \mathbf{R}^{t-1}) \\ \mathbf{P}^t = \arg \min_{\mathbf{P}} \mathcal{L}(\mathbf{S}^t, \mathbf{V}^t, \mathbf{D}^{t-1}, \mathbf{P}, \mathbf{R}^{t-1}) \\ \mathbf{D}^t = \mathbf{D}^{t-1} - (\mathbf{S}^t - \mathbf{V}^t) \\ \mathbf{R}^t = \mathbf{R}^{t-1} - (\mathbf{T} \mathbf{S}^t - \mathbf{P}^t) \end{cases} \quad (29)$$

Similar to (18), the solutions of the first two subproblems of (29) can be derived as:

$$\mathbf{S}^t = \mathcal{P}_{\mathbf{1}_I^T}(\mathbf{F}^{-1}(\mathbf{B} + \mu(\mathbf{V}^{t-1} + \mathbf{D}^{t-1}) + \mu \mathbf{T}^T(\mathbf{P}^{t-1} + \mathbf{R}^{t-1}))) \quad (30)$$

$$\mathbf{V}^t = \max(\mathbf{S}^t - \mathbf{D}^{t-1}, 0) \quad (31)$$

where  $\mathbf{F} = \mathbf{Q} + \mu \mathbf{I} + \mu \mathbf{T}^T \mathbf{T}$ , and the projection operator  $\mathcal{P}_{\mathbf{1}_I^T}(\mathbf{S}) = \mathbf{S} - \mathbf{F}^{-1} \mathbf{1}_J (\mathbf{1}_J^T \mathbf{F}^{-1} \mathbf{1}_J)^{-1} (\mathbf{1}_J^T \mathbf{S} - \mathbf{1}_I^T)$ . The third subproblem of (29) can be written as follows:

$$\min_{\mathbf{P}} \frac{\mu}{2} \|\mathbf{P} - (\mathbf{T} \mathbf{S}^t - \mathbf{R}^{t-1})\|_F^2 + \frac{\lambda}{4} \|\nabla_w \mathbf{P}\|_F^2. \quad (32)$$

The optimization problem (32) is a linear system, which can be solved by the Gauss–Seidel method according to [48,51]. In addition, the optimization problem (32) can also be rewritten as the following formulation:

$$\min_{\mathbf{P}} \frac{\mu}{2} \|\mathbf{P} - (\mathbf{TS}^t - \mathbf{R}^{t-1})\|_F^2 + \frac{\lambda}{2} \text{Tr}(\mathbf{PGP}^T), \quad (33)$$

where  $\mathbf{G} \in \mathbb{R}^{I \times I}$  can be treated as the graph Laplacian with  $G_{ii} = \sum_j W_{ij}$  and  $G_{ij} = -W_{ij}$  ( $j \neq i$ ). The analytical solution of (33) can be derived as:

$$\mathbf{P} = (\mathbf{TS}^t - \mathbf{R}^{t-1}) \left( \frac{\lambda}{\mu} \mathbf{G} + \mathbf{I} \right)^{-1}. \quad (34)$$

The algorithm of CJRM is detailed as follows.

1. **Input:** A training dictionary  $\mathbf{A} \in \mathbb{R}^{L \times J}$ , and a hyperspectral data matrix  $\mathbf{X} \in \mathbb{R}^{L \times I}$ .
2. Choose  $\beta$  and compute the weights  $W_{ij}$  according to (22).
3. Select the parameter  $\gamma$  for the RBF kernel and compute the matrices  $\mathbf{Q}$  and  $\mathbf{B}$ .
4. Set  $t = 1$ , choose  $\mu, \lambda, \mathbf{S}^1, \mathbf{V}^1, \mathbf{P}^1, \mathbf{D}^1, \mathbf{R}^1$ .
5. **Repeat**
6. Compute  $\mathbf{S}^t, \mathbf{V}^t, \mathbf{P}^t, \mathbf{D}^t, \mathbf{R}^t$  using (29).
7.  $t = t + 1$ .
8. **Until** some stopping criterion is satisfied.
9. **Output:** The estimated label of  $\mathbf{x}_i$  using (4) or (14),  $i = 1, \dots, I$ .

As for JRM, the aforementioned procedure is also suitable for the optimization problem (23). By changing the summation matrix  $\mathbf{T}$  to the identity matrix  $\mathbf{I}$ , we can obtain the ADMM iterations of JRM. For the two post-processing methods PRM and CRPM, their formulations are as same as that of (32). Thus, they can be solved by the Gauss–Seidel method according to [48,51], and we can get their analytical solutions with reference to (34).

### 3.3. Discussion

The  $\ell_2$  norm characterization of coding residual (or fidelity term)—i.e., the first term of (2)—is related to the robustness of KCR-based classification to noise, as stated in [14,15]. For the proposed pixel-wise classifiers KNLS and KFCLS, they can be treated as two instantiations of KCR-based classification, where the collaborative representation mechanism is implemented by the nonnegative constraint. Therefore, their performance is almost as same as that of KSRC and KCRC, which is experimentally demonstrated in Section 4 by using four different hyperspectral scenes. Moreover, in KCR-based classification, the accuracy of one class is usually not vulnerable to the number of training samples taken from another class, since all training samples contribute collaboratively (or competitively) to represent a test sample. In Section 4, the experimental results confirm this phenomenon, where the numbers of training samples of some classes are far less than those of the others (see Tables 1 and 2). That is to say, KNLS and KFCLS are not too sensitive to class imbalance.

Because of the limitations of remote sensing sensors, a hyperspectral image may contain outliers such as noise and missing or corrupted pixels. The proposed spatial-spectral methods can cope with these pixels. Taking CPRM as an example, the optimization problem (26) can be rewritten as follows:

$$\min_{\mathbf{u}_i} \frac{1}{2} \|\mathbf{u}_i - \mathbf{p}_i\|_2^2 + \frac{\lambda}{2} \sum_{j \in \mathcal{N}_i} W_{ij} \|\mathbf{u}_i - \mathbf{u}_j\|_2^2, \quad i = 1, 2, \dots, I \quad (35)$$

where  $\mathbf{u}_i$  is the  $i$ th column vector of  $\mathbf{U}$ . The solutions of (35) can be derived as:

$$\mathbf{u}_i = \frac{\mathbf{p}_i + \lambda \sum_{j \in \mathcal{N}_i} W_{ij} \mathbf{u}_j}{1 + \lambda \sum_{j \in \mathcal{N}_i} W_{ij}}, \quad i = 1, 2, \dots, I. \quad (36)$$

Since  $\mathcal{N}_i$  is a  $3 \times 3$  neighborhood, (36) can be treated as a  $3 \times 3$  adaptive mean filtering, and thus the outliers can be smoothed by using their neighbors.

Notably, deep learning has attracted a lot of attention very recently. In hyperspectral image classification, various deep models (e.g., stacked autoencoder [57] and convolutional neural network [35,58–60]) have been proposed with the observation of good performance in terms of accuracy and flexibility. This paper proposes two new instantiations of KCR-based classification and investigates how to efficiently incorporate spatial-spectral information in the regularization framework. Compared with the deep learning-based methods, the proposed methods have limitations in generalization performance owing to the drawbacks of traditional methods, but have advantages in the requirement of training samples and computational cost as mentioned in the existing deep learning approaches [35,57–60]. In this paper, we do not expect the proposed methods to exceed the deep learning based methods. It is unfair to compare these two different kinds of methods. Moreover, it is beyond the scope of this paper.

**Table 1.** The ground reference classes in the AVIRIS Indian Pines dataset and the number of training and test pixels used in experiments.

No.	Class Name	Train	Test
C01	Alfalfa	3	51
C02	Corn-no till	72	1362
C03	Corn-min till	42	792
C04	Corn	12	222
C05	Grass/pasture	25	472
C06	Grass/trees	38	709
C07	Grass/pasture-mowed	2	24
C08	Hay-windrowed	25	464
C09	Oats	2	18
C10	Soybeans-no till	49	919
C11	Soybeans-min till	124	2344
C12	Soybean-clean till	31	583
C13	Wheat	11	201
C14	Woods	65	1229
C15	Bldg-grass-tree drives	19	361
C16	Stone-steel towers	5	90
Total		525	9841

**Table 2.** The ground reference classes in the AVIRIS Kennedy Space Center dataset and the number of training and test pixels used in experiments.

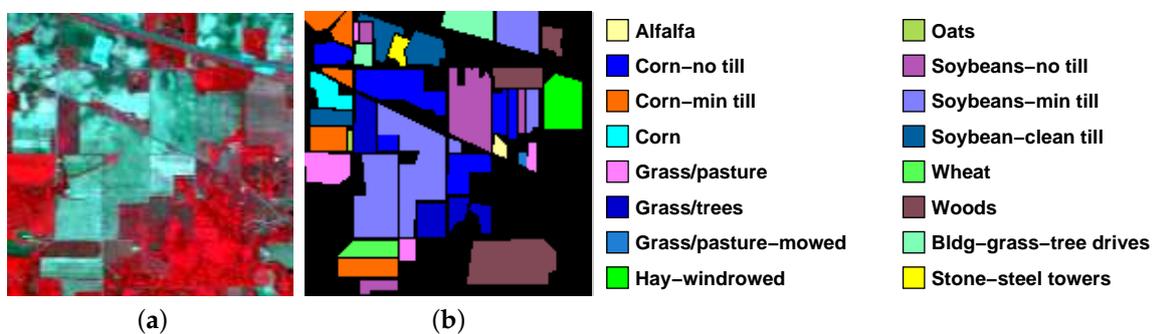
No.	Class Name	Train	Test
C01	Scrub	39	722
C02	Willow swamp	13	230
C03	Cabbage palm hammock	13	243
C04	Cabbage palm/oak hammock	13	239
C05	Slash pine	9	152
C06	Oak/broadleaf hammock	12	217
C07	Hardwood swamp	6	99
C08	Graminoid marsh	22	409
C09	Spartina marsh	26	494
C10	Cattail marsh	21	383
C11	Salt marsh	21	398
C12	Mud flats	26	477
C13	Water	47	880
Total		268	4943

## 4. Experimental Results and Analysis

### 4.1. Data Collection and Experimental Setup

In the experiments, four hyperspectral remote sensing datasets have been considered to evaluate the performance of the proposed methods.

- (1) The first one is the Indian Pines dataset taken by the Airborne Visible/Infrared Imaging Spectrometer (AVIRIS) over northwest Indiana's Indian Pines test site in 1992. This dataset contains 220 spectral bands within the wavelength range of 0.4–2.5  $\mu\text{m}$ , and consists of  $145 \times 145$  pixels. Its spectral and spatial resolutions are 10 nm and 17 m, respectively. There are sixteen ground reference classes of interest, ranging from 20 to 2468 pixels in size. Figure 3 shows the false color composite image and the ground reference map. After removing 20 water absorption bands, there are 200 spectral bands remaining in the experiments. We randomly chose about 5% of the labeled pixels for training, and used the rest for testing, as shown in Table 1.
- (2) The second one is the Kennedy Space Center dataset taken by the AVIRIS sensor over the Kennedy Space Center, Florida, in 1996. This dataset contains 224 spectral bands, covering the wavelength range of 0.4–2.5  $\mu\text{m}$ . Its spectral and spatial resolutions are 10 nm and 18 m, respectively. This image, with a size of  $512 \times 614$  pixels, contains 176 spectral bands after removing water absorption and low signal-to-noise bands. There are thirteen ground reference classes of interest, ranging from 105 to 927 pixels in size. Figure 4 shows the false color composite image and the ground reference map. In the experiments, we randomly chose 5% of the labeled pixels for training, and used the rest for testing, as shown in Table 2.
- (3) The third one—the University of Pavia dataset—is an urban image collected by the Reflective Optics System Imaging Spectrometer (ROSIS) over University of Pavia, Italy. There are 115 spectral bands in this image, covering the wavelength range of 0.43–0.86  $\mu\text{m}$ . The image consists of  $610 \times 340$  pixels, with a spatial resolution of 1.3 m per pixel. The false color composite image and the map of nine ground reference classes of interest are shown in Figure 5. In the experiments, there are 103 spectral bands remaining after the removal of noisy bands, and we randomly chose 40 pixels per class for training and used the rest for testing, as shown in Table 3.
- (4) The last one—the Center of Pavia dataset—is another urban image collected by the ROSIS sensor over the center of Pavia city. This image consists of  $1096 \times 492$  pixels, with 102 spectral bands. The reference dataset contains nine classes of interest. The false color composite image and the ground reference map are shown in Figure 6. In the experiments, we randomly chose 20 pixels per class for training and used the rest for testing, as shown in Table 4.



**Figure 3.** AVIRIS Indian Pines dataset. (a) RGB composite image. (b) Ground reference map.

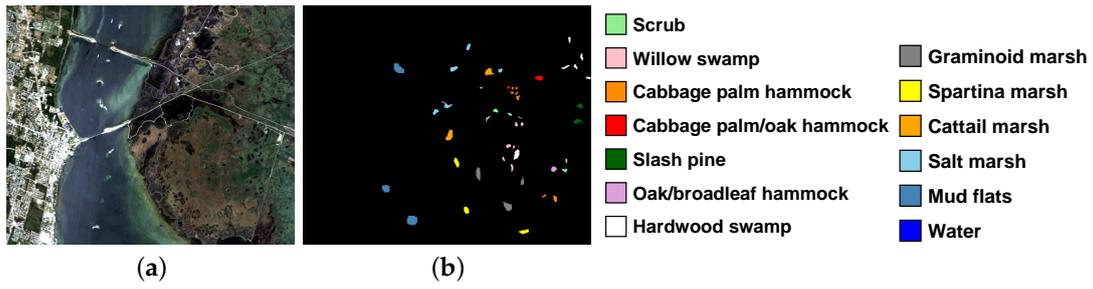


Figure 4. AVIRIS Kennedy Space Center dataset. (a) RGB composite image. (b) Ground reference map.

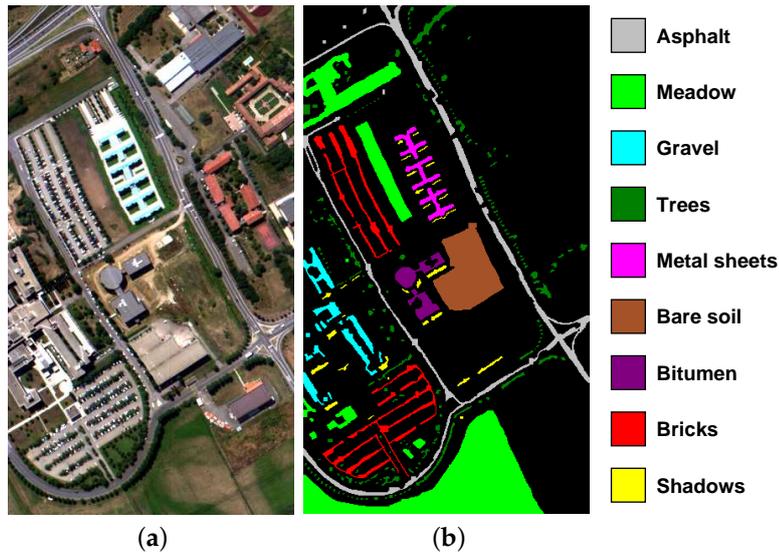
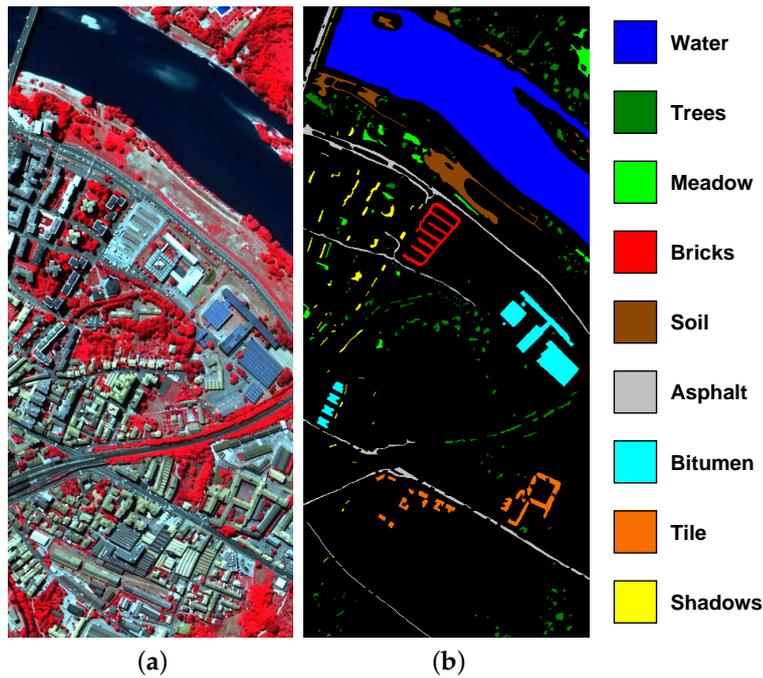


Figure 5. Reflective Optics System Imaging Spectrometer (ROSIS) University of Pavia dataset. (a) RGB composite image. (b) Ground reference map.

Table 3. The ground reference classes in the ROSIS University of Pavia dataset and the number of training and test pixels used in experiments.

No.	Class Name	Train	Test
C1	Asphalt	40	6812
C2	Meadow	40	18,646
C3	Gravel	40	2167
C4	Trees	40	3396
C5	Metal sheets	40	1338
C6	Bare soil	40	5064
C7	Bitumen	40	1316
C8	Bricks	40	3838
C9	Shadows	40	986
Total		360	43,563



**Figure 6.** ROSIS Center of Pavia dataset. (a) RGB composite image. (b) Ground reference map.

Before the following experiments, the original data were scaled in the range [0,1]. Three metrics—the overall accuracy (OA), the average accuracy (AA) and the kappa coefficient of agreement (KA)—were used to assess the classification accuracy levels, and the quantitative values were obtained by averaging ten random runs. All the experiments were performed on a 64-b quad-core CPU 3.60-GHz processor with 16-GB memory.

**Table 4.** The ground reference classes in the ROSIS Center of Pavia dataset and the number of training and test pixels used in experiments.

No.	Class Name	Train	Test
C1	Water	20	65,258
C2	Trees	20	6488
C3	Meadow	20	2885
C4	Bricks	20	2132
C5	Soil	20	6529
C6	Asphalt	20	7565
C7	Bitumen	20	7267
C8	Tile	20	3102
C9	Shadows	20	2145
Total		180	103,371

#### 4.2. Numerical and Visual Comparisons

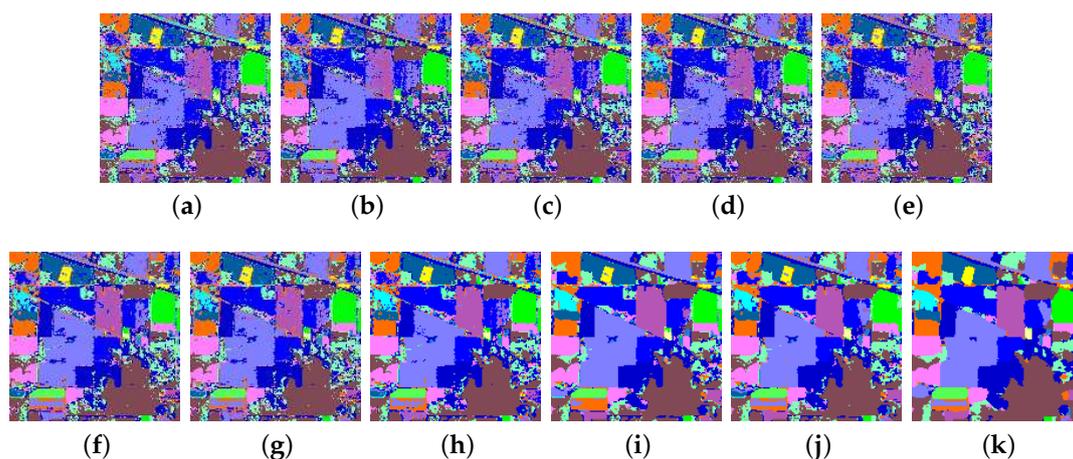
In this set of experiments, eleven investigated classification methods described above are compared numerically and visually. These methods can be divided into two categories. The first category comprises five pixel-wise KCR-based classification methods: (1) KSRC, (2) KCRC, (3) the proposed KNLS, (4) the proposed KFCLS using the classification rule “*dist*” (i.e., (4)), and (5) the proposed KFCLS using the classification rule “*prob*” (i.e., (14)). The second category comprises six spatial-spectral classification methods mentioned in Section 3.2: (1) JRM using “*dist*”, (2) JRM using

“*prob*”, (3) CJRM using “*dist*”, (4) CJRM using “*prob*”, (5) PRM, and 6) CPRM. Their parameter setting is listed in Table 5.

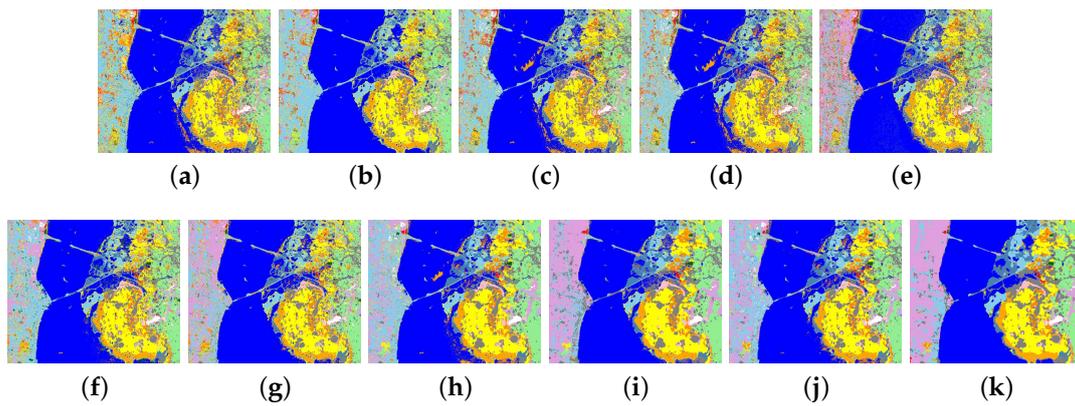
Table 6 summarizes the class-specific and global classification results for the two AVIRIS datasets, where the processing times in seconds are also included for reference. For the pixel-wise classification, the proposed KNLS and KFCLS can achieve competitive results when compared with KSRC and KCRC, and both the classification rules “*dist*” and “*prob*” are suitable for KFCLS. For the spatial-spectral classification, it can be observed that all the spatial-spectral methods perform better than the pixel-wise methods. Among the six spatial-spectral methods, CPRM yields the highest global and most of the best class-specific accuracies followed by CJRM-*prob*. The improvement of the two JRM methods over the pixel-wise methods is not significant when compared with the other four spatial-spectral methods. This is because JRM combines the spatial-spectral information by enforcing the coding coefficients directly, which is too strict and does not consider the variation of training pixels within each class. For CJRM, the classification rule “*prob*” is better than the classification rule “*dist*”. Furthermore, considering PRM using “*prob*” is equivalent to CPRM, we can conclude that the classification rule “*prob*” is more suitable for the spatial-spectral classification than the classification rule “*dist*”. As for the computational cost, KCRC is a cheap pixel-wise classifier since its objective function has an analytical solution. CJRM is the cheapest one among the four kinds of spatial-spectral methods, whereas JRM is the most expensive one.

For the two ROSIS datasets, the classification results and processing times are presented in Table 7. From this table, we can obtain almost the same conclusions as Table 6. It is apparent that KNLS and KFCLS are two competitive methods compared with KSRC and KCRC. When using the same regularization strategy and classification rule, the post-processing methods outperform the co-processing methods. For both the co-processing and post-processing methods, it is better to use the class-level regularization strategy.

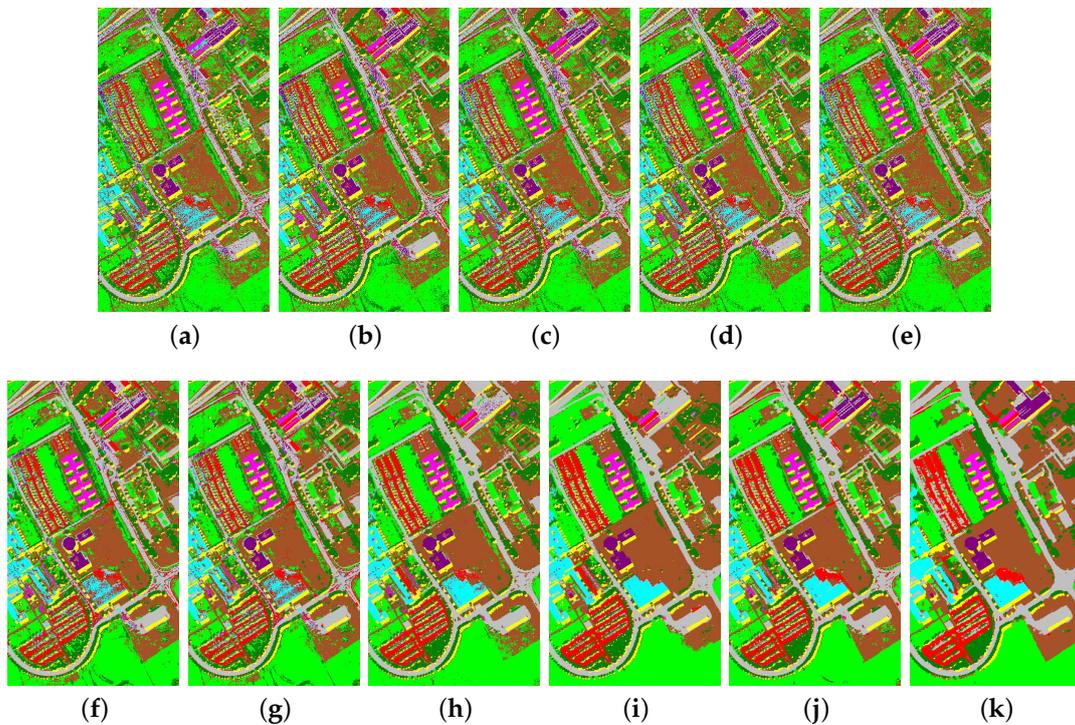
Figures 7–10 show the classification maps corresponding to one of the ten random tests in each case for the AVIRIS Indian Pines dataset, the AVIRIS Kennedy Space Center dataset, the ROSIS University of Pavia dataset, and the ROSIS Center of Pavia dataset, respectively. From these figures, it can be observed that the numerical comparisons are confirmed by inspecting these classification maps. It is evident that the maps of the spatial-spectral methods are smoother than those of the pixel-wise methods, and the maps of CPRM are closest to the ground truth maps.



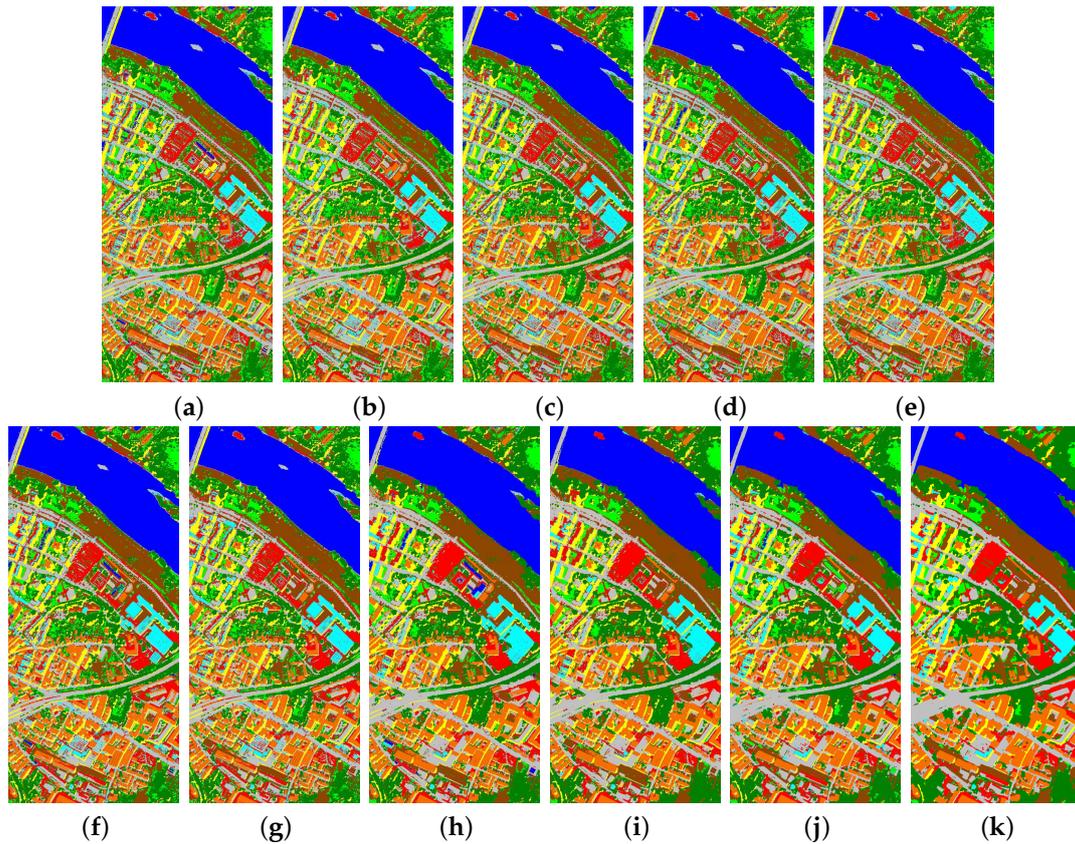
**Figure 7.** Classification maps and overall classification accuracy levels (in parentheses) obtained for the AVIRIS Indian Pines dataset using different classification methods. (a) KSRC (81.50), (b) KCRC (79.52), (c) KNLS (81.94), (d) KFCLS-*dist* (81.91), (e) KFCLS-*prob* (81.49), (f) JRM-*dist* (85.54), (g) JRM-*prob* (86.27), (h) CJRM-*dist* (89.64), (i) CJRM-*prob* (92.42), (j) PRM (91.29), (k) CPRM (92.73).



**Figure 8.** Classification maps and overall classification accuracy levels (in parentheses) obtained for the AVIRIS Kennedy Space Center dataset using different classification methods. (a) KSRC (89.86), (b) KCRC (88.99), (c) KNLS (89.90), (d) KFCLS-*dist* (89.97), (e) KFCLS-*prob* (90.17), (f) JRM-*dist* (90.59), (g) JRM-*prob* (90.57), (h) CJRM-*dist* (95.04), (i) CJRM-*prob* (95.63), (j) PRM (95.51), (k) CPRM (95.75).



**Figure 9.** Classification maps and overall classification accuracy levels (in parentheses) obtained for the ROSIS University of Pavia dataset using different classification methods. (a) KSRC (83.44), (b) KCRC (82.74), (c) KNLS (84.40), (d) KFCLS-*dist* (84.43), (e) KFCLS-*prob* (84.19), (f) JRM-*dist* (90.96), (g) JRM-*prob* (91.29), (h) CJRM-*dist* (97.23), (i) CJRM-*prob* (97.71), (j) PRM (97.98), (k) CPRM (98.57).



**Figure 10.** Classification maps and overall classification accuracy levels (in parentheses) obtained for the ROSIS Center of Pavia dataset using different classification methods. (a) KSRC (96.49), (b) KCRC (96.44), (c) KNLS (96.65), (d) KFCLS-dist (96.65), (e) KFCLS-prob (96.65), (f) JRM-dist (97.45), (g) JRM-prob (97.50), (h) CJRM-dist (98.82), (i) CJRM-prob (98.74), (j) PRM (98.84), (k) CPRM (98.93).

**Table 5.** The optimal combination of parameters for the investigated methods when applied to the four given datasets. JRM: joint regularization model; CJRM: class-level JRM; KCRC: kernel collaborative representation classification; KFCLS: kernel fully constrained least squares; KNLS: kernel nonnegative constrained least squares; KSRC: kernel sparse representation classification; PRM: post-processing regularization model; CPRM: class-level PRM.

	Pixel-Wise Classification				Spatial-Spectral Classification			
	KSRC	KCRC	KNLS	KFCLS	JRM	CJRM	PRM	CPRM
Indian Pines	$\mu = 10^{-3}$	–	$\mu = 10^{-4}$	$\mu = 10^{-4}$	$\mu = 10^{-3}$	$\mu = 10^{-4}$	$\mu = 10^{-4}$	$\mu = 10^{-4}$
	$\gamma = 2$	$\gamma = 2$	$\gamma = 2$	$\gamma = 2$	$\gamma = 2$	$\gamma = 2$	$\gamma = 2$	$\gamma = 2$
	$\lambda = 10^{-4}$	$\lambda = 10^{-3}$	–	–	$\lambda = 1$	$\lambda = 10^{-2}$	$\lambda = 10^6$	$\lambda = 10^6$
	–	–	–	–	$\beta = 100$	$\beta = 25$	$\beta = 450$	$\beta = 450$
Kennedy Space Center	$\mu = 10^{-3}$	–	$\mu = 10^{-4}$	$\mu = 10^{-4}$	$\mu = 10^{-3}$	$\mu = 10^{-4}$	$\mu = 10^{-4}$	$\mu = 10^{-4}$
	$\gamma = \frac{1}{8}$	$\gamma = \frac{1}{8}$	$\gamma = \frac{1}{8}$	$\gamma = \frac{1}{8}$	$\gamma = \frac{1}{8}$	$\gamma = \frac{1}{8}$	$\gamma = \frac{1}{8}$	$\gamma = \frac{1}{8}$
	$\lambda = 10^{-4}$	$\lambda = 10^{-3}$	–	–	$\lambda = 1$	$\lambda = 10^{-2}$	$\lambda = 10^6$	$\lambda = 10^6$
	–	–	–	–	$\beta = 100$	$\beta = 25$	$\beta = 800$	$\beta = 800$
University of Pavia	$\mu = 10^{-3}$	–	$\mu = 10^{-4}$	$\mu = 10^{-4}$	$\mu = 10^{-3}$	$\mu = 10^{-4}$	$\mu = 10^{-4}$	$\mu = 10^{-4}$
	$\gamma = \frac{1}{2}$	$\gamma = \frac{1}{2}$	$\gamma = \frac{1}{2}$	$\gamma = \frac{1}{2}$	$\gamma = \frac{1}{2}$	$\gamma = \frac{1}{2}$	$\gamma = \frac{1}{2}$	$\gamma = \frac{1}{2}$
	$\lambda = 10^{-4}$	$\lambda = 10^{-3}$	–	–	$\lambda = 1$	$\lambda = 1$	$\lambda = 10^6$	$\lambda = 10^6$
	–	–	–	–	$\beta = 100$	$\beta = 100$	$\beta = 450$	$\beta = 450$
Center of Pavia	$\mu = 10^{-3}$	–	$\mu = 10^{-4}$	$\mu = 10^{-4}$	$\mu = 10^{-3}$	$\mu = 10^{-4}$	$\mu = 10^{-4}$	$\mu = 10^{-4}$
	$\gamma = \frac{1}{8}$	$\gamma = \frac{1}{8}$	$\gamma = \frac{1}{8}$	$\gamma = \frac{1}{8}$	$\gamma = \frac{1}{8}$	$\gamma = \frac{1}{8}$	$\gamma = \frac{1}{8}$	$\gamma = \frac{1}{8}$
	$\lambda = 10^{-4}$	$\lambda = 10^{-3}$	–	–	$\lambda = 1$	$\lambda = 1$	$\lambda = 10^6$	$\lambda = 10^6$
	–	–	–	–	$\beta = 100$	$\beta = 100$	$\beta = 500$	$\beta = 500$

**Table 6.** Classification accuracies for the two AVIRIS datasets using different classification methods. For both the pixel-wise classification and spatial-spectral classification, the best results are highlighted in bold, and the second best results are underlined. AA: average accuracy; KA: kappa coefficient of agreement; OA: overall accuracy.

	Pixel-Wise Classification					Spatial-Spectral Classification					
	KSRC	KCRC	KNLS	KFCLS		JRM		CJRM		PRM	CPRM
				<i>dist</i>	<i>prob</i>	<i>dist</i>	<i>prob</i>	<i>dist</i>	<i>prob</i>		
<b>Indian Pines</b>											
C01	<b>56.67</b>	43.73	55.69	55.49	<u>56.47</u>	58.24	59.61	60.59	65.88	<u>77.45</u>	<b>77.84</b>
C02	78.35	73.28	78.55	<b>78.63</b>	<u>78.56</u>	83.37	84.05	84.06	83.66	<u>86.74</u>	<b>86.92</b>
C03	64.31	53.03	<u>64.49</u>	64.36	<b>64.53</b>	71.19	72.85	79.39	<b>89.58</b>	81.10	<u>85.34</u>
C04	52.12	41.40	<u>52.88</u>	52.79	<b>52.88</b>	56.98	60.18	68.06	<u>82.61</u>	71.71	<b>83.24</b>
C05	89.03	86.17	<u>89.17</u>	<b>89.17</b>	88.98	89.85	90.04	90.74	91.61	<u>92.22</u>	<b>92.22</b>
C06	96.46	<b>96.80</b>	96.46	<u>96.49</u>	95.85	97.24	97.08	98.28	<u>99.00</u>	98.91	<b>99.48</b>
C07	<b>74.58</b>	54.58	74.17	<u>74.17</u>	70.42	88.33	87.92	87.08	88.33	<b>92.92</b>	<u>90.00</u>
C08	98.75	<b>99.27</b>	98.86	<u>98.86</u>	98.47	99.18	99.09	99.38	99.44	<u>99.59</u>	<b>100</b>
C09	57.78	36.11	57.78	<u>57.78</u>	<b>61.67</b>	67.78	<u>76.11</u>	62.78	<b>78.33</b>	59.44	36.67
C10	72.87	61.60	<b>73.33</b>	72.96	<u>73.04</u>	77.60	79.34	<u>88.18</u>	<b>90.02</b>	87.66	88.07
C11	82.42	<b>89.78</b>	82.53	<u>83.16</u>	82.43	88.70	88.38	92.76	<u>95.68</u>	93.97	<b>96.39</b>
C12	76.76	65.04	78.16	<u>78.37</u>	<b>79.50</b>	85.51	88.64	92.26	<u>97.26</u>	94.34	<b>97.58</b>
C13	98.76	98.86	<u>98.96</u>	<b>99.00</b>	98.51	99.00	99.05	99.25	99.30	<u>99.50</u>	<b>99.80</b>
C14	95.28	<b>97.34</b>	<u>95.68</u>	95.66	95.29	97.27	97.05	98.13	<u>98.62</u>	98.28	<b>98.70</b>
C15	53.80	41.94	<u>53.91</u>	53.68	<b>57.40</b>	52.96	56.37	62.88	<u>72.27</u>	71.33	<b>81.30</b>
C16	<b>88.56</b>	79.89	<u>88.44</u>	87.33	68.78	<u>86.44</u>	79.44	<b>86.56</b>	80.89	86.22	85.00
OA(%)	81.34	78.99	<u>81.61</u>	<b>81.72</b>	81.46	85.53	86.15	89.49	<u>92.26</u>	91.00	<b>92.86</b>
AA(%)	77.28	69.93	<b>77.44</b>	<u>77.37</u>	76.42	81.23	82.20	84.40	<b>88.28</b>	86.96	<u>87.41</u>
KA(%)	78.66	75.69	<u>78.98</u>	<b>79.09</b>	78.80	83.44	84.17	88.00	<u>91.17</u>	89.72	<b>91.84</b>
Time(s)	<u>9.28</u>	<b>0.97</b>	15.89	19.50	19.19	47.26	46.95	<u>14.48</u>	<b>14.28</b>	20.08	19.28
<b>Kennedy Space Center</b>											
C01	95.69	<b>97.09</b>	95.80	95.82	95.79	97.58	97.62	99.47	<u>99.92</u>	99.40	<b>99.96</b>
C02	85.09	85.04	<u>85.22</u>	<b>85.35</b>	84.87	85.74	85.78	<b>93.35</b>	<u>93.09</u>	89.96	91.09
C03	<u>90.62</u>	<b>91.48</b>	90.08	90.08	90.29	97.16	97.20	98.11	98.19	<u>98.19</u>	<b>98.93</b>
C04	<b>46.78</b>	41.88	46.15	46.40	<u>46.65</u>	50.92	50.63	52.80	54.60	<b>65.10</b>	<u>64.27</u>
C05	61.64	59.14	61.64	<u>61.71</u>	<b>62.30</b>	70.66	70.99	<b>82.89</b>	<u>82.50</u>	79.87	80.53
C06	<b>45.58</b>	34.10	44.70	44.61	<u>45.16</u>	36.27	36.36	<u>62.30</u>	<b>65.58</b>	61.80	61.38
C07	83.54	<u>84.34</u>	83.94	83.84	<b>84.44</b>	90.10	90.40	97.58	<b>98.59</b>	96.46	<u>97.88</u>
C08	<b>89.29</b>	88.19	88.90	88.88	<u>89.07</u>	93.81	93.94	97.87	<u>99.07</u>	98.34	<b>99.27</b>
C09	96.01	<b>96.66</b>	<u>96.32</u>	96.30	96.28	97.96	97.96	<u>98.34</u>	98.30	<b>98.34</b>	98.30
C10	95.30	94.15	<u>95.30</u>	<b>95.33</b>	93.24	96.66	96.58	98.64	<u>99.27</u>	98.80	<b>99.48</b>
C11	94.30	<b>95.00</b>	94.60	94.60	<u>94.92</u>	95.38	95.15	97.09	<u>97.94</u>	97.49	<b>99.45</b>
C12	<u>88.05</u>	87.42	87.34	87.25	<b>88.28</b>	87.25	87.48	92.01	<u>94.03</u>	93.52	<b>94.30</b>
C13	<b>99.84</b>	<u>99.75</u>	99.69	99.72	99.34	99.89	99.90	<b>100</b>	<b>100</b>	<b>100</b>	<b>100</b>
OA(%)	<b>88.45</b>	87.76	88.32	<u>88.33</u>	88.29	89.97	90.00	93.56	<u>94.26</u>	94.08	<b>94.60</b>
AA(%)	<b>82.44</b>	81.10	82.28	<u>82.30</u>	<u>82.36</u>	84.57	84.61	90.04	<u>90.85</u>	90.56	<b>91.14</b>
KA(%)	<b>87.12</b>	86.31	86.96	<u>86.98</u>	86.94	88.80	88.83	92.81	<u>93.59</u>	93.39	<b>93.97</b>
Time(s)	<u>65.67</u>	<b>6.06</b>	117.8	142.0	140.3	965.5	963.1	<b>113.4</b>	<u>112.2</u>	156.1	155.3

**Table 7.** Classification accuracies for the two ROSIS datasets using different classification methods. For both the pixel-wise classification and spatial-spectral classification, the best results are highlighted in bold, and the second best results are underlined.

	Pixel-Wise Classification					Spatial-Spectral Classification					
	KSRC	KCRC	KNLS	KFCLS		JRM		CJRM		PRM	CPRM
				<i>dist</i>	<i>prob</i>	<i>dist</i>	<i>prob</i>	<i>dist</i>	<i>prob</i>		
<b>University of Pavia</b>											
C1	73.25	72.78	74.77	<u>74.77</u>	<b>75.22</b>	80.03	80.29	94.73	<u>96.19</u>	95.49	<b>98.16</b>
C2	83.24	82.61	83.55	<u>83.57</u>	<b>83.67</b>	87.70	88.32	96.64	<u>98.09</u>	96.55	<b>98.12</b>
C3	78.68	77.64	<b>78.80</b>	<u>78.77</u>	78.52	84.72	84.63	86.77	87.08	<u>92.32</u>	<b>92.51</b>
C4	91.65	<b>93.75</b>	<u>92.68</u>	92.61	91.23	<u>95.00</u>	94.70	94.87	94.86	<b>95.40</b>	93.37
C5	<b>99.42</b>	99.23	99.33	<u>99.34</u>	98.27	99.24	98.67	<u>99.28</u>	98.36	98.94	<b>99.90</b>
C6	84.77	<b>85.64</b>	<u>85.55</u>	85.48	84.72	93.04	93.17	98.78	<u>99.55</u>	99.12	<b>99.91</b>
C7	<u>92.65</u>	<b>94.17</b>	92.54	92.36	92.29	96.75	96.78	99.59	<b>99.87</b>	99.51	<u>99.72</u>
C8	77.99	<b>80.03</b>	<u>78.69</u>	78.57	78.37	87.93	87.82	95.88	<b>97.30</b>	<u>96.44</u>	<u>96.14</u>
C9	<u>99.27</u>	99.22	99.25	<b>99.28</b>	98.80	99.29	98.92	99.37	<u>99.43</u>	99.23	<b>99.54</b>
OA(%)	82.96	83.05	<b>83.57</b>	<u>83.55</u>	83.39	88.45	88.71	96.13	<u>97.19</u>	96.60	<b>97.65</b>
AA(%)	86.77	<u>87.23</u>	<b>87.24</b>	87.20	86.79	91.52	91.48	96.21	<u>96.75</u>	<u>97.00</u>	<b>97.48</b>
KA(%)	78.21	<u>78.37</u>	<b>78.97</b>	<u>78.94</u>	78.72	85.16	85.47	94.94	<u>96.32</u>	95.57	<b>96.92</b>
Time(s)	<u>59.59</u>	<b>5.74</b>	101.7	119.1	117.2	344.1	342.1	<u>75.84</u>	<b>74.86</b>	124.5	119.0
<b>Center of Pavia</b>											
C1	99.67	<b>99.73</b>	99.69	99.69	<u>99.70</u>	99.86	99.90	<b>100</b>	<b>100</b>	<b>100</b>	<b>100</b>
C2	91.09	<b>91.33</b>	91.16	<u>91.18</u>	91.07	91.71	91.67	<u>97.66</u>	<b>97.77</b>	95.86	96.62
C3	<u>89.71</u>	<b>90.43</b>	89.60	<u>89.59</u>	89.25	<u>90.93</u>	90.10	83.58	81.81	<b>91.21</b>	88.72
C4	<b>89.40</b>	88.84	88.99	<u>89.00</u>	88.72	90.85	90.78	99.38	99.09	<u>99.39</u>	<b>99.52</b>
C5	87.95	<u>89.43</u>	89.16	89.07	<b>89.82</b>	93.21	93.39	<u>96.46</u>	<b>97.08</b>	95.54	96.23
C6	96.84	<b>97.62</b>	96.93	96.94	<u>97.01</u>	97.95	98.01	99.55	99.50	<u>99.70</u>	<b>99.83</b>
C7	86.29	85.52	<u>86.57</u>	<b>86.57</b>	86.23	88.78	88.64	<b>94.53</b>	<u>94.46</u>	93.25	93.44
C8	98.83	98.76	<u>98.93</u>	<b>98.94</b>	98.83	99.22	99.15	96.94	<u>96.66</u>	<u>99.52</u>	<b>99.75</b>
C9	<b>99.98</b>	99.97	99.98	<u>99.98</u>	99.97	<b>100</b>	<u>99.99</u>	95.96	95.20	94.04	93.69
OA(%)	96.74	<b>96.89</b>	96.84	96.84	<u>96.85</u>	97.56	97.56	98.56	98.52	<u>98.57</u>	<b>98.61</b>
AA(%)	93.31	<b>93.51</b>	<u>93.44</u>	93.44	<u>93.40</u>	94.72	94.63	96.01	95.73	<b>96.50</b>	<u>96.42</u>
KA(%)	94.40	<b>94.66</b>	94.58	94.57	<u>94.60</u>	95.80	95.80	97.52	97.45	<u>97.53</u>	<b>97.61</b>
Time(s)	<u>75.93</u>	<b>5.81</b>	91.89	109.3	106.9	462.3	460.6	<u>70.98</u>	<b>69.86</b>	116.8	110.7

#### 4.3. Analysis of Parameters

In the first set of experiments, we investigated the impact of the input parameters on KFCLS. Apart from the ADMM parameter  $\mu$  that is empirically set to  $10^{-4}$ , KFCLS has only one parameter  $\gamma$ , which is used for the RBF kernel. Figure 11 shows the impact of  $\gamma$  on the four given datasets, where  $\gamma$  is varied from  $2^{-9}$  to  $2^7$ . It can be observed that for all four given datasets, there is a wide optimal range for the choice of  $\gamma$ . When  $\gamma$  is small, the classification rule “*prob*” is more robust than the classification rule “*dist*” in most cases, and the difference between them is unapparent when  $\gamma$  is large.

In the next set of experiments, we investigated the impact of the input parameters on CJRM, PRM, and CPRM. Notably, JRM is dropped owing to its low accuracy and heavy computational cost. Apart from the parameters  $\mu$  and  $\gamma$  that are empirically set to the same values as those used in KFCLS, there are two parameters needing to be tuned. One is the balance parameter  $\lambda$  used in (24)–(26), which is varied in the range  $[10^{-4}, 10^2]$  for CJRM and  $[10^{-3}, 10^9]$  for PRM and CPRM; and the other is the weight parameter  $\beta$  used in (22), which is varied in the range  $[5, 500]$  for CJRM and  $[100, 1000]$  for PRM and CPRM. Figure 12 shows the classification accuracies for CJRM when applied to the four given datasets. It can be observed that the tuning of  $\lambda$  should synchronize with that of  $\beta$ , the optimal parameters of CJRM-*dist* are almost the same as those of CJRM-*prob*, and it is not difficult for us to get a good result in all cases, since there is a wide range for us to choose a suboptimal combination

of parameters. Figure 13 shows the classification accuracies for PRM and CPRM when applied to the four given datasets. It is evident that the optimal value of  $\lambda$  is  $10^6$  in all cases. Notably, a small positive constant  $\epsilon = 10^{-6}$  is used in (22), and the majority of  $W_{ij}$  will be very small if  $\beta$  is relatively large. In order to connect all the spatially adjacent pixels, it is preferable to fix  $\lambda = 1/\epsilon$ . As for the parameter  $\beta$ , we can choose it in a wide range.

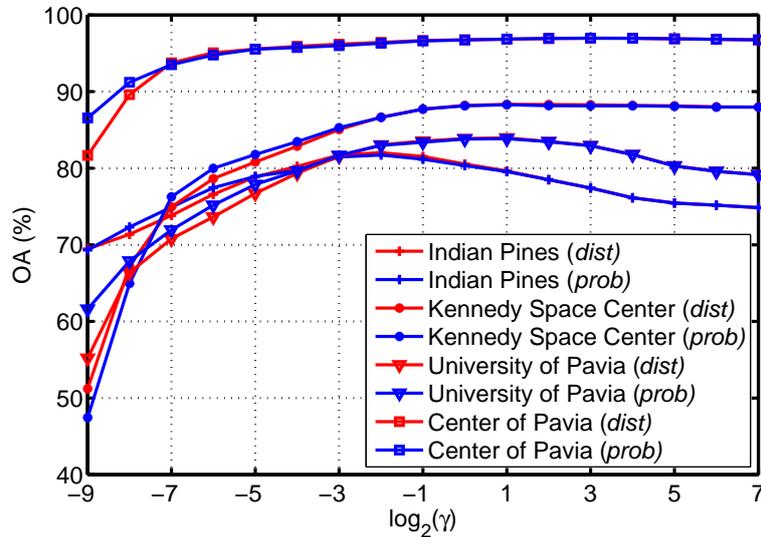


Figure 11. OA as a function of  $\gamma$  for KFCLS when applied to the four given datasets.

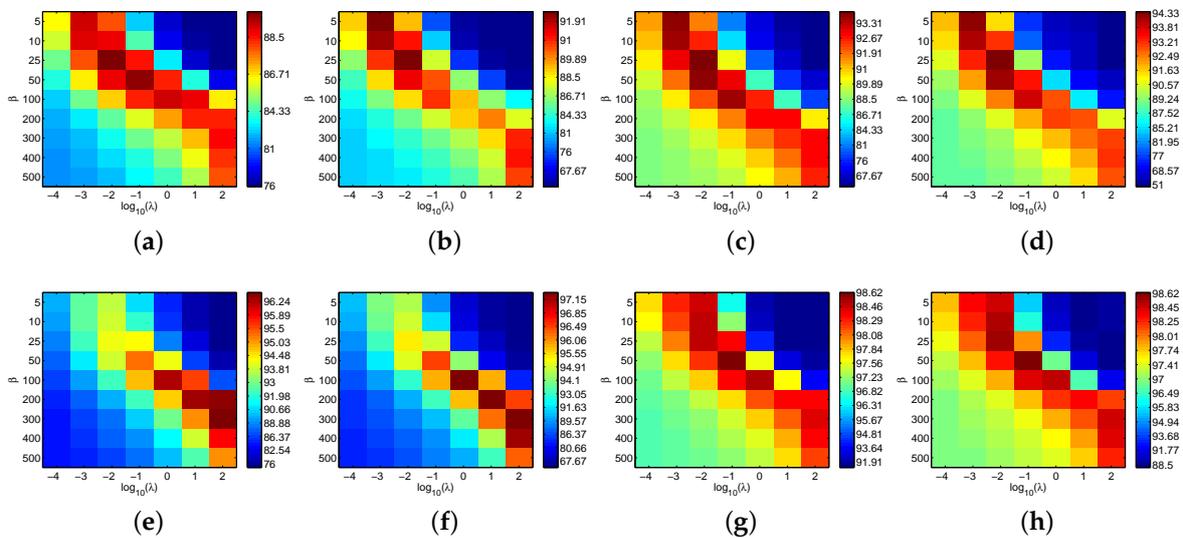
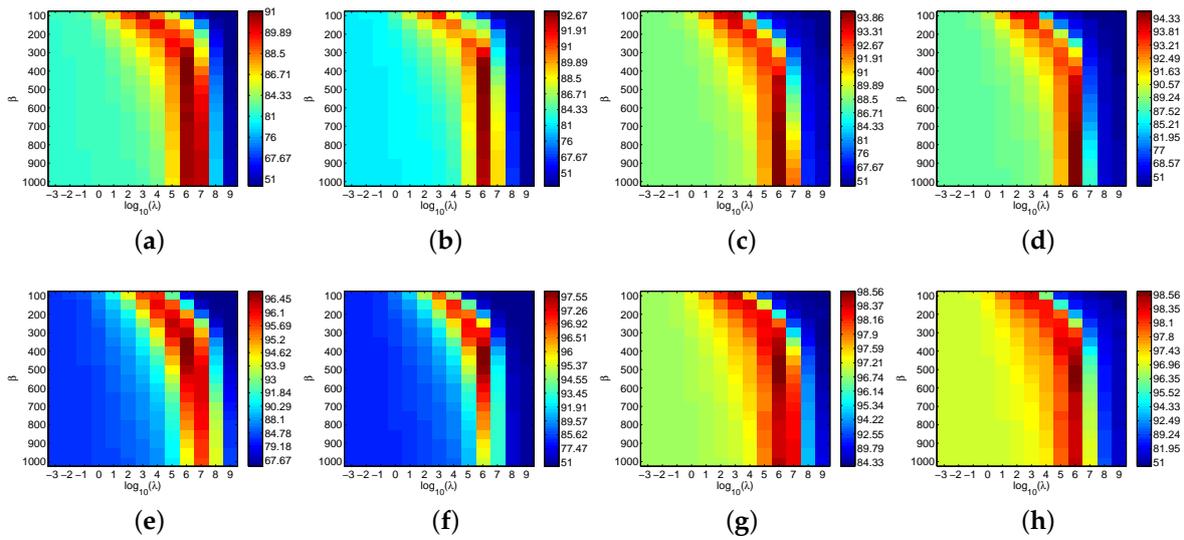


Figure 12. OA with respect to  $\lambda$  and  $\beta$  for CJRM when applied to the four given datasets. (a) CJRM-dist (Indian Pines), (b) CJRM-prob (Indian Pines), (c) CJRM-dist (Kennedy Space Center), (d) CJRM-prob (Kennedy Space Center), (e) CJRM-dist (University of Pavia), (f) CJRM-prob (University of Pavia), (g) CJRM-dist (Center of Pavia), (h) CJRM-prob (Center of Pavia).



**Figure 13.** OA with respect to  $\lambda$  and  $\beta$  for PRM and CPRM when applied to the four given datasets. (a) PRM (Indian Pines), (b) CPRM (Indian Pines), (c) PRM (Kennedy Space Center), (d) CPRM (Kennedy Space Center), (e) PRM (University of Pavia), (f) CPRM (University of Pavia), (g) PRM (Center of Pavia), (h) CPRM (Center of Pavia).

#### 4.4. Impact of the Number of Training Pixels

In this set of experiments, we evaluated the eleven classification methods compared in Section 4.2 in an ill-posed scenario, where different numbers of training pixels are considered. The parameters of these methods are fixed to be the same as those used in Section 4.2. For the two AVIRIS datasets, different percentages of the labeled pixels per class, varied in the range [1%, 20%], were randomly chosen for training, where a minimum of two training pixels per class were taken for very small classes. For the two ROSIS datasets, different numbers of training pixels per class were randomly chosen to build the training sets. Specifically, for University of Pavia, the number was varied to be 10, 20, 40, 60, 80, and 100, and for Center of Pavia, the number was varied to be 5, 10, 20, 40, 60, and 80. Table 8 presents the classification results of the compared methods for the two AVIRIS datasets. It can be observed that for all the compared methods, the OAs increase monotonically as the number of training pixels increases. For the pixel-wise classification, there are no significant gaps between the five methods, and the two investigated classification rules are suitable for the proposed KFCLS. For the spatial-spectral classification, CPRM performs the best, and JRM performs the worst.

Table 9 presents the classification results for the two ROSIS datasets. From this table, we can conclude almost the same results as Table 8. It is evident that the OAs increase monotonically as the number of training pixels increases. For the pixel-wise classification, the proposed KNLS and KFCLS get competitive results compared with KSRC and KCRC. For the spatial-spectral classification, CPRM consistently yields better results than the other five methods, and the improvement of JRM over KFCLS is not significant. Among the three methods CJRM-*dist*, CJRM-*prob*, and PRM, CJRM-*prob* outperformed the others in most cases when applied to the University of Pavia dataset, and they obtained almost the same results when applied to the Center of Pavia dataset.

**Table 8.** OA as a function of the number of training pixels per class for different classification methods when applied to the two AVIRIS datasets. The standard deviation (in parentheses) of the ten random tests is also reported in each case. For both the pixel-wise classification and spatial-spectral classification, the best results are highlighted in bold, and the second-best results are underlined.

	Pixel-Wise Classification					Spatial-Spectral Classification					
	KSRC	KCRC	KNLS	KFCLS		JRM		CJRM		PRM	CPRM
				<i>dist</i>	<i>prob</i>	<i>dist</i>	<i>prob</i>	<i>dist</i>	<i>prob</i>		
<b>Indian Pines</b>											
1%	<b>66.47</b> (1.83)	66.12 (1.83)	<u>66.39</u> (1.78)	66.36 (1.80)	65.83 (1.95)	69.96 (2.18)	70.53 (2.04)	72.28 (2.36)	74.98 (2.73)	<u>76.50</u> (2.82)	<b>79.11</b> (2.80)
3%	77.59 (1.12)	75.67 (1.10)	<b>77.90</b> (1.21)	<u>77.90</u> (1.00)	77.55 (1.00)	82.17 (0.93)	82.89 (0.81)	85.49 (0.69)	<u>89.42</u> (0.71)	87.49 (0.59)	<b>89.74</b> (0.72)
5%	81.34 (0.52)	78.99 (0.75)	<u>81.61</u> (0.56)	<b>81.72</b> (0.42)	81.46 (0.57)	85.53 (0.36)	86.15 (0.33)	89.50 (0.81)	<u>92.25</u> (1.34)	91.00 (0.79)	<b>92.86</b> (1.18)
10%	85.92 (0.59)	83.25 (0.63)	<u>86.22</u> (0.57)	<b>86.23</b> (0.58)	85.93 (0.51)	90.40 (0.64)	90.98 (0.62)	93.27 (0.77)	93.85 (0.84)	<u>94.37</u> (0.74)	<b>95.75</b> (0.76)
15%	88.38 (0.28)	85.98 (0.20)	<b>88.69</b> (0.30)	<u>88.68</u> (0.31)	88.40 (0.41)	92.47 (0.60)	92.88 (0.58)	95.54 (0.48)	<u>96.75</u> (0.37)	95.98 (0.53)	<b>96.82</b> (0.31)
20%	89.41 (0.41)	86.81 (0.46)	<u>89.70</u> (0.43)	<b>89.70</b> (0.41)	89.41 (0.36)	93.17 (0.44)	93.50 (0.49)	96.05 (0.26)	<u>96.34</u> (0.45)	96.29 (0.28)	<b>97.10</b> (0.21)
<b>Kennedy Space Center</b>											
1%	<b>80.26</b> (2.03)	<u>80.24</u> (2.18)	80.16 (2.09)	80.10 (2.09)	79.97 (2.30)	82.37 (2.32)	82.49 (2.43)	84.71 (2.54)	85.19 (2.76)	<u>85.94</u> (2.40)	<b>87.02</b> (2.67)
3%	86.61 (0.89)	86.04 (1.09)	<b>86.72</b> (0.78)	<u>86.69</u> (0.78)	86.69 (0.97)	88.31 (1.50)	88.58 (1.53)	91.79 (1.56)	92.29 (1.80)	<u>92.47</u> (1.69)	<b>92.86</b> (1.71)
5%	<b>88.45</b> (0.93)	87.76 (0.85)	88.32 (1.02)	<u>88.33</u> (1.02)	88.29 (0.97)	89.97 (1.52)	90.00 (1.56)	93.56 (2.08)	<u>94.26</u> (2.17)	94.08 (1.83)	<b>94.60</b> (1.86)
10%	91.00 (0.57)	89.88 (0.33)	91.12 (0.50)	<u>91.13</u> (0.51)	<b>91.13</b> (0.53)	92.93 (0.57)	93.07 (0.56)	96.70 (1.42)	97.16 (1.48)	<u>97.90</u> (0.60)	<b>98.34</b> (0.69)
15%	91.98 (0.53)	90.63 (0.41)	<u>92.15</u> (0.45)	<b>92.15</b> (0.44)	92.12 (0.44)	94.31 (0.56)	94.40 (0.57)	98.28 (0.59)	<u>98.60</u> (0.56)	98.49 (0.47)	<b>98.86</b> (0.48)
20%	92.72 (0.40)	91.21 (0.44)	<b>92.87</b> (0.52)	92.84 (0.50)	<u>92.87</u> (0.47)	95.09 (0.62)	95.18 (0.58)	99.23 (0.33)	<b>99.45</b> (0.30)	99.18 (0.26)	<u>99.42</u> (0.24)

**Table 9.** OA as a function of the number of training pixels per class for different classification methods when applied to the two ROSIS datasets. The standard deviation (in parentheses) of the ten random tests is also reported in each case. For both the pixel-wise classification and spatial-spectral classification, the best results are highlighted in bold, and the second-best results are underlined.

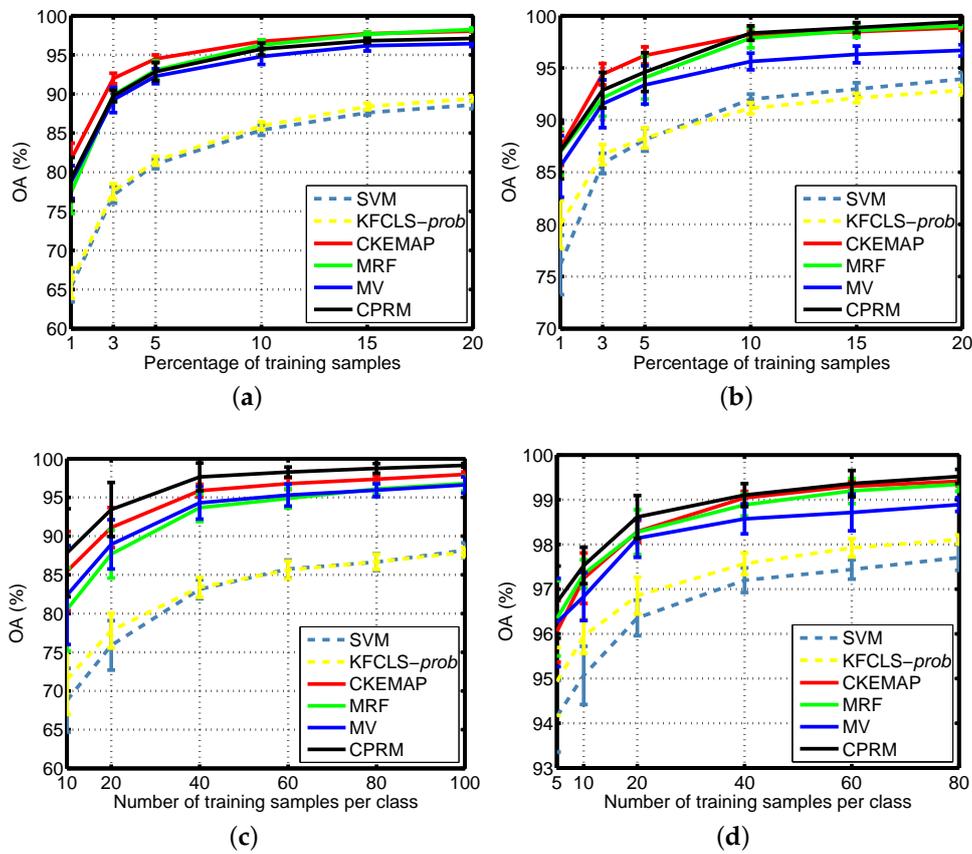
	Pixel-Wise Classification					Spatial-Spectral Classification					
	KSRC	KCRC	KNLS	KFCLS		JRM		CJRM		PRM	CPRM
				<i>dist</i>	<i>prob</i>	<i>dist</i>	<i>prob</i>	<i>dist</i>	<i>prob</i>		
<b>University of Pavia</b>											
10	71.50 (4.17)	72.33 (4.21)	<b>72.44</b> (4.39)	<u>72.36</u> (4.48)	71.52 (4.56)	76.57 (5.41)	76.62 (5.39)	84.38 (5.48)	<u>85.62</u> (5.96)	85.59 (6.91)	<b>87.85</b> (5.69)
20	77.50 (2.33)	77.81 (2.14)	<b>78.14</b> (2.24)	<u>78.13</u> (2.25)	77.78 (2.24)	83.11 (2.92)	83.21 (2.90)	91.30 (2.71)	<u>92.45</u> (3.61)	91.97 (3.44)	<b>93.44</b> (3.49)
40	82.96 (1.05)	83.05 (1.14)	<b>83.57</b> (1.17)	<u>83.55</u> (1.19)	83.39 (1.30)	88.45 (1.84)	88.71 (1.92)	96.13 (1.64)	<u>97.19</u> (1.47)	96.60 (2.07)	<b>97.65</b> (1.80)
60	85.42 (1.26)	85.43 (1.43)	<b>85.78</b> (1.23)	<u>85.78</u> (1.25)	85.63 (1.15)	90.80 (1.39)	90.93 (1.40)	96.63 (1.02)	<u>97.38</u> (1.03)	97.16 (0.76)	<b>98.27</b> (0.64)
80	86.38 (0.75)	86.26 (0.97)	<b>86.78</b> (0.91)	<u>86.77</u> (0.91)	86.68 (0.99)	91.33 (0.79)	91.37 (0.86)	97.80 (0.83)	<u>98.30</u> (0.74)	98.17 (0.62)	<b>98.74</b> (0.63)
100	87.70 (0.82)	87.54 (0.89)	<b>88.05</b> (0.68)	<u>88.04</u> (0.68)	87.90 (0.61)	92.34 (0.90)	92.41 (0.89)	98.12 (0.56)	98.40 (0.49)	<u>98.59</u> (0.70)	<b>99.14</b> (0.24)
<b>Center of Pavia</b>											
5	94.46 (0.99)	94.71 (0.95)	<u>94.82</u> (0.74)	94.79 (0.75)	<b>94.93</b> (0.76)	95.97 (0.74)	96.14 (0.69)	96.43 (0.74)	<u>96.63</u> (0.71)	96.60 (0.83)	<b>96.71</b> (0.81)
10	95.50 (0.53)	<b>96.03</b> (0.41)	95.89 (0.46)	95.88 (0.47)	<u>95.95</u> (0.38)	96.86 (0.47)	96.93 (0.45)	97.50 (0.31)	97.49 (0.32)	<u>97.52</u> (0.43)	<b>97.53</b> (0.41)
20	96.74 (0.42)	<b>96.89</b> (0.43)	96.84 (0.38)	96.84 (0.38)	<u>96.85</u> (0.41)	97.56 (0.48)	97.56 (0.51)	98.56 (0.44)	98.52 (0.47)	<u>98.57</u> (0.49)	<b>98.61</b> (0.48)
40	97.56 (0.21)	97.55 (0.16)	<u>97.58</u> (0.22)	<b>97.58</b> (0.22)	97.57 (0.24)	98.11 (0.24)	98.11 (0.24)	98.77 (0.26)	98.25 (0.39)	<u>99.06</u> (0.24)	<b>99.10</b> (0.26)
60	97.91 (0.19)	97.80 (0.20)	<u>97.93</u> (0.21)	<b>97.94</b> (0.21)	97.92 (0.20)	98.42 (0.25)	98.41 (0.25)	99.07 (0.30)	98.99 (0.32)	<u>99.33</u> (0.30)	<b>99.36</b> (0.29)
80	98.10 (0.12)	97.91 (0.17)	<u>98.12</u> (0.11)	<b>98.12</b> (0.11)	98.11 (0.10)	98.55 (0.17)	98.54 (0.17)	99.22 (0.17)	99.11 (0.21)	<u>99.49</u> (0.13)	<b>99.52</b> (0.16)

#### 4.5. Comparison to Other Classification Techniques

In this set of experiments, CPRM is compared with three other techniques that can incorporate the spatial-spectral information into KFCLS-*prob*. For these methods, the free parameters introduced by KFCLS are set to the same values as those used in KFCLS. The first method is the composite kernel technique [33] using the original spectral features and the extended multiattribute profile (EMAP) features [61,62], termed as CKEMAP, where the EMAP features are extracted from the first three principal components of the hyperspectral image and built by the area and standard deviation attributes as reported in [63]. The additional parameters of CKEMAP are chosen using cross-validation. The second method is the pixel-wise KFCLS-*prob* followed by Markov random fields (MRF) [36,37], where the MRF technique is utilized to incorporate the spatial-spectral information by refining the posterior probabilistic outputs. The free parameters of MRF are chosen using cross-validation. The last method is the pixel-wise KFCLS-*prob* followed by majority voting within superpixel regions [39], termed as MV, where the superpixel segmentation algorithm and its free parameters are chosen with reference to [64,65]. Moreover, two baseline classifiers KFCLS-*prob* and SVM are also included for reference. For SVM, the RBF kernel is used and the free parameters are chosen using cross-validation.

Figure 14 shows the classification accuracies of the six compared methods when different numbers of training pixels are used. It can be observed that CKEMAP performs the best for the two AVIRIS

datasets and CPRM performs the best for the two ROSIS datasets. Among the three post-processing methods (i.e., MRF, MV, and the proposed CPRM), CPRM outperforms the other methods in most cases.



**Figure 14.** OA as a function of the number of training pixels when applied to the four given datasets. (a) AVIRIS Indian Pines dataset, (b) AVIRIS Kennedy Space Center dataset, (c) ROSIS University of Pavia dataset, (d) ROSIS Center of Pavia dataset.

## 5. Conclusions

This paper considers using the nonnegative constraint to achieve the collaborative representation mechanism under SRC and CRC in the kernel space, and thereby proposes KNLS for hyperspectral image classification by replacing  $\ell_1$ -norm or  $\ell_2$ -norm with the nonnegative constraint. In order to provide the posterior probabilistic outputs, we propose KFCLS by enforcing the summation of the nonnegative coding coefficients of each pixel to be one, and subsequently introduce two classification rules to determine the class labels. Compared with KSRC and KCRC, KFCLS can get competitive results and its coding coefficients are more meaningful and useful for the subsequent processing steps. Furthermore, in order to incorporate the spatial-spectral information into KFCLS using regularization technique, we investigated the co-processing and post-processing methods by applying coefficient-level and class-level regularization strategies. Experimental results conducted on four real hyperspectral images have demonstrated: (1) the proposed KFCLS can get competitive results comparing with the other pixel-wise classifiers; (2) the proposed classification rule “*prob*” is effective; (3) the class-level regularization strategy is better than the coefficient-level regularization strategy; and (4) CPRM is an effective and efficient post-processing method and the most efficient method among the investigated four kinds of methods. In future work, we expect that the suggested regularization method can facilitate the development of spectral unmixing.

**Acknowledgments:** This work was supported by the National Natural Science Foundation of China (Grant numbers 61601201, 61772274, and 61471199), the Natural Science Foundation of Jiangsu Province (Grant numbers BK20160188 and BK20150160) and the Fundamental Research Funds for the Central Universities (Grant number 30917015104 and JUSRP51635B). The authors would like to thank D. Landgrebe from Purdue University for providing the AVIRIS Indian Pines dataset and P. Gamba from the University of Pavia, Italy, for providing the ROSIS University of Pavia dataset and Center of Pavia dataset.

**Author Contributions:** Jianjun Liu and Zebin Wu wrote the paper. Jianjun Liu, Zebin Wu and Jinlong Yang analyzed the data. Jianjun Liu and Zhiyong Xiao conceived and designed the experiments. Jianjun Liu performed the experiments. Jianjun Liu contributed analysis tools.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Wright, J.; Yang, A.Y.; Ganesh, A.; Sastry, S.S.; Ma, Y. Robust face recognition via sparse representation. *IEEE Trans. Pattern Anal. Mach. Intell.* **2009**, *31*, 210–227.
2. Yang, J.; Zhang, L.; Xu, Y.; Yang, J.Y. Beyond sparsity: The role of L1-optimizer in pattern classification. *Pattern Recognit.* **2012**, *45*, 1104–1118.
3. Mei, X.; Ling, H. Robust visual tracking and vehicle classification via sparse representation. *IEEE Trans. Pattern Anal. Mach. Intell.* **2011**, *33*, 2259–2272.
4. Yuan, X.T.; Liu, X.; Yan, S. Visual classification with multitask joint sparse representation. *IEEE Trans. Image Process.* **2012**, *21*, 4349–4360.
5. Chen, Y.; Nasrabadi, N.M.; Tran, T.D. Hyperspectral image classification using dictionary-based sparse representation. *IEEE Trans. Geosci. Remote Sens.* **2011**, *49*, 3973–3985.
6. Song, B.; Li, J.; Dalla Mura, M.; Li, P.; Plaza, A.; Bioucas-Dias, J.M.; Benediktsson, J.A.; Chanussot, J. Remotely sensed image classification using sparse representations of morphological attribute profiles. *IEEE Trans. Geosci. Remote Sens.* **2014**, *52*, 5122–5136.
7. Xue, Z.; Du, P.; Su, H.; Zhou, S. Discriminative sparse representation for hyperspectral image classification: A semi-supervised perspective. *Remote Sens.* **2017**, *9*, 386.
8. Bian, X.; Chen, C.; Xu, Y.; Du, Q. Robust hyperspectral image classification by multi-layer spatial-spectral sparse representations. *Remote Sens.* **2016**, *8*, 985.
9. He, Z.; Li, J.; Liu, L. Tensor block-sparsity based representation for spectral-spatial hyperspectral image classification. *Remote Sens.* **2016**, *8*, 636.
10. Xu, Y.; Wu, Z.; Li, J.; Plaza, A.; Wei, Z. Anomaly detection in hyperspectral images based on low-rank and sparse representation. *IEEE Trans. Geosci. Remote Sens.* **2016**, *54*, 1990–2000.
11. Liu, J.; Xiao, Z.; Chen, Y.; Yang, J. Spatial-spectral graph regularized kernel sparse representation for hyperspectral image classification. *ISPRS Int. J. Geo-Inf.* **2017**, *6*, 258.
12. Melgani, F.; Bruzzone, L. Classification of hyperspectral remote sensing images with support vector machines. *IEEE Trans. Geosci. Remote Sens.* **2004**, *42*, 1778–1790.
13. Li, J.; Bioucas-Dias, J.M.; Plaza, A. Semisupervised hyperspectral image segmentation using multinomial logistic regression with active learning. *IEEE Trans. Geosci. Remote Sens.* **2010**, *48*, 4085–4098.
14. Lei, Z.; Meng, Y. Sparse representation or collaborative representation: Which helps face recognition? In Proceedings of the IEEE International Conference on Computer Vision, Barcelona, Spain, 6–13 November 2011; pp. 471–478.
15. Zhang, L.; Yang, M.; Feng, X.; Ma, Y.; Zhang, D. Collaborative representation based classification for face recognition. *arXiv* **2012**, arXiv:1204.2358.
16. Yang, M.; Zhang, L.; Zhang, D.; Wang, S. Relaxed collaborative representation for pattern classification. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Providence, RI, USA, 16–21 June 2012; pp. 2224–2231.
17. Li, W.; Du, Q. Joint within-class collaborative representation for hyperspectral image classification. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2014**, *7*, 2200–2208.
18. Li, J.; Zhang, H.; Huang, Y.; Zhang, L. Hyperspectral image classification by nonlocal joint collaborative representation with a locally adaptive dictionary. *IEEE Trans. Geosci. Remote Sens.* **2014**, *52*, 3707–3719.
19. Camps-Valls, G.; Bruzzone, L. Kernel-based methods for hyperspectral image classification. *IEEE Trans. Geosci. Remote Sens.* **2005**, *43*, 1351–1362.

20. Gao, S.; Tsang, I.W.H.; Chia, L.T. Kernel sparse representation for image classification and face recognition. In *European Conference on Computer Vision*; Springer: Berlin/Heidelberg, Germany, 2010; pp. 1–14.
21. Chen, Y.; Nasrabadi, N.M.; Tran, T.D. Hyperspectral image classification via kernel sparse representation. *IEEE Trans. Geosci. Remote Sens.* **2013**, *51*, 217–231.
22. Wang, D.; Lu, H.; Yang, M.H. Kernel collaborative face recognition. *Pattern Recognit.* **2015**, *48*, 3025–3037.
23. Liu, J.; Wu, Z.; Li, J.; Plaza, A.; Yuan, Y. Probabilistic-Kernel collaborative representation for spatial-spectral hyperspectral image classification. *IEEE Trans. Geosci. Remote Sens.* **2016**, *54*, 2371–2384.
24. Li, J.; Zhang, H.; Zhang, L. Column-generation kernel nonlocal joint collaborative representation for hyperspectral image classification. *ISPRS J. Photogramm. Remote Sens.* **2014**, *94*, 25–36.
25. Li, W.; Du, Q.; Xiong, M. Kernel collaborative representation with Tikhonov regularization for hyperspectral image classification. *IEEE Geosci. Remote Sens. Lett.* **2015**, *12*, 48–52.
26. Lee, D.D.; Seung, H.S. Algorithms for non-negative matrix factorization. *Adv. Neural Inf. Process. Syst.* **2000**, *13*, 556–562.
27. Iordache, M.D.; Bioucas-Dias, J.M.; Plaza, A. Sparse unmixing of hyperspectral data. *IEEE Trans. Geosci. Remote Sens.* **2011**, *49*, 2014–2039.
28. Foucart, S.; Koslicki, D. Sparse recovery by means of nonnegative least squares. *IEEE Signal Process. Lett.* **2014**, *21*, 498–502.
29. Slawski, M.; Hein, M. Sparse recovery by thresholded non-negative least squares. *Adv. Neural Inf. Process. Syst.* **2011**, *24*, 1926–1934.
30. Heinz, D.C. Fully constrained least squares linear spectral mixture analysis method for material quantification in hyperspectral imagery. *IEEE Trans. Geosci. Remote Sens.* **2001**, *39*, 529–545.
31. Broadwater, J.; Chellappa, R.; Banerjee, A.; Burlina, P. Kernel fully constrained least squares abundance estimates. In *Proceedings of the IEEE International Geoscience and Remote Sensing Symposium, Barcelona, Spain, 23–28 July 2007*; pp. 4041–4044.
32. Fauvel, M.; Tarabalka, Y.; Benediktsson, J.A.; Chanussot, J.; Tilton, J.C. Advances in spectral-spatial classification of hyperspectral images. *Proc. IEEE* **2013**, *101*, 652–675.
33. Camps-Valls, G.; Gomez-Chova, L.; Muñoz-Marí, J.; Vila-Francis, J.; Calpe-Maravilla, J. Composite kernels for hyperspectral image classification. *IEEE Geosci. Remote Sens. Lett.* **2006**, *3*, 93–97.
34. Li, J.; Huang, X.; Gamba, P.; Bioucas-Dias, J.M.B.; Zhang, L.; Benediktsson, J.A.; Plaza, A. Multiple feature learning for hyperspectral image classification. *IEEE Trans. Geosci. Remote Sens.* **2015**, *53*, 1592–1606.
35. Liang, H.; Li, Q. Hyperspectral imagery classification using sparse representations of convolutional neural network features. *Remote Sens.* **2016**, *8*, 99.
36. Tarabalka, Y.; Fauvel, M.; Chanussot, J.; Benediktsson, J.A. SVM- and MRF-based method for accurate classification of hyperspectral images. *IEEE Geosci. Remote Sens. Lett.* **2010**, *7*, 736–740.
37. Li, J.; Bioucas-Dias, J.M.; Plaza, A. Hyperspectral image segmentation using a new Bayesian approach with active learning. *IEEE Trans. Geosci. Remote Sens.* **2011**, *49*, 3947–3960.
38. Wu, Z.; Shi, L.; Li, J.; Wang, Q.; Sun, L.; Wei, Z.; Plaza, J.; Plaza, A. GPU parallel implementation of spatially adaptive hyperspectral image classification. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2017**, *PP*, 1–13.
39. Tarabalka, Y.; Chanussot, J.; Benediktsson, J.A. Segmentation and classification of hyperspectral images using watershed transformation. *Pattern Recognit.* **2010**, *43*, 2367–2379.
40. Priya, T.; Prasad, S.; Wu, H. Superpixels for spatially reinforced bayesian classification of hyperspectral images. *IEEE Geosci. Remote Sens. Lett.* **2015**, *12*, 1071–1075.
41. Liu, J.; Wu, Z.; Sun, L.; Wei, Z.; Xiao, L. Hyperspectral image classification using kernel sparse representation and semilocal spatial graph regularization. *IEEE Geosci. Remote Sens. Lett.* **2014**, *11*, 1320–1324.
42. Feng, J.; Cao, Z.; Pi, Y. Polarimetric contextual classification of PolSAR images using sparse representation and superpixels. *Remote Sens.* **2014**, *6*, 7158–7181.
43. Wang, J.; Jiao, L.; Liu, H.; Yang, S.; Liu, F. Hyperspectral image classification by spatial-spectral derivative-aided kernel joint sparse representation. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2015**, *8*, 2485–2500.
44. Zhang, S.; Li, S.; Fu, W.; Fang, L. Multiscale superpixel-based sparse representation for hyperspectral image classification. *Remote Sens.* **2017**, *9*, 139.
45. Tong, F.; Tong, H.; Jiang, J.; Zhang, Y. Multiscale union regions adaptive sparse representation for hyperspectral image classification. *Remote Sens.* **2017**, *9*, 872.

46. Yuan, H.; Tang, Y.Y.; Lu, Y.; Yang, L.; Luo, H. Hyperspectral image classification based on regularized sparse representation. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2014**, *7*, 2174–2182.
47. Aubert, G.; Kornprobst, P. Mathematical problems in image processing: Partial differential equations and calculus of variations. *Appl. Intell.* **2006**, *40*, 291–304.
48. Zhang, X.; Burger, M.; Bresson, X.; Osher, S. Bregmanized nonlocal regularization for deconvolution and sparse reconstruction. *SIAM J. Imaging Sci.* **2010**, *3*, 253–276.
49. Liu, J.; Wu, Z.; Wei, Z.; Xiao, L.; Sun, L. Spatial-spectral kernel sparse representation for hyperspectral image classification. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2013**, *6*, 2462–2471.
50. Yang, J.; Zhang, Y. Alternating direction algorithms for  $\ell_1$ -Problems in compressive sensing. *SIAM J. Sci. Comput.* **2009**, *33*, 250–278.
51. Goldstein, T.; Osher, S. The split bregman method for L1-Regularized problems. *SIAM J. Imaging Sci.* **2009**, *2*, 323–343.
52. Mairal, J.; Bach, F.; Ponce, J. Sparse modeling for image and vision processing. *Found. Trends Comput. Graph. Vis.* **2014**, *8*, 85–283.
53. Bioucas-Dias, J.M.; Figueiredo, M.A. Alternating direction algorithms for constrained sparse regression: Application to hyperspectral unmixing. In Proceedings of the IEEE Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing, Reykjavik, Iceland, 14–16 June 2010; pp. 1–4.
54. Wen, Y.W.; Wang, M.; Cao, Z.; Cheng, X.; Ching, W.K.; Vassiliadis, V.S. Sparse solution of nonnegative least squares problems with applications in the construction of probabilistic Boolean networks. *Numer. Linear Algebra Appl.* **2015**, *22*, 883–899.
55. Combettes, P.L.; Wajs, V.R. Signal recovery by proximal forward-backward splitting. *SIAM J. Multiscale Model. Simul.* **2005**, *4*, 1168–1200.
56. Kang, X.; Li, S.; Fang, L.; Li, M.; Benediktsson, J.A. Extended random walker-based classification of hyperspectral images. *IEEE Trans. Geosci. Remote Sens.* **2015**, *53*, 144–153.
57. Chen, Y.; Lin, Z.; Zhao, X.; Wang, G.; Gu, Y. Deep learning-based classification of hyperspectral data. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2014**, *7*, 2094–2107.
58. Makantasis, K.; Karantzas, K.; Doulamis, A.; Doulamis, N. Deep supervised learning for hyperspectral data classification through convolutional neural networks. In Proceedings of the IEEE International Geoscience and Remote Sensing Symposium, Milan, Italy, 26–31 July 2015; pp. 4959–4962.
59. Hu, W.; Huang, Y.; Wei, L.; Zhang, F.; Li, H. Deep convolutional neural networks for hyperspectral image classification. *J. Sens.* **2015**, *2015*, 1–12.
60. Chen, Y.; Jiang, H.; Li, C.; Jia, X.; Ghamisi, P. Deep feature extraction and classification of hyperspectral images based on convolutional neural networks. *IEEE Trans. Geosci. Remote Sens.* **2016**, *54*, 6232–6251.
61. Dalla Mura, M.; Atli Benediktsson, J.; Waske, B.; Bruzzone, L. Extended profiles with morphological attribute filters for the analysis of hyperspectral data. *Int. J. Remote Sens.* **2010**, *31*, 5975–5991.
62. Mura, M.D.; Benediktsson, J.A.; Waske, B.; Bruzzone, L. Morphological attribute profiles for the analysis of very high resolution images. *IEEE Trans. Geosci. Remote Sens.* **2010**, *48*, 3747–3762.
63. Li, J.; Marpu, P.R.; Plaza, A.; Bioucas-Dias, J.M.; Benediktsson, J.A. Generalized composite kernel framework for hyperspectral image classification. *IEEE Trans. Geosci. Remote Sens.* **2013**, *51*, 4816–4829.
64. Fang, L.; Li, S.; Duan, W.; Ren, J.; Benediktsson, J.A. Classification of hyperspectral images by exploiting spectral-spatial information of superpixel via multiple kernels. *IEEE Trans. Geosci. Remote Sens.* **2015**, *53*, 6663–6674.
65. Liu, J.; Xiao, Z.; Xiao, L. Superpixel-guided multiscale kernel collaborative representation for hyperspectral image classification. *Remote Sens. Lett.* **2016**, *7*, 975–984.

