

Article

Novel Semi-Supervised Hyperspectral Image Classification Based on a Superpixel Graph and Discrete Potential Method

Yifei Zhao ^{1,2,3}, Fenzhen Su ^{1,3,*} and Fengqin Yan ^{1,3}

¹ State Key Laboratory of Resources and Environmental Information System, Institute of Geographical Sciences and Natural Resources Research, Chinese Academy of Sciences, Beijing 100101, China; zhaoyf@reis.ac.cn (Y.Z.); yanfq@reis.ac.cn (F.Y.)

² College of Resources and Environment, University of Chinese Academy of Science, Beijing 100049, China

³ Collaborative Innovation Center of South China Sea Studies, Nanjing University, Nanjing 210093, China

* Correspondence: sufz@reis.ac.cn; Tel.: +86-10-6488-8956

Received: 29 March 2020; Accepted: 8 May 2020; Published: 11 May 2020



Abstract: Hyperspectral image (HSI) classification plays an important role in the automatic interpretation of the remotely sensed data. However, it is a non-trivial task to classify HSI accurately and rapidly due to its characteristics of having a large amount of data and massive noise points. To address this problem, in this work, a novel, semi-supervised, superpixel-level classification method for an HSI was proposed based on a graph and discrete potential (SSC-GDP). The key idea of the proposed scheme is the construction of the weighted connectivity graph and the division of the weighted graph. Based on the superpixel segmentation, a weighted connectivity graph is constructed using the weighted connection between a superpixel and its spatial neighbors. The generated graph is then divided into different communities/sub-graphs by using a discrete potential and the improved semi-supervised Wu–Huberman (ISWH) algorithm. Each community in the weighted connectivity graph represents a class in the HSI. The local connection strategy, together with the linear complexity of the ISWH algorithm, ensures the fast implementation of the suggested SSC-GDP method. To prove the effectiveness of the proposed spectral–spatial method, two public benchmarks, Indian Pines and Salinas, were utilized to test the performance of our proposal. The comparative test results confirmed that the proposed method was superior to several other state-of-the-art methods.

Keywords: hyperspectral image; superpixel; weighted connectivity graph; discrete potential; semi-supervised classification

1. Introduction

With the rapid development of hyperspectral remote sensing, computers, and communication technology, a large number of hyperspectral data containing hundreds of narrow spectral bands and rich spatial information have been collected in the past few decades. Compared with multispectral images, a hyperspectral image (HSI) with detailed spectral information and abundant spatial texture significantly improves its identification ability for land cover and is thus widely applied to various fields, such as ocean exploration, environmental monitoring, urban planning, and others [1–3]. However, it is still a challenging problem regarding how to classify HSI quickly and accurately in the field of remote sensing because of its characteristics of high dimension, a large amount of data, and massive noise pixels.

In the remote sensing literature, feature extraction and feature selection are two groups of effective dimensionality-reduction techniques to cope with a high dimensional HSI. Some feature extraction methods are principal component analysis (PCA) and its various improved versions [4–6], locally linear

embedding [7], linear discriminant analysis [8], and so on. Typical feature selection approaches include techniques based on the optimization algorithm [9,10], as well as those that are geometry-based [11] and clustering-based [12]. As an important preprocessing step for HSI classification, these methods not only effectively improve the speed of the classification algorithm, but also avoid the occurrence of Hughes' phenomenon (the classification accuracy shows a tendency of first rising and then decreasing with the increase of the number of bands). At the same time, the volume of the HSI decreases sharply when reducing the dimensionality of pixels.

To manage the problem of noisy data, the spatial structure information of pixels is considered in the classification process. In the discontinuity-preserving relaxation [13], guided filter [14], and recursive filtering [15] methods, spatial information is used to de-noise the raw HSI while attempting to maintain a class boundary. The Markov random field technique [16,17] adopted a fixed-size window centered on the tested pixel to smooth the data. A large number of experimental results demonstrate that these methods can significantly improve the classification accuracy.

For the problem of the large amount of data of an HSI, it is natural to classify hyperspectral images by using machine learning methods, such as a support vector machine [18,19] and multinomial logistic regression [13], among others. Generally speaking, spectral information alone does not yield satisfactory classification results. As such, spatial information provided by the HSI itself is gradually integrated into classifiers. In multiple kernel technology [20,21], the similarity between two pixels was measured better by combining the spatial information with the spectrum of pixels. Additionally, some spectral-spatial classification methods also incorporated spatial information into other techniques to improve the classification performance, such as sparse representation [22,23] and low-rank representation [24,25].

Among various spectral-spatial classification approaches, superpixel-based HSI classification methods have recently attracted increasingly more attention in remote sensing [26–30]. Superpixels are the homogeneous regions containing a set of spatially adjacent pixels with similar textures and colors. Moreover, the shape and size of a superpixel can be adaptive according to the local structures of the image. Superpixel segmentation algorithms include entropy rate superpixel segmentation (ERS) [31], simple linear iterative clustering (SLIC) [32], seeds [33], watersheds [34], and so on. Superpixel homogeneity implies that the pixels within a superpixel are more likely to belong to the same class. A majority of HSI classifiers based on superpixels make full use of superpixel homogeneity to correct the pre-classification results obtained using the pixel-wise classification algorithms. As a result, the classification accuracy is greatly improved. Recently, based on an affine hull model and singular value decomposition, Lu et al. [35] defined a set-to-set distance and proposed a spectral-spatial classification method for an HSI at the superpixel level. Tu et al. [36] suggested another superpixel-level HSI classification process. The distance between two superpixels was calculated by selecting partial pixels from each superpixel. Very recently, Xie et al. proposed an effective superpixel-level classification method by introducing a new similarity between superpixels and using the k-nearest neighbor (KNN) rule [37]. In the superpixel-level HSI classifiers, each superpixel is regarded as a new sample such that the volume of hyperspectral data can be greatly reduced. It implies that the classification process will likely be completed in a shorter time. Thus, it is worth trying to develop new superpixel-level HSI classification methods in the field of remote sensing.

Graph is a powerful tool for data representation. In remote sensing, the graph-based methods have previously been investigated to classify hyperspectral data [38–42]. Camps-Valls et al. first introduced a semi-supervised, graph-based classification method for an HSI [43]. They adopted different spectral and spatial kernels to construct pixel-based weighted graphs. Gao et al. presented a bi-layer, graph-based learning method to classify an HSI with a limited number of labeled pixels [44]. Cui et al. [45] utilized extended label propagation and a rolling guidance filter to develop a semi-supervised HSI classification method. This method adopted a graph-based label propagation algorithm to predict the pixel labels and modified some mislabeled pixels using superpixels. Taking each superpixel as a node, Sellars et al. [46] proposed a graph-based learning framework for HSI classification. The final

classification was completed using a local and global consistency algorithm. In most of the existing graph-based classification methods, the construction of graphs is at the pixel level, except for the method proposed by Sellars et al. [46]. Generally speaking, the classification result and speed of these graph-based methods depend largely on the construction and partitioning of the graph. Therefore, it is of great importance to put forward new superpixel-based methods for graph construction and fast graph division techniques.

Deep-learning-model-based HSI classification methods have been extensively studied in recent years [47]. Early work on HSI classification using deep learning techniques can be traced back to the deep model based on stacked auto-encoders [48]. Adopting the convolutional kernels automatically learned from HSI, Ding et al. [49] introduced an HSI classification approach using convolutional neural networks. Based on the convolutional neural network and a generative adversarial network, Chen et al. designed a semi-supervised, fine-grained classifier and demonstrated the effectiveness of their method on an Indian Pines dataset [50]. Combined with band selection, Sellami et al. proposed a semi-supervised HSI classification method based on a 3D-convolutional auto-encoder [51].

Graphs are also known as complex networks in computer science and physics. Several facts show that complex networks commonly have a significant feature, namely, a community structure. Although there is no universally accepted formal definition of the community at present, a community is generally considered to be a sub-network in which the nodes in a community are densely linked to each other and the connections between the communities are sparse [52]. For an HSI classification, such a community can be regarded as a class in a hyperspectral dataset. In many community detection methods [53–56], the unsupervised Wu–Huberman algorithm (UWH), based on a discrete potential [55], is a powerful algorithm used to effectively discover communities in complex networks because of its fast performance. However, several assumptions of the UWH limit its application. Therefore, we hereby attempt to improve the UWH to be a semi-supervised method to accomplish the task of community detection in weighted connected networks.

To address the aforementioned problems, this study suggests a semi-supervised superpixel-level HSI classification method based on a graph and discrete potential (SSC-GDP), aiming at classifying HSIs accurately and quickly. In the proposed classification scheme, each superpixel is viewed as a node in a graph. This leads to a significant reduction in the volume of the hyperspectral data to be classified. Subsequently, we link each superpixel to its spatial neighbors to construct a weighted connectivity graph. The weight of each edge represents the affinity of a superpixel and its neighbors. Communities in the constructed weighted connectivity graph are detected using the improved semi-supervised Wu–Huberman algorithm (ISWH). Finally, the HSI classification is completed by mapping each community in the generated graph to a class of an HSI. The sparseness of the generated graph and the linear complexity of the ISWH ensure the fast implementation of the proposed classification scheme. This method is a novel attempt at applying community detection techniques to solve the problem of remote sensing data classification. Experimental results confirm the effectiveness of the proposed classification scheme. In summary, this work makes the following contributions:

- (1) A novel, semi-supervised, superpixel-level classification scheme for HSI is proposed based on a discrete potential technique, which is an effective superpixel graph-based method.
- (2) It is the first time that the discrete potential technique is adopted to classify HSIs.
- (3) The unsupervised Wu–Huberman algorithm is improved to be a semi-supervised method to better detect communities hidden in weighted connectivity graphs.
- (4) Unlike existing superpixel-based methods, the proposed classification algorithm works on the whole HSI image, not just the reference data.

2. Methodology

In this section, we describe the classification framework we propose in detail. It involves three main parts, that is, the superpixel segmentation, the construction of the weighted graph, and the community detection in the generated graph. The proposed method is illustrated in Figure 1.

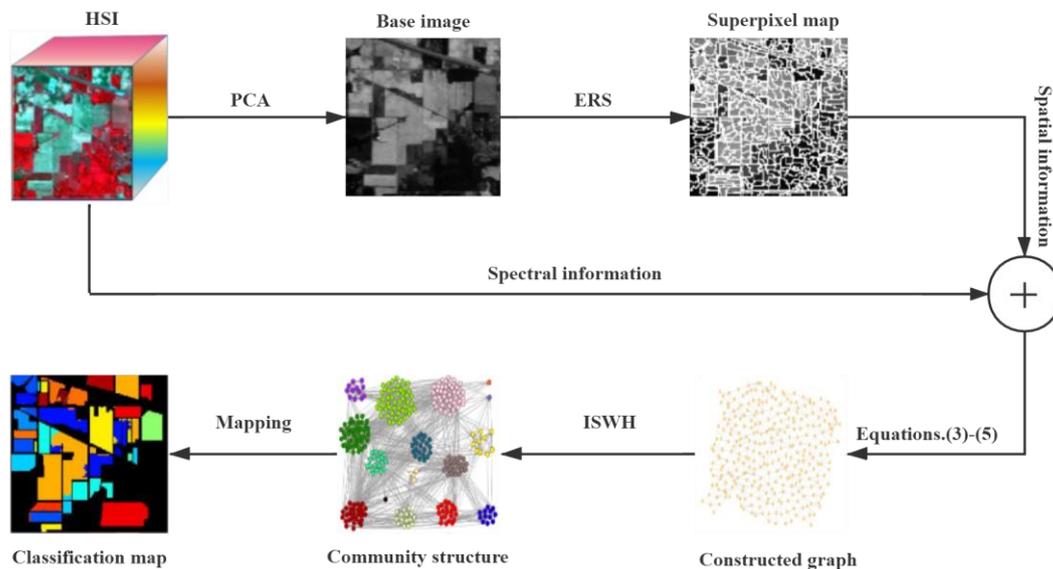


Figure 1. The proposed classification framework. The base image is generated using the first principal component from the PCA method. For clarity, the weights of each edge are omitted in the constructed graph and community structure. ERS: Entropy rate superpixel segmentation, HSI: Hyperspectral image, ISWH: Improved semi-supervised Wu–Huberman algorithm, PCA: Principal component analysis.

2.1. Superpixel Segmentation

Let $HSI = \{x_1, x_2, \dots, x_n\} \subset \mathbb{R}^B$ be a hyperspectral dataset with n pixels and B bands. The superpixel segmentation of an HSI is used to partition the HSI into different non-overlapping regions with an adaptive size and shape such that in each region, the spectrum of the pixels is as similar as possible and the position of the pixels is as a spatially nearest neighbor. It can be mathematically expressed as:

$$HSI = \cup_{i=1}^m S_i, S_i \cap S_j = \emptyset, i \neq j; i, j = 1, 2, \dots, m, \quad (1)$$

where S_i is a superpixel containing n_i pixels and m is the number of superpixels.

Hyperspectral data with hundreds of spectral bands do not allow us to partition HSI into superpixels by directly using the classical superpixel segmentation algorithms. This is because the classical superpixel segmentation methods were originally designed to segment color images and its workspace has a dimension of no more than three dimensions. Therefore, to split an HSI into superpixels, it is necessary to carry out a dimension reduction on the HSI in advance. In this study, we chose the popular unsupervised PCA method to reduce the dimension of an HSI and take the first principal component to generate the base image since the first principal component contains abundant information about the original hyperspectral data.

Plenty of works have demonstrated that as a popular segmentation method, the graph-based ERS method [31] has commonly been used in superpixel segmentation of HSIs [20,27,36] due to its advantages of better performance in terms of efficiency and effectiveness. Thus, the ERS segmentation method was used in our proposal to effectively generate a superpixel map. Alternatively, as one of the recommended six practical algorithms [57], a popular SLIC algorithm [26,30] can also be adopted to replace the ERS in our proposal because of its advantage of good adherence to class boundaries. However, this would involve more computational costs. There is no significant difference between the classification results obtained using ERS and SLIC methods in our proposed classification scheme.

In the graph-based ERS method, the base image generated by the first principal component is first mapped to a graph. Each pixel in the base image is treated as a node and the similarity between a pair of pixels is considered as the weight of an edge. For the user-predefined number of superpixels m user-predefined, the ERS method will partition the graph into more than m disjoint subsets S_i by clustering and optimizing the following objective function with respect to the selected edge set A :

$$\begin{cases} \max_A [H(A) + \lambda B(A)], \\ \text{s.t. } A \subseteq E \text{ and } N_A \geq m, \end{cases} \quad (2)$$

where $H(A)$ is the entropy rate term that is responsible for the compactness and homogeneity of the obtained subset, $B(A)$ is the balancing function that facilitates the generation of the subsets with the approximate same size, E is the edge set of the graph, N_A denotes the number of connected subgraphs in the graph, and λ is the weight of the balancing term. The greedy algorithm can be used to solve the optimization problem in Equation (2). A detailed description of the ERS method can be found in Liu et al. [31]. The procedure for the generation of the superpixels of HSI is illustrated in Figure 1.

2.2. The Construction of the Weighted Connectivity Graph

A large amount of work has proved that graph-based methods are popular and effective techniques in HSI classification [38–46] because graphs have the advantage of an amazing and flexible data presentation. In the graph-based HSI classification methods, nodes in the generated graph can represent pixels [38–45] or superpixels [46]. Considering superpixels as nodes in a graph implies a sharp reduction in the volume of the constructed graph. In other words, it is possible to quickly divide a graph into non-overlapping subgraphs. While doing so, it is natural that the superpixels obtained using the ERS method are regarded as nodes in the constructed graph.

In the graph-based classification methods, the construction of a graph is essential because the classification result is closely related to it. The two commonly used ways to generate graphs in machine learning are the k -nearest neighbors method and the ϵ -neighborhood technique. Notably, unlike other datasets, hyperspectral datasets contain massive amounts of spatial information in addition to a large amount of spectral information. This means that we can use the spectral features of pixels, together with the spatial information provided by superpixels, to construct graphs. Additionally, the closeness between nodes and the sparseness of the constructed graph are favorable for fast and accurate graph division. Starting from this idea, we introduce the following method for generating the desired graph.

Suppose that $G = (\mathbf{V}, \mathbf{E}, \mathbf{W})$ is a weight graph, where $\mathbf{V} = \{v_1, v_2, \dots, v_m\}$ is the set of nodes such that each node represents a superpixel, \mathbf{E} is a collection of edges connecting a pair of nodes, and \mathbf{W} is a weight set. In this work, we took the following steps to construct the weighted connectivity graph we needed:

- (i) For the computation of the distance between two spatially adjacent superpixels, assume that superpixels S_i and S_j are spatially adjacent. Based on B bands' worth of information of the pixel x_i and the spatial information of superpixel S_j , calculate the Euclidean distance $d(x_i, S_j)$ from a pixel $x_i \in S_i$ to superpixel S_j using the local mean-based pseudo-nearest-neighbor rule (LMPNN) [37,58]. Then, sort them in ascending order, denoted as $d(x'_1, S_j) \leq d(x'_2, S_j) \leq \dots \leq d(x'_{n_i}, S_j)$. The distance from superpixel S_i to superpixel S_j can be defined as:

$$d(S_i, S_j) = \sum_{h=1}^{n_i} \frac{1}{h} d(x'_h, S_j) \quad (3)$$

- (ii) For the calculation of weights, calculate the weights between a superpixel S_i and all its k_i spatial neighbors $S_{i,1}, S_{i,2}, \dots, S_{i,k_i}$ using Equation (4):

$$w_{i,j} = d^{-1}(S_i, S_{i,j}) / \sum_{h=1}^{k_i} d^{-1}(S_i, S_{i,h}), \quad j = 1, 2, \dots, k_i \quad (4)$$

- (iii) For the construction of the weighted graph, connect each superpixel S_i to its k_i spatial neighbors with the weights:

$$W_{i,j} = \begin{cases} \max\{w_{i,j}, w_{j,i}\}, & \text{if } v_i \text{ and } v_j \text{ are closest neighbors to each other,} \\ w_{i,j} \times w_{j,i}, & \text{otherwise.} \end{cases} \quad (5)$$

Here, v_j is the closest neighbor of v_i if (1) v_j is one of the neighbors of v_i , and (2) after sorting $w_{j,i}$ ($i = 1, 2, \dots, k_j$) and $w_{j,i}$ ($i = 1, 2, \dots, k_j$) in descending order, $w_{i,j}$ is located in the first half of the $w_{j,i}$ sequence and $w_{j,i}$ is also in the first half of the $w_{i,j}$ sequence.

The graph generated using Equation (5) is a weighted connectivity graph because we have not considered whether these superpixels only contain reference data in the process of connecting nodes. This means that reference data, together with the background pixels, will be involved in the classification process. Although doing so may have an effect on the classification results and increase the computation time, from an application perspective, the proposed method will be easy to use for solving other practical problems in remote sensing.

2.3. The Discrete Potential and the Improved Semisupervised Wu–Huberman Algorithm

For a given graph G , by connecting one node in the graph to the anode of a battery and the other node to the negative pole, the graph G can be represented as an electric circuit, with each edge acting as a resistance. In the UWH method [55], two nodes (distance > 2) were chosen randomly to attach to a battery with a fixed voltage. The potential/voltage of each node could be calculated by solving Kirchhoff's equations under the assumption that each edge has the same resistance and the fixed voltages were 1 and 0. The voltage of each node is updated for the number of times specified by the user. These voltages should stay between 0 and 1. Supposing that each community was approximately equal in size, two communities will be detected by using a certain threshold or a maximum voltage gap. To correct errors caused by random selection, this process is repeated many times and the final result is determined by the majority voting rule. Similarly, multiple communities are found one by one according to the above process. The complexity of the UWH algorithm is about $O(r(|V|+|E|))$, where r is the number of repetitions. For a detailed description of the UWH method, interested readers may consult Wu and Huberman [55].

Note that the assumptions of each edge with the same resistance and each community with approximately equal volumes limit the application of the UWH method. Additionally, it is also difficult to select an optimal threshold or find an ideal voltage gap, especially for a graph with a fuzzy or diverse community structure. To address these drawbacks, we hereby improve the UWH method to be a semi-supervised Wu–Huberman method such that the communities hidden in the weighted connectivity graph (WCG) will be detected better.

Supposed that there are C communities in a WCG and b nodes are labeled in each community. The basic idea of the improvement is to first generate C electric circuits and then physically compute C potentials for each unlabeled node in turn in terms of the ISWH. By comparing these C potentials, we finally decide which community the unlabeled node belongs to. Specifically, the graph is called the electric circuit generated by the c -label, where the potentials of the b nodes with label c are assigned a value of 1 and the potentials of the remaining labeled nodes (with labels other than c) are assigned a value of 0. The current will flow from the high potential to the low potential through each edge in the WCG.

According to Kirchhoff’s node equation and Ohm’s law:

$$\sum_{j=1}^{k_i} I_j = \sum_{j=1}^{k_i} \frac{P_j^c - P_i^c}{W_{i,j}^{-1}} = 0, \tag{6}$$

where I_j is the current flowing from v_j to v_i , k_i is the number of neighbors of node v_i , $W_{i,j}$ denotes the weight between v_i and v_j , and P_i^c represents the potential of node v_i in the electric circuit generated by the c -label. In Equation (6), $W_{i,j}^{-1}$ is used as the resistance instead of $W_{i,j}$. Because a larger $W_{i,j}$ shows that nodes v_i and v_j have a close neighbor relationship, they are more likely to belong to the same community. From a potential perspective, this means that there should be a small voltage drop between them. In the electric circuit generated by the c -label, the potential of each unlabeled node v_i will be obtained by rewriting Equation (6) into the following form:

$$P_i^c = \sum_{j=1}^{k_i} W_{i,j} P_j^c / \sum_{j=1}^{k_i} W_{i,j} \tag{7}$$

That is, the potential of an unlabeled node is the weighted average of the potentials of its neighbors.

Starting from b nodes with label c , we apply the breadth-first search algorithm of the graph, as well as Equation (7), to calculate P_i^c of each unlabeled node v_i one by one in the electric circuit generated by the c -label. P_i^c is then constantly updated by repeating this process many times. As a result, the potentials of unlabeled nodes belonging to the c th community gradually approach 1 due to the influence of b nodes with a unit potential. When c varies from 1 to C , we obtain C ordered potentials of each unlabeled node v_i , e.g., $P_i^1, P_i^2, \dots, P_i^C$. P_i^c is between 0 and 1, which reflects the relationship between node v_i and the c th community. Finally, the label of node v_i is assigned according to the following equation:

$$\text{Label}(v_i) = \arg \max_{1 \leq c \leq C} P_i^c \tag{8}$$

The process of community detection using the ISWH method is shown in Figure 2. For simplicity, each edge in the network is assigned the same weight. There are two apparent communities in this network. Taking the gray node as an example, it is assigned to the left community by comparing its two potentials 0.77 and 0.11, as well as according to Equation (8). Similarly, the remaining unlabeled nodes are classified one by one, and two communities are effectively detected.

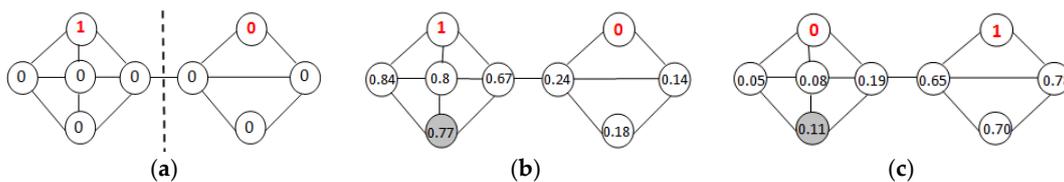


Figure 2. The process of discovering community structures using the ISWH method. The digit in each circle represents the potential of the node. A red digit indicates that the labeled node has a known potential. (a) A network of nine nodes with two communities. (b,c) Potential of each node obtained after five successive updates by assigning different initial values to the corresponding two labeled nodes.

In the ISWH, it takes $O(k_i)$ time to calculate the potential of each unlabeled node. Therefore, the total time spent in each repetition is about $O(|E|)$. Initializing the potential of all nodes needs $O(|V|)$ time. The complexity of the ISWH algorithm is about $O(rC(|V|+|E|))$. The number of nodes $|V|$ in graph G is the number of superpixels m . The number of edges $|E|$ is approximately a multiple of m because we only connect a superpixel to its k_i spatial neighbors. Therefore, the ISWH approximately takes $O(rCm)$ to complete the community discovery.

2.4. The Proposed SSC-GDP Method

To objectively prove the validity of the proposed method, we randomly labeled b pixels in each class instead of b superpixels due to the good feasibility of the pixel-wise labeling method in practical applications. According to the homogeneity of a superpixel, the superpixel acquires the label once one or more of its pixels are labeled. Because superpixel segmentation is not always perfect, two or more pixels may be labeled with different labels within the same superpixel. In this case, we adopted the majority voting rule to label it. Furthermore, it is also possible for a superpixel to contain more than two labeled pixels with the same label. In other words, the actual number of labeled superpixels is less than or equal to b in each class (the number of labeled nodes per community is not greater than b).

By converting the problem of label propagation in the graph into the potential transmission process in the electric circuit, the proposed method realizes HSI classification with the help of the community detection technique in complex networks. The proposed classification method is summarized below.

Input: HSI; b —the number of labeled pixels per class; m —the number of superpixels; r —the number of repetitions; C —the number of classes; $t = 0$.

Output: Classification result.

Step 1: Call the ERS algorithm to segment the HSI into superpixels.

Step 2: Construct the weighted connectivity graph using Equations (3)–(5).

Step 3: for $c = 1$ to C

{generate the electric circuit using the c -label.

Do {update the potential P_i^c of each unlabeled node v_i in the generated electric circuit through the breadth-first search algorithm of the graph, as well as Equation (7), $t = t + 1$ }

while $t < r$ }

Step 4: Assign the label of node v_i according to Equation (8).

3. Results

3.1. Experimental Setup

To evaluate the effectiveness of the proposed SSC-GDP algorithm, we took two Airborne Visible/Infrared Imaging Spectrometer (AVIRIS) datasets, namely an Indian Pines image and a Salinas image, as examples in our experiments. These two benchmark images are widely used to test the performance of HSI classification methods.

3.1.1. Data Description

Regarding the Indian Pines dataset, the Indian Pines image, acquired using the AVIRIS sensor in June 1992, covers the agricultural Indian Pines test site of Northwestern Indiana. The image has 220 bands of size 145×145 , with a spatial resolution of 20 m per pixel and a spectral range from 0.4 to $2.5 \mu\text{m}$. In our experiments, 20 water absorption bands were discarded. The ground truth contains 16 classes from different types of crops and a total of 10,249 labeled pixels (reference data). This is a disequilibrium data set where different classes vary greatly in size. The false-color composite image and the corresponding reference data of the Indian Pines dataset are represented in Figure 3 and Table 1, respectively.

Regarding the Salinas dataset, the Salinas image was also acquired using the AVIRIS sensor over Salinas Valley, CA, USA. The image is of size $512 \times 217 \times 224$, with a spatial resolution of 3.7 m per pixel. As with the Indian Pines image, 20 water absorption spectral bands were removed. It contains 16 different classes and 54,129 labeled pixels (reference data). The main challenge for this dataset is the accurate division between the 8th class and the 15th class because their spectral features are very similar. Figure 4 and Table 1 show the false-color composite image and the corresponding ground truth data, respectively, of the Salinas dataset.

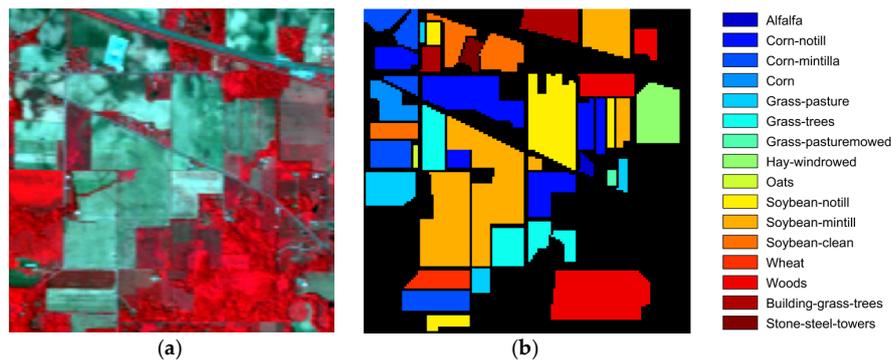


Figure 3. Indian Pine dataset: (a) false-color composite image and (b) reference image.

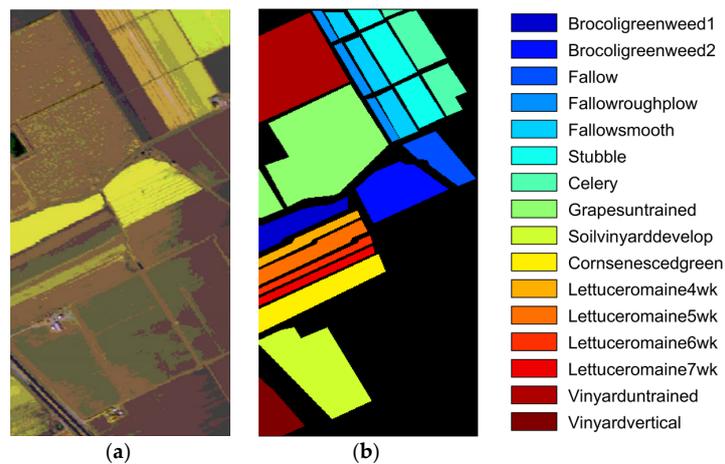


Figure 4. Salinas dataset: (a) false-color composite image and (b) reference image.

Table 1. Ground-truth classes for the Indian Pine and Salinas datasets.

Indian Pine		Pixels	Salinas		Pixels
Class	Name		Class	Name	
1	Alfalfa	46	1	Brocoli_green_weeds_1	2009
2	Corn-notill	1428	2	Brocoli_green_weeds_2	3726
3	Corn-mintill	830	3	Fallow	1976
4	Corn	237	4	Fallow_rough_plow	1394
5	Grass-pasture	483	5	Fallow_smooth	2678
6	Grass-trees	730	6	Stubble	3959
7	Grass-pasture-mowed	28	7	Celery	3579
8	Hay-windrowed	478	8	Grapes_untrained	11,271
9	Oats	20	9	Soil_vinyard_develop	6203
10	Soybean-notill	972	10	Corn_senesced_green_weeds	3278
11	Soybean-mintill	2455	11	Lettuce_roumaine_4wk	1068
12	Soybean-clean	593	12	Lettuce_roumaine_5wk	1927
13	Wheat	205	13	Lettuce_roumaine_6wk	916
14	Woods	1265	14	Lettuce_roumaine_7wk	1070
15	Buildings-Grass-Trees-Drives	386	15	Vinyard_untrained	7268
16	Stone-Steel-Towers	93	16	Vinyard_vertical_trellis	1807

3.1.2. Evaluation Protocol

For all experiments carried out in this work, each one was independently repeated 10 times; the average and standard deviation are reported in Tables 2 and 3. Three commonly used evaluation criteria—overall accuracy (OA), average accuracy (AA), and the kappa coefficient (κ)—were adopted to evaluate the performance of each HSI classification method.

Table 2. Classification accuracy (in percent) of the eight methods used on the Indian Pines image. EPF: edge-preserving filters, IFRF: image fusion and recursive filtering, LORSAL: logistic regression via variable splitting and augmented Lagrangian, SMLR-S: sparse multinomial logistic regression with a spatially adaptive total variation regularization, SCMK: superpixel-based classification via multiple kernels, SuperPCA: superpixel-wise PCA, SVM-SD: support vector machine based on superpixel and discontinuity-preserving relaxation. OA: overall accuracy, AA: average accuracy, κ : kappa coefficient.

Class	EPF	IFRF	LORSAL	SMLR-S	SCMK	SuperPCA	SVM-SD	Ours
1	99.03 ± 1.56	99.68 ± 1.02	88.39 ± 2.25	94.19 ± 1.36	96.78 ± 3.4	96.77 ± 3.4	92.58 ± 2.66	100 ± 0
2	45.33 ± 12.9	70.32 ± 9.2	36.74 ± 0.23	51.47 ± 0.2	69.23 ± 9.15	70.59 ± 13.74	64.46 ± 10.01	95.84 ± 1.92
3	63.9 ± 6.15	79.31 ± 8	51.96 ± 0.37	66.28 ± 0.29	72.77 ± 6.4	71.6 ± 13.1	80.04 ± 8.26	95.62 ± 2.21
4	94.23 ± 7.38	86.89 ± 6.85	48.33 ± 1.2	99.55 ± 0	88.69 ± 9.49	84.41 ± 11.58	78.74 ± 9.71	94.9 ± 8.06
5	88.76 ± 4.99	83.74 ± 7.02	69.59 ± 0.47	70.45 ± 0.48	87.29 ± 4.91	86.32 ± 6.22	78.8 ± 10.43	99.43 ± 0.99
6	97.02 ± 3.1	94.34 ± 2.75	86.43 ± 0.22	98.21 ± 0.09	96.9 ± 3.55	75.96 ± 12.33	96.77 ± 4.71	100 ± 0
7	93.85 ± 3.24	100 ± 0	96.92 ± 3.97	100 ± 0	96.16 ± 4.05	95.38 ± 3.97	100 ± 0	100 ± 0
8	98.23 ± 3.06	100 ± 0	86.37 ± 0.24	100 ± 0	99.53 ± 0.14	98.6 ± 3.07	91.92 ± 4.61	100 ± 0
9	100 ± 0	100 ± 0	70 ± 25.39	100 ± 0	100 ± 0	100 ± 0	100 ± 0	100 ± 0
10	70.41 ± 12.08	88.72 ± 4.55	69.75 ± 0.25	92.15 ± 0.1	78.43 ± 7.05	71.67 ± 8.05	88.14 ± 5.18	93.48 ± 5.77
11	69.04 ± 6.28	64.02 ± 8.01	39.89 ± 0.14	64.98 ± 0.1	68.99 ± 6.72	61.09 ± 20.52	64.38 ± 7.13	93.78 ± 2.14
12	64.98 ± 18.94	76.65 ± 5.31	57.56 ± 0.56	80.99 ± 0.38	71.69 ± 6.04	71.75 ± 6.65	66.09 ± 18.52	85.41 ± 2.45
13	99.37 ± 0.33	99.47 ± 0.25	97.11 ± 0.28	100 ± 0	97.16 ± 1.84	98.42 ± 1.11	97.84 ± 3.05	96.84 ± 2.74
14	87.62 ± 9.72	90.3 ± 6.63	93.12 ± 0.05	99.92 ± 0	94.99 ± 3.24	68.05 ± 12.13	76.28 ± 12.99	100 ± 0
15	73.32 ± 10.02	94.99 ± 3.68	33.29 ± 0.96	59.46 ± 0.84	87.41 ± 5.32	83.91 ± 13.53	89.06 ± 8.89	100 ± 0
16	99.74 ± 0.54	98.98 ± 0.54	79.36 ± 2.37	100 ± 0	97.95 ± 2.02	95.26 ± 7.65	95.26 ± 3.48	88.46 ± 11.1
OA (%)	74.95 ± 3.16	82.41 ± 2.39	59.63 ± 0.52	77.79 ± 2.51	83.07 ± 1.82	81.79 ± 3.03	81.54 ± 1.79	96.19 ± 0.55
AA (%)	84.05 ± 6.27	89.21 ± 3.99	69.05 ± 2.43	86.1 ± 0.24	87.75 ± 4.58	83.11 ± 8.57	85.02 ± 6.85	96.49 ± 2.34
κ	0.72 ± 0.03	0.80 ± 0.03	0.55 ± 0.01	0.75 ± 0.03	0.81 ± 0.02	0.79 ± 0.03	0.79 ± 0.02	0.96 ± 0.01

Table 3. Classification results (in percent) of various classifiers used on the Salinas dataset.

Class	EPF	IFRF	LORSAL	SMLR-S	SCMK	SuperPCA	SVM-SD	Ours
1	99.79 ± 0.47	100 ± 0	96.04 ± 1.95	99.9 ± 0	100 ± 0	100 ± 0.31	99.44 ± 0.02	100 ± 0
2	99.78 ± 0.16	99.16 ± 0.62	84.67 ± 0.97	100 ± 0	99.77 ± 0.03	89.62 ± 0.38	99.74 ± 0.01	100 ± 0
3	84.27 ± 3.93	100 ± 0	87.65 ± 3.11	96.96 ± 0.03	97.84 ± 2.66	95.84 ± 3.58	76.12 ± 0.05	100 ± 0
4	99.67 ± 0.12	97.23 ± 1.26	94.8 ± 1.68	99.28 ± 0.02	99.39 ± 0.57	90.39 ± 0.62	96.04 ± 0.03	98.66 ± 1.22
5	99.21 ± 0.34	98.97 ± 0.43	91.99 ± 1.6	99.03 ± 0.02	97.11 ± 0.87	96.6 ± 1.51	97.64 ± 0.04	97.12 ± 0.73
6	99.98 ± 0.01	99.88 ± 0.05	77.61 ± 0.08	100 ± 0	99.63 ± 0.25	81.5 ± 0.5	99.48 ± 0.02	99.77 ± 0.13
7	99.72 ± 0.03	99.75 ± 0.12	83.62 ± 0.18	99.97 ± 0	99.17 ± 0.11	89.37 ± 0.38	99.06 ± 0.01	100 ± 0
8	76.81 ± 12.82	74.48 ± 9.73	62.41 ± 3.13	84.27 ± 0.02	77.11 ± 7.71	80.04 ± 4.87	84.47 ± 0.02	98.98 ± 0.77
9	98.98 ± 0.26	99.98 ± 0	68.17 ± 0.54	100 ± 0	96.77 ± 2.4	81.75 ± 2.37	70.75 ± 0.01	99.75 ± 0.56
10	90.43 ± 3.83	99.69 ± 0.27	75.48 ± 1.2	93.45 ± 0.04	91.19 ± 3.67	84.86 ± 1.31	88.9 ± 0.07	96.89 ± 2.76
11	98.2 ± 1.65	92.87 ± 2.93	85.33 ± 0.39	99.82 ± 0.03	97.07 ± 2.54	92.68 ± 2.35	47.68 ± 0.04	97.23 ± 4.11
12	100 ± 0	99.19 ± 1.56	78.35 ± 0.83	99.84 ± 0.01	97.84 ± 3.26	87.24 ± 1.32	33.86 ± 0.02	87.73 ± 3.38
13	97.8 ± 0.42	88.42 ± 6.85	98.12 ± 0.05	98.23 ± 0.05	98.04 ± 0.31	98.11 ± 0.94	91.64 ± 0.05	98.3 ± 0.61
14	94.36 ± 2.16	93.54 ± 1.64	89.39 ± 2.63	99.9 ± 0.01	95.86 ± 1.84	90.17 ± 1.64	97.7 ± 0.07	92.23 ± 6.39
15	70.85 ± 13.47	86.26 ± 9.58	63.46 ± 1.27	76.32 ± 0.03	83.76 ± 7.15	71.87 ± 1.51	81.2 ± 0.03	98.15 ± 0.73
16	95.5 ± 3.57	98.83 ± 0.4	88.51 ± 2.06	99 ± 0.02	98.22 ± 2.05	95.28 ± 2.04	97.62 ± 0.04	100 ± 0
OA (%)	89.53 ± 1.87	92.06 ± 2.07	75.50 ± 2.12	92.87 ± 0.01	92.87 ± 1.29	84.71 ± 3.27	85.34 ± 0.01	98.46 ± 0.21
AA (%)	94.08 ± 2.7	95.52 ± 2.22	82.85 ± 1.35	96.62 ± 0.02	95.55 ± 2.21	89.08 ± 1.6	85.08 ± 0.03	97.8 ± 1.34
κ	0.88 ± 0.02	0.91 ± 0.02	0.73 ± 0.02	0.92 ± 0.01	0.92 ± 0.01	0.83 ± 0.04	0.83 ± 0.01	0.98 ± 0.002

The classification results of the proposed SLC-DP method were visually and quantitatively compared with those provided by several state-of-the-art HSI classification approaches, i.e., edge-preserving filters (EPF) [59], image fusion and recursive filtering (IFRF) [60], logistic regression via variable splitting and augmented Lagrangian (LORSAL) [61], sparse multinomial logistic regression with a spatially adaptive total variation regularization (SMLR-S) [62], superpixel-based classification via multiple kernels (SCMK) [20], superpixel-wise PCA (SuperPCA) [63], and a support vector machine (SVM) based on superpixel and discontinuity-preserving relaxation (SVM-SD) [63]. The EPF, IFRF, SMLR-S, SCMK, SuperPCA, and SVM-SD algorithms are spectral–spatial classifiers, whereas the LORSAL method does not consider the spatial information of pixels in the classification.

3.2. Classification Results

Table 2 reports the classification results provided by eight methods on the Indian Pines dataset, in which the number of superpixels m was equal to 700 and 20 pixels per class were labeled. As seen in Table 2, compared with the seven other pixel-wise approaches, the proposed SSC-GDP method achieved the best classification result (96.19%), at least 13% higher than the others. Furthermore, our proposal correctly classified 7 of the 16 classes. The LORSAL classifier provided the lowest classification accuracy (59.63%) because the spatial context of the HSI was not taken into account in the classification. The use of the SVM and multiple kernels technique made SCMK superior to the other five spectral–spatial HSI methods, namely EPF, IFRF, SKLR-S, SuperPCA, and SVM-SD. The EPF classifier did not produce a satisfactory classification result (74.95%), which was partially due to the complexity of the class boundary and the application of a PCA technique.

The visual classification maps produced by different classifiers are shown in Figure 5. One can see from Figure 5 that, unlike several other classification results, misclassification in our classification map occurred only between spatially adjacent classes. This was because, on the one hand, only spatially adjacent superpixels were interconnected in the constructed weighted graph; on the other hand, the class label was incorrectly transmitted in the ISWH method using mixed superpixels containing reference data and background pixels, as well as those containing only background pixels. Although the removal of background pixels and the reprocessing of mixed superpixels may improve the classification accuracy, this is not convenient for practical applications.

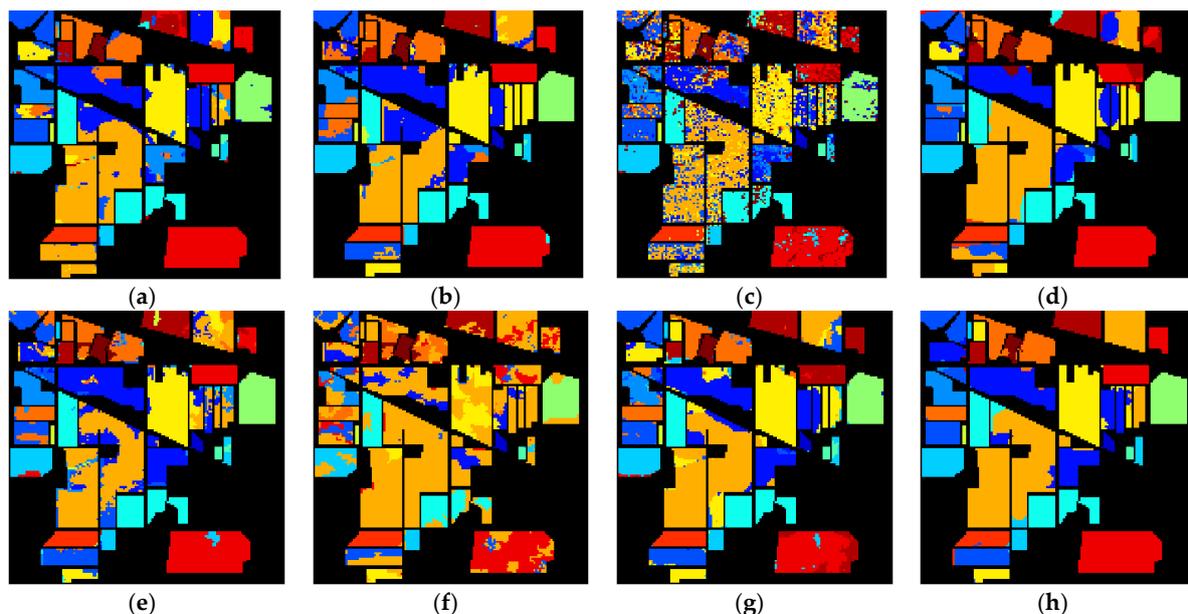


Figure 5. The classification maps of the Indian Pines dataset produced using (a) EPF, (b) IFRF, (c) LORSAL, (d) SMLR-S, (e) SCMK, (f) SuperPCA, (g) SVM-SD, and (h) our proposed method.

Table 3 lists the statistical results obtained by using various classifiers on the Salinas image, where the number of superpixels m was equal to 1000 and 20 pixels per class were marked. It is easy to understand from Table 3 that the proposed classification scheme was superior to the other seven competitive methods according to three indices. For the 8th class and the 15th class that are easily misclassified, our proposal achieved satisfactory classification results (98.98% and 98.15%, respectively). As can be seen from Figure 6h, a small number of pixels located near their class boundaries were misclassified and the remaining pixels were correctly classified. This was because some mixed superpixels were misclassified during the classification process. Additionally, the participation of background data in the classification process was a cause of misclassification. The classification accuracy of the LORSAL method was nearly 20% lower than our proposal. Compared with the SCMK and SMLR-S classifiers, the proposed method improved the classification results by about 5% in terms of the OA. Among the seven spectral–spatial methods, the classification accuracy of the SuperPCA method was unsatisfactory. This may have been due to the use of PCA, which resulted in the loss of some information. From a quantitative and visual perspective, our results confirmed that the proposed superpixel-level classification approach displayed a good performance in the case of limited labeled pixels.

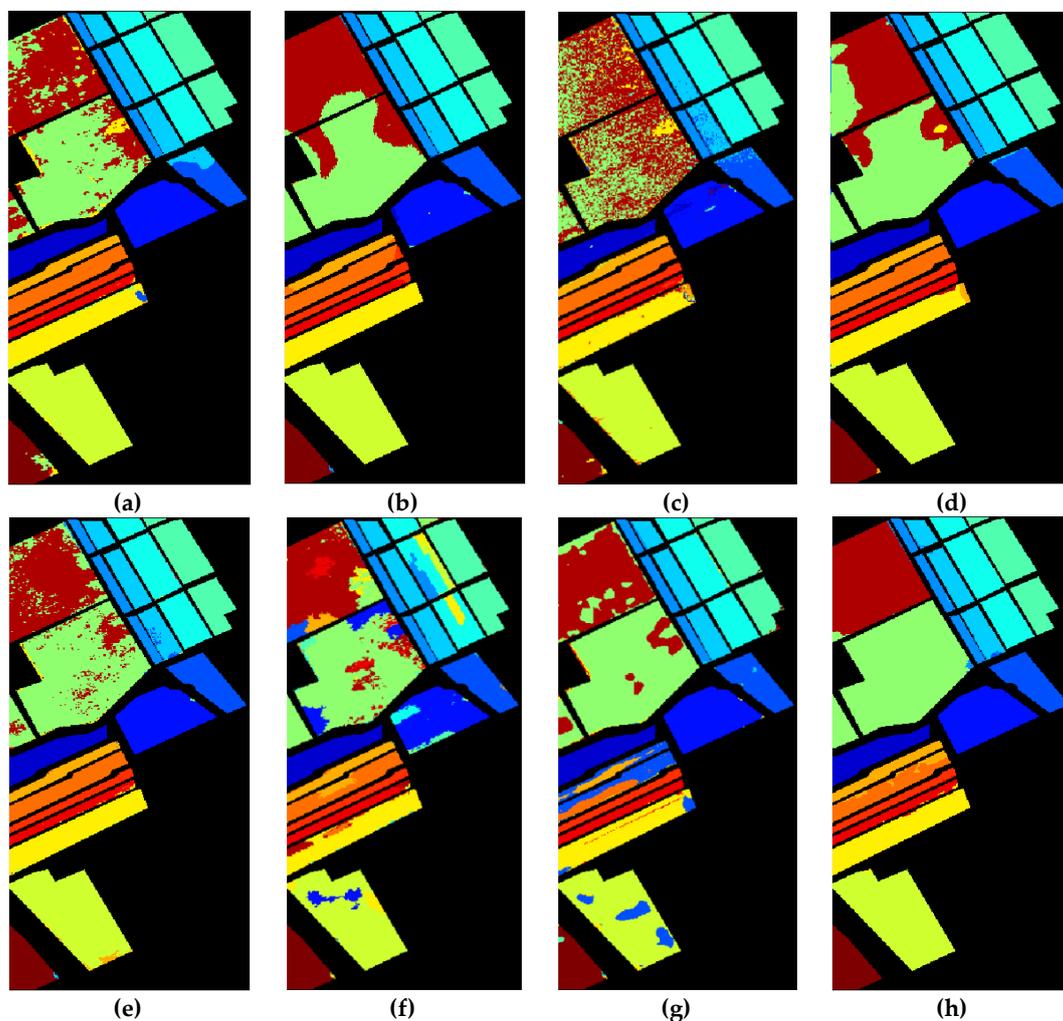


Figure 6. The classification maps of the Salinas dataset produced using (a) EPF, (b) IFRF, (c) LORSAL, (d) SMLR-S, (e) SCMK, (f) SuperPCA, (g) SVM-SD, and (h) our proposed method.

3.3. Effect of the Number of Superpixels

The classification results obtained by the proposed method varied with the number of superpixels, as shown in Figure 7. Here, ratio = 3 means that three pixels per class were marked. As the number of superpixels m changed from 300 to 1000, the classification accuracy on the Indian Pines dataset appeared to rise at first and then decrease. This may have been due to the fact: (i) A small number of superpixels means large superpixel volume. Accordingly, the probability that the pixels with different class labels are in one superpixel will be increased. This will have an impact on the classification results of the proposed superpixel-level classification algorithm. (ii) In general, a superpixel with a small size indicates it has a better homogeneity. This fact seemed to help us group them correctly. However, similar to the case of pixels, the separability between superpixel blocks was reduced. The difference between the classification results obtained by the same labeled ratio gradually reduced when the labeled ratio varied from 3 to 20. In particular, in the case of ratio=10, 15, and 20, the difference between different classification accuracies was less than 1% when the number of superpixels changed from 500 to 900. This indicates that the proposed classification framework was not sensitive to the number of superpixels in this range. The experimental results show that the OA reached its maximum for $m = 700$ on this dataset. As can be observed in Figure 7b, all OA values for the Salinas image were greater than 94.5%, even for ratio = 3. This result explains why the proposed method can be thought of as a successful attempt to explore superpixel-level HSI classification. For the same labeled ratio, a small classification difference (not greater than 1.5%) means that the proposal was insensitive to segmentation scales (from 500 to 1200) on this dataset. It should be pointed out that the optimal number of superpixels used in this work was an experimental result. In a variety of applications, it remains a challenging problem to determine the optimal segmentation scale without reference data.

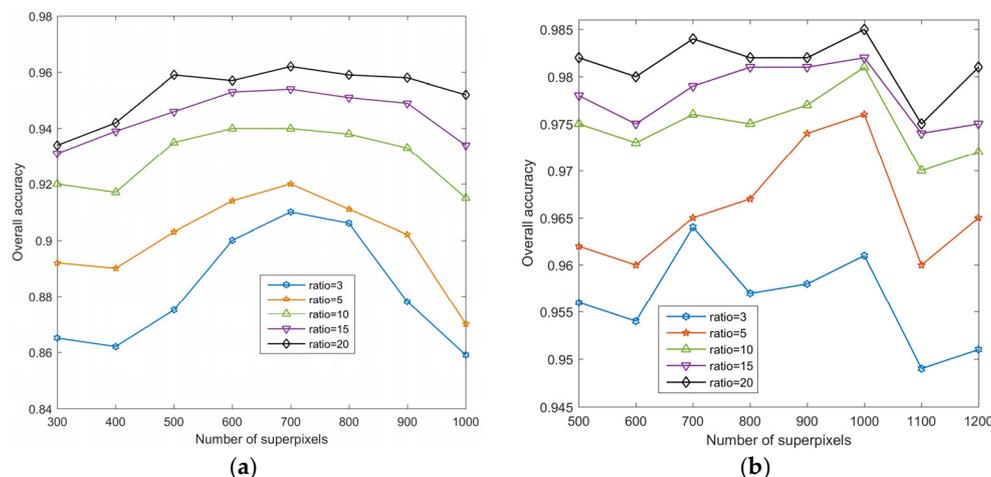


Figure 7. The effect of the number of superpixels on the classification accuracy: (a) the Indian Pines dataset and (b) the Salinas dataset.

3.4. Comparison of Several Competitive Methods

The comparison results for numerous competitive classifiers on the Indian Pines and Salinas datasets are shown in Figure 8. It is indicated in Figure 8 that our classification results were significantly better than those of the other seven state-of-the-art algorithms for different labeled ratios. For the Indian Pines image shown in Figure 8a, unlike the other seven competitive methods, the classification accuracy provided by the proposed method showed a slow upward trend with the increase in the number of labeled pixels. This may have been because two or more pixels within the same superpixel were labeled. This does not help improve the classification results of our method. The LORSAL method did not obtain a satisfactory classification result because of the lack of spatial information in the classification. The fact that the classification accuracy of the SMLR-S and SVM-SD approaches increased steadily with the increase of the number of labeled pixels implies that the algorithm was dependent

on the mark proportion. The classification results of the SCMK, IFRF, EPF, and SuperPCA methods exhibited a similar tendency. As a whole, SCMK and IFRF outperformed the other two. As can be seen from Figure 8b, our method achieved satisfactory classification results (higher than 95%) on the Salinas dataset. The proposed method had about a 5% advantage over the other seven classifiers. Because the regular class shape of this dataset enhanced the smoothing effect of the spatial adaptive regularization technique, the SMLR-S method was superior to the other five spectral–spatial classification algorithms. The SuperPCA algorithm, which is closely related to the superpixel segmentation scale, did not achieve a desired classification effect on the image. Experimental results on these two datasets confirmed that the proposed superpixel-level classification scheme was preferable to the other seven pixel-wise classification algorithms.

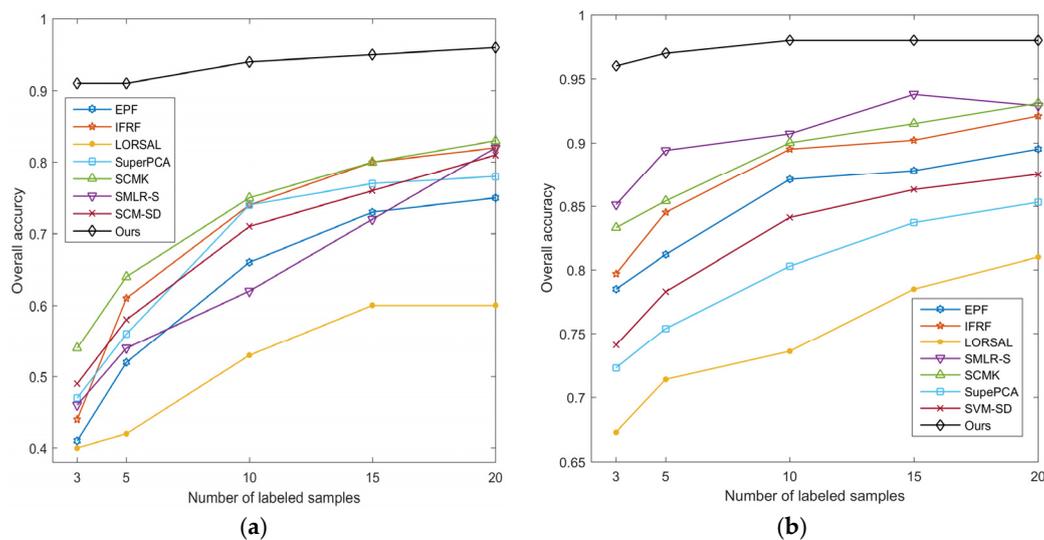


Figure 8. Comparison results of eight classifiers on the (a) Indian Pines and (b) Salinas datasets.

3.5. Impact of the Number of Updates on the Classification Results

In this experiment, 20 pixels per class were randomly labeled and the number of superpixels m was 700 and 1000 for the Indian Pines and Salinas datasets, respectively. In the ISWH method, the potential of each unlabeled node is continually updated using Equation (7) in the electric circuit generated by the c -label. As the number of updates increases, the difference between the potentials of the nodes belonging to the c th community and the remaining nodes increases. This is propitious for the detection of the c th community. This fact was demonstrated for the Indian Pines dataset, as shown in Figure 9. The classification results of the Indian Pines image became increasingly better when the number of updates increased from 5 to 20, and then tended to be stable (about 0.5% difference). For the Salinas image, after 10 updates, the classification accuracy reached 98.01%, and then slightly increased by about 0.2%. Afterward, it showed a steady trend when the number of updates was greater than or equal to 15. This may be explained by the fact that this dataset had a high spatial resolution and a good separation between classes. The fast convergence of the updating process illustrates the stability of the ISWH algorithm. At the same time, this is also one of the reasons why the proposed algorithm could realize fast classification. For convenience, the number of updates was uniformly taken as 20 in our experiments.

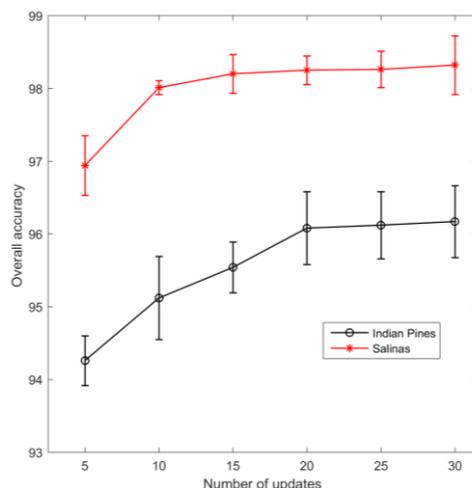


Figure 9. The variation of classification results on two hyperspectral datasets with the number of updates.

3.6. Execution Time

In this work, all the experiments were accomplished on the software Matlab 2015b. The parameters of the laptop used were AMD 2600 CPU with 3.5GHz, 16GB memory, and a Windows system. We report the running times of the eight HSI classification algorithms on the Indian Pines and Salinas datasets in Table 4. Each method was independently tested 10 times, with 20 labeled pixels per class. The number of superpixels was 700 and 1000 for the Indian Pines and Salinas datasets, respectively. The LORSAL algorithm showed the fastest performance due to considering only spectral information in the classification. Because of the adoption of sparse multinomial logistic regression, edge detection, and a weighted Markov random field, as well as the optimization of iterative updates, the SMLR-S method required more time to complete the classification task. Using a probability optimization strategy in EPD and SVM-SD algorithms led to their relatively high computational complexity.

Table 4. Running time (in seconds) of eight methods on two hyperspectral datasets.

Dataset	EPF	IFRF	LORSAL	SMLR-S	SCMK	SuperPCA	SVM-SD	Ours
Indian	18.53 ± 0.08	5.98 ± 0.03	0.74 ± 0.02	27.92 ± 0.07	5.35 ± 0.07	8.64 ± 0.16	12.37 ± 0.16	9.71 ± 0.20
Salinas	32.93 ± 0.44	13.11 ± 0.19	2.6 ± 0.08	124.44 ± 0.21	10.49 ± 0.12	17.67 ± 0.14	18.31 ± 0.42	17.86 ± 0.13

The main computing time of the proposed method was the construction of the weighted connectivity graph because it needed to calculate the distance between each superpixel and its spatial neighbors. Unlike the SCMK, SuperPCA, and SVM-SD methods, the proposed SSC-GDP method used the whole HSI, not just the reference data. Note that the Indian pines dataset consisted of 10,249 reference pixels and 10,766 background pixels, and the Salinas image was made up of 54,129 reference pixels and 56,975 background pixels. Thus, our algorithm was the most time-efficient of the four superpixel-related approaches. In terms of the classification results and computing time, our proposal is a satisfactory classification framework.

4. Discussion

The basis of the proposed SSC-GDP method was the assumption of superpixel homogeneity, i.e., pixels in a superpixel will share the same class label with a high probability. This assumption allowed us to represent a superpixel as a new sample in HSI classification. As a result, the volume of hyperspectral data is reduced greatly. This implies that classification is likely to be completed in a short time. Different from dimensionality reduction, this is a data reduction technique from a pixel perspective. A high homogeneity of superpixels and an appropriate number of superpixels are key

factors affecting the classification results. In existing superpixel-based, pixel-wise HSI classifiers and four superpixel-level HSI classification approaches [35–37,46], the optimal number of superpixels is determined using the classification results of reference data. However, the hyperspectral data obtained in practice have little reference data. Therefore, the development of methods that can be used to find the optimal number of superpixels for the HSI without reference data is worthy of discussion.

To our best knowledge, there are four superpixel-level HSI classification methods [35–37,46]. Compared with the approaches proposed in References [35,46], the advantage of the proposed SSC-GDP classification framework is that it is nonparametric and easy to calculate. The use of a local connection strategy in the graph construction, as well as the linear complexity of the ISWH algorithm, makes our proposal superior to the method developed by Xie et al. [37]. Additionally, these four methods only consider the reference data in the process of classification. However, in our proposal, both the reference data and background data are involved in the classification because we have not considered whether superpixels only contain reference data when constructing graphs.

For superpixel-level HSI classification methods, it is of fundamental importance to properly measure the similarity between two superpixels. Based on an affine hull model and singular value decomposition [35], KNN [36], the extended LMPNN [37], and covariance matrix and kernel technique [46], the similarity defined in these four methods may measure the distance between two superpixels better. However, the calculation of these similarities is complicated, multi-parametric, or not fast enough. Unlike in the case of pixels, the characteristic of superpixels with adaptive shapes and sizes also makes it difficult to properly define and quickly compute the similarity between superpixels. To classify HSI quickly and accurately, it is necessary to design new methods to address this problem in our future work.

It is well known that superpixel segmentation is not always perfect. Therefore, a superpixel may contain a large number of pixels belonging to the same class and a few heterogeneous pixels. Furthermore, massive noisy pixels in the HSI will affect the similarity between superpixels. To weaken their influence on the calculation of the distance between superpixels, this problem is handled carefully using a weighting technique and the local average pseudo-nearest neighbor method defined in Equation (3). However, recalculating the distance from a pixel x_i to each local average pseudo-nearest neighbor will require more computational cost. This influences the efficiency of our method and is a weakness of our proposal. Additionally, our method can work well on those hyperspectral datasets whose class distribution is spatially continuous because the potential of labeled nodes can be transmitted effectively. For those hyperspectral datasets composed of fragmented classes, the potential of a labeled node needs to pass through different classes or background data to be transferred to the node to be classified, thus it will significantly attenuate. This means that the probability of misclassification will increase, which limits the working space of our method. In our classification results, the misclassified pixels were mainly located near the class boundary. Our future work will investigate how to avoid or reduce the misclassified pixels near the class boundary.

5. Conclusions

This paper suggests a novel, semi-supervised, superpixel-level classification method for HSI based on a graph and discrete potential method. The proposal aimed to rapidly and accurately classify an HSI. The advantages of the introduced classification algorithm are: (i) Data reduction since taking each superpixel as a new sample greatly reduced the volume of the original hyperspectral dataset, and thus saved the classification time. (ii) Denoising, where the assumption of homogeneity of superpixels played an important role in removing noise pixels in the classification process. (iii) Good classification performance, where in the case of only a few labeled pixels per class, our method still showed a good classification ability. Additionally, the proposed classification method ran on the whole HSI image, not just the reference data. This implies that the proposal has a better application prospect in the field of remote sensing. This scheme provides a new idea for HSI classification by introducing a community

discovery technique. Experimental results and comparison results have demonstrated the effectiveness of the method.

Author Contributions: Conceptualization, F.S. and Y.Z.; methodology, F.S. and Y.Z.; software, Y.Z.; formal analysis, Y.Z. and F.Y.; data curation, F.Y.; writing—original draft preparation, Y.Z.; writing—review & editing, Y.Z. and F.S.; visualization, Y.Z. and F.Y.; funding acquisition, F.S. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the National Natural Science Foundation of China (grant no. 41890854).

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

References

1. Muller-Karger, F.; Roffer, M.; Walker, N.; Oliver, M. Satellite remote sensing in support of an integrated ocean observing system. *IEEE Geosci. Remote Sens. Mag.* **2013**, *1*, 8–18. [[CrossRef](#)]
2. Heldens, W.; Heiden, U.; Esch, T.; Stein, E.; Müller, A. Can the future Enmap mission contribute to urban applications? A literature survey. *Remote Sens.* **2011**, *3*, 1817–1846. [[CrossRef](#)]
3. Ryan, J.; Davis, C.; Tuffillaro, N.; Kudela, R.; Gao, B. Application of the hyperspectral imager for the coastal ocean to phytoplankton ecology studies in Monterey Bay CA, USA. *Remote Sens.* **2014**, *6*, 1007. [[CrossRef](#)]
4. Prasad, S.; Bruce, L. Limitations of principal components analysis for hyperspectral target recognition. *IEEE Geosci. Remote Sens. Lett.* **2008**, *5*, 625–629. [[CrossRef](#)]
5. Liu, L.; Li, C.; Lei, Y.; Yin, J.; Zhao, J. Feature extraction for hyperspectral remote sensing image using weighted PCA-ICA. *Arab. J. Geosci.* **2017**, *10*, 307. [[CrossRef](#)]
6. Falco, N.; Benediktsson, J.; Bruzzone, L. A study on the effectiveness of different independent component analysis algorithms for hyperspectral image classification. *IEEE Trans. Geosci. Remote Sens.* **2014**, *7*, 2183–2199. [[CrossRef](#)]
7. Roweis, S.; Saul, L. Nonlinear dimensionality reduction by locally linear embedding. *Science* **2000**, *290*, 2323–2326. [[CrossRef](#)]
8. Wang, Q.; Meng, Z.; Li, X. Locality adaptive discriminant analysis for spectral-spatial classification of hyperspectral images. *IEEE Trans. Geosci. Remote Sens. Lett.* **2017**, *99*, 1–5. [[CrossRef](#)]
9. Feng, J.; Jiao, L.; Liu, F.; Sun, T.; Zhang, X.-R. Unsupervised feature selection based on maximum information and minimum redundancy for hyperspectral images. *Patt. Recognit.* **2016**, *51*, 295–309. [[CrossRef](#)]
10. Xie, F.; Li, F.; Lei, C.; Yang, J.; Zhang, Y. Unsupervised band selection based on artificial bee colony algorithm for hyperspectral image classification. *Appl. Soft Comput.* **2019**, *75*, 428–440. [[CrossRef](#)]
11. Wang, L.; Jia, X.; Zhang, Y. A novel geometry-based feature-selection technique for hyperspectral imagery. *IEEE Trans. Geosci. Remote Sens. Lett.* **2007**, *4*, 171–175. [[CrossRef](#)]
12. Xie, F.; Lei, C.; Li, F.; Huang, D.; Yang, J. Unsupervised hyperspectral feature selection based on fuzzy c-means and grey wolf optimizer. *Inter. J. Remote Sen.* **2019**, *40*, 3344–3367. [[CrossRef](#)]
13. Li, J.; Khodadadzadeh, M.; Plaza, J.; Jia, X.; Bioucas-Dias, J.M. A discontinuity preserving relaxation scheme for spectral-spatial hyperspectral image classification. *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.* **2017**, *9*, 625–639. [[CrossRef](#)]
14. Li, J.; Bioucas-Dias, J.; Plaza, A. Semi-supervised hyperspectral image segmentation using multinomial logistic regression with active learning. *IEEE Trans. Geosci. Remote Sens.* **2010**, *48*, 4085–4098.
15. Dundar, T.; Ince, T. Sparse representation-based hyperspectral image classification using multiscale superpixels and guided filter. *IEEE Trans. Geosci. Remote Sens. Lett.* **2019**, *16*, 246–250. [[CrossRef](#)]
16. Li, W.; Prasad, S.; Fowler, J.E. Hyperspectral image classification using gaussian mixture models and markov random fields. *IEEE Trans. Geosci. Remote Sens. Lett.* **2013**, *11*, 153–157. [[CrossRef](#)]
17. Li, J.; Bioucas-Dias, J.; Plaza, A. Spectral-spatial hyperspectral image segmentation using subspace multinomial logistic regression and Markov random fields. *IEEE Trans. Geosci. Remote Sens.* **2012**, *50*, 809–823. [[CrossRef](#)]
18. Melgani, F.; Bruzzone, L. Classification of hyperspectral remote sensing images with support vector machines. *IEEE Trans. Geosci. Remote Sens.* **2004**, *42*, 1778–1790. [[CrossRef](#)]

19. Yu, H.; Gao, L.; Li, J.; Li, S.; Zhang, B.; Benediktsson, J. Spectral-spatial hyperspectral image classification using subspace-based support vector machines and adaptive Markov random fields. *Remote Sens.* **2016**, *8*, 355. [[CrossRef](#)]
20. Fang, L.; Li, S.; Duan, W.; Ren, J.; Benediktsson, J. Classification of hyperspectral images by exploiting spectral-spatial information of superpixel via multiple kernels. *IEEE Trans. Geosci. Remote Sens.* **2015**, *53*, 6663–6674. [[CrossRef](#)]
21. Camps-Valls, G.; Gómez-Chova, L.; Muñoz-Marí, J.; Vila-Frances, J.; Calpe-Maravilla, J. Composite Kernels for Hyperspectral Image Classification. *IEEE Trans. Geosci. Remote Sens. Lett.* **2006**, *3*, 93–97. [[CrossRef](#)]
22. Fang, L.; Li, S.; Kang, X.; Benediktsson, J. Spectral-spatial hyperspectral image classification via multiscale adaptive sparse representation. *IEEE Trans. Geosci. Remote Sens.* **2014**, *52*, 7738–7749. [[CrossRef](#)]
23. Gao, Q.; Lim, S.; Jia, X. Spectral-spatial hyperspectral image classification using a multiscale conservative smoothing scheme and adaptive sparse representation. *IEEE Trans. Geosci. Remote Sens.* **2019**, *99*, 1–13. [[CrossRef](#)]
24. Mei, S.; Hou, J.; Chen, J.; Chau, L.; Du, Q. Simultaneous Spatial and Spectral Low-Rank Representation of Hyperspectral Images for Classification. *IEEE Trans. Geosci. Remote Sens.* **2018**, *56*, 2872–2886. [[CrossRef](#)]
25. Zhan, T.; Sun, L.; Xu, Y.; Yang, G.; Zhang, Y.; Wu, Z. Hyperspectral Classification via Superpixel Kernel Learning-Based Low Rank Representation. *Remote Sensing* **2018**, *10*, 1639. [[CrossRef](#)]
26. Liu, C.; Li, J.; He, L. Superpixel-Based semi-supervised active learning for hyperspectral image classification. *IEEE J. Sel. Topics Appl. Earth Obs. Remote Sens.* **2019**, *12*, 357–370.
27. Jia, S.; Deng, B.; Zhu, J.; Jia, X.; Li, Q. Local binary pattern-based hyperspectral image classification with superpixel guidance. *IEEE Trans. Geosci. Remote Sens.* **2018**, *56*, 749–759. [[CrossRef](#)]
28. Saranathan, A.; Parente, M. Uniformity-based superpixel segmentation of hyperspectral images. *IEEE Trans. Geosci. Remote Sens.* **2016**, *54*, 1419–1430. [[CrossRef](#)]
29. Liang, M.; Jiao, L.; Meng, Z. A superpixel-based relational auto-encoder for feature extraction of hyperspectral images. *Remote Sens.* **2019**, *11*, 2454. [[CrossRef](#)]
30. Liu, H.; Li, J.; He, L.; Wang, Y. Superpixel-guided layer-wise embedding CNN for remote sensing image classification. *Remote Sens.* **2019**, *11*, 174. [[CrossRef](#)]
31. Liu, E.; Tuzel, O.; Ramalingam, S.; Chellappa, R. Entropy rate superpixel segmentation. In Proceedings of the CVPR2011, Providence, RI, USA, 20–25 June 2011; pp. 2097–2104. [[CrossRef](#)]
32. Achanta, R.; Shaji, A.; Smith, K.; Lucchi, A.; Fua, P. SLIC superpixels compared to state-of-the-art superpixel methods. *IEEE Trans. Patt. Anal. Mach. Intell.* **2012**, *34*, 2274–2282. [[CrossRef](#)] [[PubMed](#)]
33. Bergh, M.; Boix, X.; Roig, G.; Van-Gool, L. SEEDS: Superpixels Extracted Via Energy-Driven Sampling. *Int. J. Comput. Vis.* **2015**, *111*, 298–314. [[CrossRef](#)]
34. Vincent, L.; Soille, P. Watersheds in digital spaces: An efficient algorithm based on immersion simulations. *IEEE Trans. Patt. Anal. Mach. Intell.* **1991**, *13*, 583–598. [[CrossRef](#)]
35. Lu, T.; Li, S.; Fang, L.; Bruzzone, L.; Benediktsson, J. Set-to-set distance-based spectral-spatial classification of hyperspectral images. *IEEE Trans. Geosci. Remote Sens.* **2016**, *54*, 1–13. [[CrossRef](#)]
36. Tu, B.; Wang, J.; Kang, X.; Zhang, G.; Ou, X.; Guo, L. KNN-based representation of superpixels for hyperspectral image classification. *IEEE J. Sel. Topics Appl. Earth Obs. Remote Sens.* **2018**, *11*, 4032–4047. [[CrossRef](#)]
37. Xie, F.; Lei, C.; Jin, C.; An, N. A novel spectral-spatial classification method for hyperspectral image at superpixel Level. *Appl. Sci.* **2020**, *10*, 463. [[CrossRef](#)]
38. Leng, Q.; Yang, H.; Jiang, J. Label noise cleansing with sparse graph for hyperspectral image classification. *Remote Sens.* **2019**, *11*, 1116. [[CrossRef](#)]
39. Shao, Y.; Sang, N.; Gao, C.; Ma, L. Spatial and class structure regularized sparse representation graph for semi-supervised hyperspectral image classification. *Pattern Recognit.* **2018**, *81*, 81–94. [[CrossRef](#)]
40. Gao, F.; Wang, Q.; Dong, J.; Xu, Q. Spectral and Spatial Classification of Hyperspectral Images Based on Random Multi-Graphs. *Remote Sens.* **2018**, *10*, 1271. [[CrossRef](#)]
41. Cui, B.; Xie, X.; Ma, X.; Ren, G.; Ma, Y. Superpixel-Based Extended Random Walker for Hyperspectral Image Classification. *IEEE Trans. Geosci. Remote Sens.* **2018**, *56*, 3233–3243. [[CrossRef](#)]
42. Chen, M.; Wang, Q.; Li, X. Discriminant analysis with graph learning for hyperspectral image classification. *Remote Sens.* **2018**, *10*, 836. [[CrossRef](#)]

43. Camps-Valls, G.; Bandos-Marsheva, T.; Zhou, D. Semi-Supervised Graph-Based Hyperspectral Image Classification. *IEEE Trans. Geosci. Remote Sens.* **2007**, *45*, 3044–3054. [[CrossRef](#)]
44. Gao, Y.; Ji, R.; Cui, P.; Dai, Q.; Hua, G. Hyperspectral image classification through bilayer graph-based learning. *IEEE Trans. Imag. Proc.* **2014**, *23*, 2769–2778. [[CrossRef](#)] [[PubMed](#)]
45. Cui, B.; Xie, X.; Hao, S.; Cui, J.; Lu, Y. Semi-supervised classification of hyperspectral images based on extended label propagation and rolling guidance filtering. *Remote Sens.* **2018**, *10*, 515. [[CrossRef](#)]
46. Sellars, P.; Aviles-Rivero, A.; Papadakis, N.; Coomes, D.; Faul, A.; Schönlieb, C.B. *Semi-Supervised Learning with Graphs: Covariance Based Superpixels for Hyperspectral Image Classification*; IGARSS: Yokohama, Japan, 2019; pp. 592–595.
47. Zhu, X.; Tuia, D.; Mou, L.; Xia, G.S.; Zhang, L.; Xu, F.; Fraundorfer, F. Deep learning in remote sensing: A comprehensive review and list of resources. *IEEE Geosci. Remote Sens. Mag.* **2017**, *5*, 8–36. [[CrossRef](#)]
48. 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. [[CrossRef](#)]
49. Ding, C.; Li, Y.; Xia, Y.; Wei, W.; Zhang, L.; Zhang, Y. Convolutional neural networks based hyperspectral image classification method with adaptive kernels. *Remote Sens.* **2017**, *9*, 618. [[CrossRef](#)]
50. Chen, Y.; Huang, L.; Zhu, L.; Yokoya, N.; Jia, X. Fine-Grained Classification of Hyperspectral Imagery Based on Deep Learning. *Remote Sens.* **2019**, *11*, 2690. [[CrossRef](#)]
51. Sellami, A.; Farah, M.; Farah, I.R.; Solaiman, B. Hyperspectral imagery classification based on semi-supervised 3-D deep neural network and adaptive band selection. *Expert Syst. Appl.* **2019**, *129*, 246–259. [[CrossRef](#)]
52. Girvan, M.; Newman, M. Community structure in social and biological networks. *Proc. Natl. Acad. Sci. USA* **2002**, *99*, 7821–7826. [[CrossRef](#)]
53. Fortunato, S. Community detection in graphs. *Phys. Rep.* **2010**, *486*, 75–174. [[CrossRef](#)]
54. Li, X.; Wu, X.; Xu, S.; Qing, S.; Chang, P.-C. A novel complex network community detection approach using discrete particle swarm optimization with particle diversity and mutation. *Appl. Soft Comput.* **2019**, *81*, 105476. [[CrossRef](#)]
55. Wu, F.; Huberman, B. Finding communities in linear time: A physics approach. *Euro. Phys. J. B* **2004**, *38*, 331–338. [[CrossRef](#)]
56. Osaba, E.; Del Ser, J.; Camacho, D.; Bilbao, M.; Yang, X. Community detection in networks using bio-inspired optimization: Latest developments, new results and perspectives with a selection of recent meta-heuristics. *Appl. Soft Comput.* **2020**, *87*, 106010. [[CrossRef](#)]
57. Gou, J.; Zhan, Y.; Rao, Y.; Shen, X.; Wang, X.; He, W. Improved pseudo nearest neighbor classification. *Knowl. Based Syst.* **2014**, *70*, 361–375. [[CrossRef](#)]
58. Kang, X.; Li, S.; Benediktsson, J. Spectral–spatial hyperspectral image classification with edge-preserving filtering. *IEEE Trans. Geosci. Remote Sens.* **2014**, *52*, 2666–2677. [[CrossRef](#)]
59. Kang, X.; Li, S.; Benediktsson, J. Feature extraction of hyperspectral images with image fusion and recursive filtering. *IEEE Trans. Geosci. Remote Sens.* **2014**, *52*, 3742–3752. [[CrossRef](#)]
60. Li, J.; Bioucas-Dias, J.; Plaza, A. Exploiting spatial information in semi-supervised hyperspectral image segmentation. In Proceedings of the 2010 2nd Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing, Reykjavik, Iceland, 14–16 June 2010; pp. 1–4.
61. Sun, L.; Wu, Z.; Liu, J.; Xiao, L.; Wei, Z. Supervised spectral–spatial hyperspectral image classification with weighted Markov random fields. *IEEE Trans. Geosci. Remote Sens.* **2015**, *53*, 1490–1503. [[CrossRef](#)]
62. Jiang, J.; Ma, J.; Chen, C.; Wang, Z.; Cai, Z.; Wang, L. SuperPCA: A Superpixel-wise PCA approach for unsupervised feature extraction of hyperspectral imagery. *IEEE Trans. Geosci. Remote Sens.* **2018**, *56*, 4581–4593. [[CrossRef](#)]
63. Xie, F.; Lei, C.; Yang, J.; Jin, C. An effective classification scheme for hyperspectral image based on superpixel and discontinuity preserving relaxation. *Remote Sens.* **2019**, *11*, 1149. [[CrossRef](#)]

