



Article

Hyperspectral Image Classification Promotion Using Clustering Inspired Active Learning

Chen Ding ^{1,2,3} , Mengmeng Zheng ^{1,2,3} , Feixiong Chen ^{1,2,3} , Yuankun Zhang ^{1,2,3}, Xusi Zhuang ^{1,2,3}, Enquan Fan ^{1,2,3}, Dushi Wen ^{1,2,3}, Lei Zhang ^{4,5,*}, Wei Wei ^{4,5} and Yanning Zhang ^{4,5}

- ¹ School of Computer Science and Technology, Xi'an University of Posts and Telecommunications, Xi'an 710121, China; dingchen@xupt.edu.cn (C.D.); zmm@stu.xupt.edu.cn (M.Z.); FeixiongChen@stu.xupt.edu.cn (F.C.); qwrty2@stu.xupt.edu.cn (Y.Z.); 04181171@stu.xupt.edu.cn (X.Z.); enquanfanxupt@stu.xupt.edu.cn (E.F.); wds3721@xupt.edu.cn (D.W.)
- ² Shaanxi Key Laboratory of Network Data Analysis and Intelligent Processing, Xi'an 710121, China
- ³ Xi'an Key Laboratory of Big Data and Intelligent Computing, Xi'an 710121, China
- ⁴ Shaanxi Key Lab of Speech & Image Information Processing (SAIIP), School of Computer Science and Engineering, Northwestern Polytechnical University, Xi'an 710129, China; weiweinwpu@nwpu.edu.cn (W.W.); ynzhang@nwpu.edu.cn (Y.Z.)
- ⁵ National Engineering Laboratory for Integrated Aero-Space-Ground-Ocean Big Data Application Technology, Xi'an 710129, China
- * Correspondence: nwpuzhanglei@nwpu.edu.cn; Tel.: +86-187-2922-5124

Abstract: Deep neural networks (DNNs) have promoted much of the recent progress in hyperspectral image (HSI) classification, which depends on extensive labeled samples and deep network structure and has achieved surprisingly good generalization capacity. However, due to the expensive labeling cost, the labeled samples are scarce in most practice cases, which causes these DNN-based methods to be prone to over-fitting and influences the classification result. To mitigate this problem, we present a clustering-inspired active learning method for enhancing the HSI classification result, which mainly contributes to two aspects. On one hand, the modified clustering by fast search and find of peaks clustering method is utilized to select highly informative and diverse samples from unlabeled samples in the candidate set for manual labeling, which empowers us to appropriately augment the limited training set (i.e., labeled samples) and thus improves the generalization capacity of the baseline DNN model. On the other hand, another K-means clustering-based pseudo-labeling scheme is utilized to pre-train the DNN model with all samples in the candidate set. By doing this, the pre-trained model can be effectively generalized to unlabeled samples in the testing set after being fine tuned-based on the augmented training set. The experiment accuracies on two benchmark HSI datasets show the effectiveness of the proposed method.

Keywords: hyperspectral image classification; active learning; candidate set; pseudo-labels; clustering



Citation: Ding, C.; Zheng, M.; Chen, F.; Zhang, Y.; Zhuang, X.; Fan, E.; Wen, D.; Zhang, L.; Wei, W.; Zhang, Y. Hyperspectral Image Classification Promotion Using Clustering Inspired Active Learning. *Remote Sens.* **2022**, *14*, 596. <https://doi.org/10.3390/rs14030596>

Academic Editor: Edoardo Pasolli

Received: 16 December 2021

Accepted: 24 January 2022

Published: 26 January 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

A hyperspectral image (HSI) contains not only spatial information but also abundant spectral information. The substances, which are difficultly distinguished in natural images can be easily recognized in hyperspectral imagery. As a result, HSIs have been widely applied in resource exploration, mineral detection, environmental investigation and lesion detection, etc. [1–5].

HSI classification is an essential HSI application which focuses on assigning each pixel a unique class label. To date, a large number of HSI classification methods have been proposed from different perspectives, depending on the HSI classification methods whether using deep learning-based methods to obtain HSI features and classification results, the HSI classification methods can be roughly divided into the non-deep learning-based method and the deep learning-based method.

The non-deep learning-based method has been utilized for HSI classification methods for decades. Within the non-deep learning-based method, the feature extraction module and the classifier module are always independently modeled. In addition, pre-defined criteria are utilized within the shallow-structure feature extraction module to generate the desired features. The existing non-deep learning-based methods usually include spectral matching-based methods [6,7], statistic model-based methods [8,9], kernel-based methods [10–12], sparse representation-based methods [13] and spatial-spectral information-based methods [14]. Though these methods show advantages in some applications, the features via non-deep learning-based methods prevent the accuracy in some HSI classification tasks.

The deep learning-based method provides a new way to generate deep structure-related features. In addition, the generated feature can fit the classifier well, because the feature extraction module and the classifier module are naturally integrated into one framework within the deep learning-based method. As a result, the deep learning-based method obtains better HSI performance compared with the non-deep learning-based method and dominates the recent HSI classification community [15–23], i.e., light-weight spectral-spatial feature extraction and fusion network [16], spectral-spatial kernel generation network [17], attention aided CNNs [18], spectral-spatial information based Resnet [19], adaptive hybrid attention network [20], residual spectral-spatial attention network [21] and spectral-spatial based deep belief network [23]. In addition to the above methods, other different deep learning-based methods have been proposed. Hu et al. first utilized convolutional neural networks (CNNs) [24] for HSI classification based on spectral information only. Work [25] proposed a two-channel deep convolutional neural network (2D-CNN). Within 2D-CNN, it learns the spectral and spatial feature separately from those two channels first, and then concatenates and obtains spectral-spatial features for classification via a fully connected layer. In [26], the three-channel deep convolutional neural network (3D-CNN) was proposed for HSI classification, which utilized a 3D data cube (containing both spectral and spatial information) as the input and achieved better results. In addition to the above methods, the pre-learned convolutional kernels based deep learning methods were also used in HSI classification tasks, such as PCA-Net [27], MCFSDP-Net [28] and K-means Net [29].

Although the deep learning-based method obtains good HSI classification results, one important premise behind this method is that a large number of labeled training samples can be provided. However, it is laborious and difficult to obtain large amounts of labeled pixels within HSI [30]. Instead, only a small amount of labeled data (termed as small sample problem in the following) can be provided in applications, which easily leads to over-fitting when training deep neural networks and thus degrades the classification performance [31]. As a result, how to address the problems has become the research focus in recent years. A pixel-pair method was proposed to solve the small sample HSI classification problem, which constructed a new data pair combination to increase the number of training samples [32]. Limited to the number of training samples, a self-taught feature learning-based method was proposed to solve the HSI classification task [33]. In addition to the above deep learning feature-based methods, residual networks [34], dense convolutional networks [35] and capsule networks [36] have been utilized in small sample HSI classification. Recently, the domain adaption-based method [37], the Siamese CNN-based method [38] and the attention combined parallel network-based method [39] were also proposed to address the HSI classification with limited samples, which also improved the accuracy of the small sample HSI classification result. In addition, for the increasing sample quantity-based methods, deep convolutional GAN is well suited for data processing, which can generate fake samples to increase the number of training samples [40,41]. In [42], generative adversarial networks (GANs) were explored for HSI classification for the first time, containing two CNN frameworks: one CNN framework is utilized to discriminate the inputs, and another CNN framework is utilized to generate so-called fake inputs. The aforementioned CNNs are trained together, the generated fake inputs are as real as possible, and the discriminative CNN tries to classify the real and

fake inputs to solve the small sample HSI classification tasks. Although this method can enhance HSI classification accuracy with limited samples via the generative capacity of GANs, the quality of the generated samples is often ignored, which limits the improvement of the classification result.

This paper presents a cluster-inspired active learning method for HSI classification with limited labeled samples, which mainly contributes to two aspects. Firstly, the modified clustering by fast search and find of peaks (MCFSFDP) clustering method is utilized to select highly informative and diverse samples from unlabeled samples in the candidate set for manually labeling by an expert, which empowers us to appropriately augment the limited training set (i.e., labeled samples) and thus improve the generalization capacity of the baseline DNN model. Secondly, another K-means clustering-based pseudo-labeling scheme is utilized to pre-train the DNN model with the unlabeled samples in the candidate set. By doing this, the pre-trained model can be effectively generalized to unlabeled samples in the testing set after being fine-tuned based on the augmented training set.

This paper is organized as follows. In Section 2, the proposed method is described in detail, including data pre-processing, actively selecting core samples from the candidate set via MCFSFDP, the pre-trained DNN model via pseudo-labeling of unlabeled samples in candidate set generated via K-means, and network training and testing. In Sections 3 and 4, the results and discussion are presented. In Section 5, the conclusions of this paper are summarized.

2. The Proposed Method

The cluster inspired active learning method includes four major steps: (1) data pre-processing, which extracts the spectral information of each pixel as the sample and divides all the samples into the training set, candidate set and the testing set; (2) actively selecting core samples from the candidate set via MCFSFDP—the effective MCFSFDP clustering method is utilized to actively select core samples from unlabeled samples in the candidate set for manually labeling; (3) the K-means clustering-based pseudo-labeling scheme is utilized to pre-train the DNN model with samples in candidate set; and (4) fine-tuning and testing, using core samples and small samples as new augmented training samples to fine-tune the network and obtain the final classification result of the testing samples. The flowchart of our proposed method is shown in Figure 1.

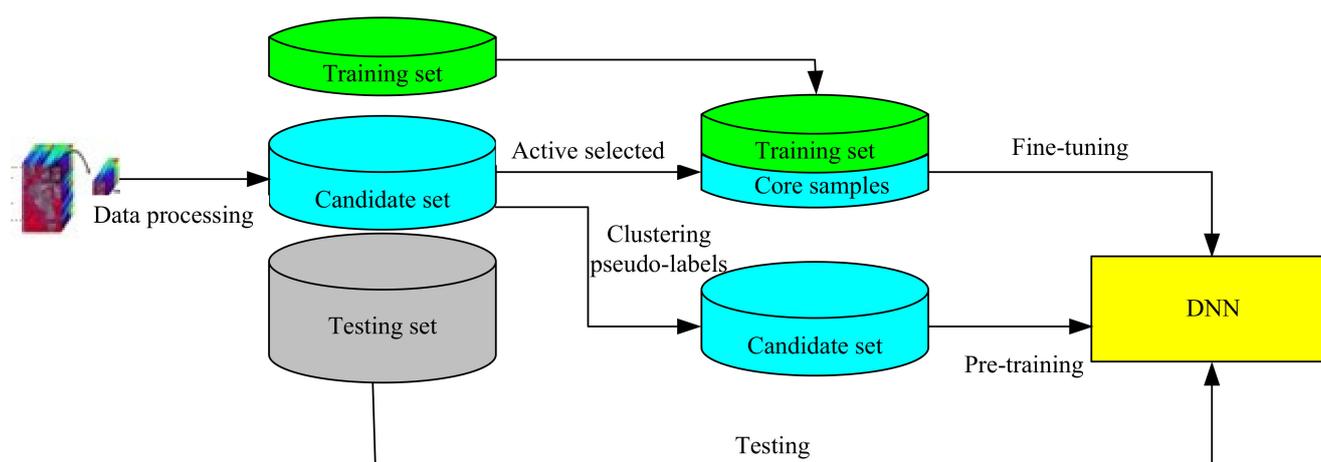


Figure 1. The flow chart of the proposed method.

2.1. Data Pre-Processing

In this paper, the HSI used in the classification task is denoted as R . An HSI consists of 3D data; we only use the spectral information of each pixel as the sample. We randomly select M pixels from R as limited training samples; in other words, the quantity of the small sample is denoted by M . These selected training samples include all the categories, and each

category has almost the same number of pixels. The pixel P_i includes the corresponding spectral information with a size of $h \times 1$ as the training sample. h denotes the spectral number of R . $\{P_i\}_{i=1}^M$ denotes the limited samples, and the limited samples have these manually labeled labels.

Then, we extract N pixels from R and their corresponding spectral information $\{C_j\}_{j=1}^N$ as unlabeled samples in the candidate set, N denotes the number of samples in the candidate set, i.e., the number of the candidate samples.

Finally, the rest samples are testing samples. K denotes the number of testing samples. $\{Q_u\}_{u=1}^K$ denotes the testing samples. The samples are also denoted as column vectors, with sizes of $h \times 1$ mathematically.

The samples in the testing set are all used for testing. The core samples are actively selected for labeling via the active learning method, which are selected from the candidate samples. In addition, the K-means clustering method will automatically give the samples in candidate set pseudo-labels for the network pre-training. Here, M plus N is almost equal to K . The samples in the training set, the candidate set and the testing set are not overlapping.

The sample is extracted from R is shown in Figure 2.

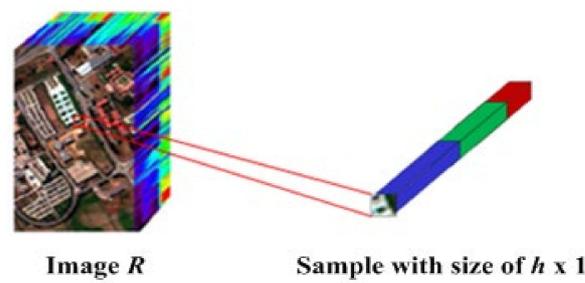


Figure 2. The sample is extracted from image R .

2.2. Actively Selecting Core Samples via MCFSDP

To actively select the core samples for manually labeling from unlabeled candidate samples, the clustering-based method may be suitable. In our opinion, clustering by fast search and find of peaks (CFSFDP) [43] is a representative method. The idea of this method is that “the cluster centers are determined as those points that not only have higher density than their neighbors, but also keep a certain distance from the point with higher density than them”. In this clustering method, the two thresholds, i.e., distance and density, are important to determine the cluster centers. The points which have higher distances and densities at the same time can be determined as the cluster centers.

In our opinion, CFSFDP is useful in actively selecting the cluster centers and clustering process; however, the wild points (i.e., the inter-class points) are important and difficult to distinguish. To solve this problem, the effective clustering method based on modified clustering by fast search and find of peaks (MCFSDP), is proposed to actively select core samples by choosing the adaptive distance threshold [28]. The MCFSDP algorithm is similar to the CFSFDP algorithm in [43], the class center must have two characters, the first character is “a higher density than their neighbors” and the second is “a relatively large distance from points with higher densities”. Different from the CFSFDP, the MCFSDP chooses the class centers only by larger distance, which can effectively acquire the cluster centers and the wild points and enhance the quality of the selected samples. The details of the proposed method are as follows.

The samples $\{C_j\}_{j=1}^N$ in candidate set are used for actively selected core samples via clustering based active learning method; for simplicity, each sample C_j in candidate set is denoted as point j , which is actually a column vector. For each point j , we calculate the local density ρ_j and distance δ_j from the point with higher density; if point j has the highest density, the largest distance between j and the other points is denoted as δ_j .

The local density ρ_j of point j is given in Formula (1):

$$\rho_j = \sum_k \chi(d_{jk} - d_c) \quad (1)$$

Formula (1) represents the number of samples around the point j in a threshold radius d_c . The values of δ_j and ρ_j are depended on the Euclidean distance d_{jk} , d_{jk} is determined by any pair of the point j and point k . Where $\chi(d_{jk} - d_c) = 1$ if $d_{jk} - d_c < 0$, otherwise, $\chi(d_{jk} - d_c) = 0$, here, d_c is considered as a cut-off distance. ρ_j denotes the number of points which in the radius d_c and j is the center point.

δ_j is the minimum distance between j and any other points with higher density, which is shown in Formula (2):

$$\delta_j = \min_{k:\rho_k > \rho_j} (d_{jk}) \quad (2)$$

where ρ_k denotes the local density of k . For the point with maximum local density, we usually take $\delta_j = \max_k(d_{jk})$. δ_j is much larger than the typical nearest neighbor distance only for points that are local or global maxima in the density. The cluster centers are recognized as points for which the value of δ_j is anomalously large and the value of ρ_j is higher than a value density at the same time.

The distance and density of each point are directly shown in the decision graph. We provided the decision graph of samples in candidate set with a size of 200×1 for the Indian Pines dataset [44], as shown in Figure 3. The Indian Pines dataset is often used in the hyperspectral image classification task, which was gathered by AVIRIS sensor over the Indian Pines test site in North-western Indiana and consists of 145×145 pixels and 224 spectral reflectance bands in the wavelength range 0.4–2.5 μm .

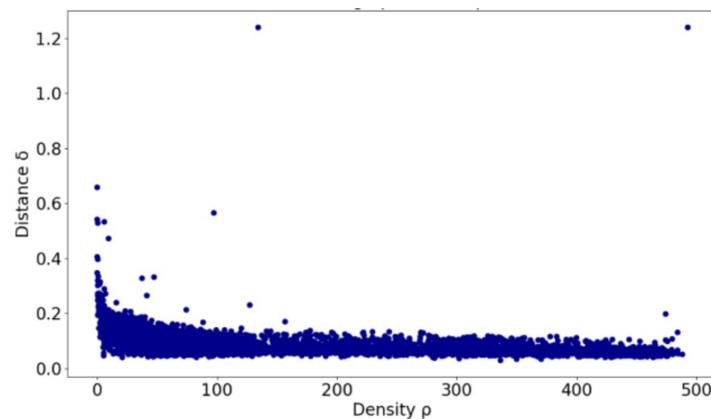


Figure 3. Decision graph of samples in candidate set with a size of 200×1 for Indian Pines.

In the threshold determining step, the MCFSFDP is different from CFSFDP [43], the MCFSFDP is used to select core samples for manually labeling. The distance δ is considered as the only threshold from the decision graph to select samples. This operation can select not only the cluster centers but also the wild points to enhance the quality of samples for increasing the classification result. Because the wild points are in the boundary of any pair of two clusters, which are usually difficult to distinguish, training this type of sample is useful for improving the classification result.

For selecting the core samples adaptively, we should select an optimal distance threshold value δ_A .

$$n_v = f(\delta_v) \quad (3)$$

In Formula (3), δ_v denotes the distance, which contains points, and $f(\delta_v)$ denotes the mapping relationship of the number n_v of points whose distances are larger than or equal to δ_v , as shown in Figure 4a.

$$c_v = [n_{v+1} - n_v] / (\delta_{v+1} - \delta_v) \tag{4}$$

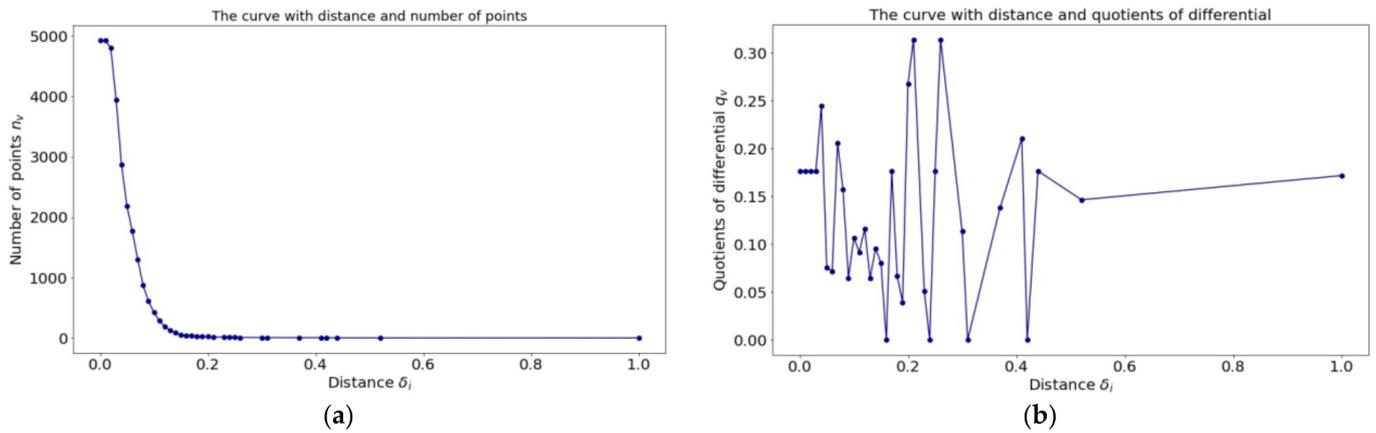


Figure 4. The curves for determining the adaptive distance δ_A in the candidate set of Indian Pines dataset with sample size of 200×1 . (a) shows the curve of the point-number over distance δ_v ; (b) gives the curve of the quotients of differential over distance δ_v .

In Formula (4), where $\delta_{v+1} \geq \delta_v$, c_v denotes the differential of n_v . Formula (5) denotes the variation quantity of the number points with δ_v , as shown in Figure 4b. Formula (4) is the intermediate result of Formulas (3) and (5).

$$q_v = |c_v / c_{v+1}| \tag{5}$$

In the MCFSFDP method, the adaptive distance threshold is denoted as δ_A , and the points whose distance are larger than δ_A are automatically selected as core samples. δ_v is an important point that must ensure that the number n_v and n_{v+1} of points are stable, and at the same time, that the value q_v is larger than the value q_{v+1} . At this point, δ_v is selected as the adaptive distance δ_A .

In the Indian Pines dataset, as can be seen from Figure 4a, we can find the distance range (0.15–0.17), and the n_v begins to approach stability. As can be seen from Figure 4b, c_v with the distance value δ_v in range (0.15–0.17) has a local maxima of 0.15. Therefore, 0.15 is considered as the adaptive distance δ_A in the Indian Pines dataset.

With the adaptive distance δ_A , the points j with the distance value $\delta_j > \delta_A$ are adaptively chosen as core samples for manual labeling.

Then, the labeled core samples are added into training samples to form the augmented training set. The number of core samples is denoted as T , and the number of training samples after expansion is $M + T$. $\{B_g\}_{g=1}^{M+T}$ denotes the final training dataset.

2.3. K-Means Clustering-Based Pseudo-Labeling Scheme

After selecting the core samples via MCFSFDP, we use K-means clustering to obtain the pseudo-labels of the samples $\{C_j\}_{j=1}^N$ in candidate set. The steps are as follows:

Step 1: Randomly selecting k samples from $\{C_j\}_{j=1}^N$ as the initial cluster centers, i.e., $\mu_1, \dots, \mu_f, \dots, \mu_k$.

3. Experiments and Analysis

To validate the feasibility and effectiveness of the proposed method, two HSI datasets were used in the experiments. In this section, we firstly introduce the datasets. Secondly, the experimental parameter settings are illustrated. Finally, ablation experiments and comparative experiments are performed to show the HSI classification results of the proposed method.

3.1. Datasets

In this paper, two widely used public HSI image datasets were adopted in our experiments.

Dataset 1: In order to evaluate the proposed method, the first dataset was the Indian Pines image, which was imaged by the Airborne Visual Infrared Imaging Spectrometer (AVIRIS) [44], as shown in Figure 6a. The ground truth is shown in Figure 6b. The size of this image is 145×145 pixels with 224 spectral bands, and the wavelength ranges from 0.4 to $2.5 \mu\text{m}$. Among the pixels, only 10,249 pixels are feature pixels, and the remaining 10,776 pixels are background pixels. For the exact purpose of eliminating the bands that cannot be reflected by water, the number of bands was reduced to 200. In the actual classification, since background pixels need to be eliminated, there were 16 categories in total. Each category of image samples number is given in Table 1.

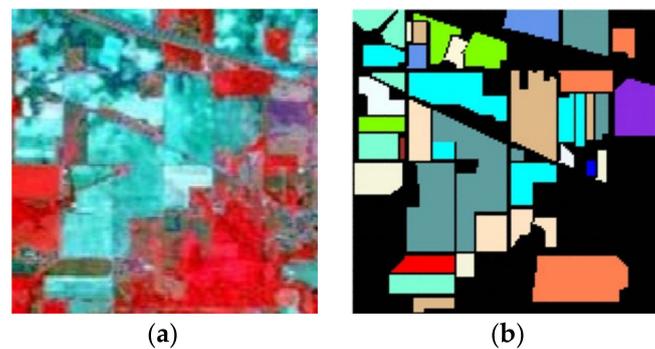


Figure 6. The Indian Pines image in Dataset 1. (a) shows the composite image; (b) shows the ground truth of the Indian Pines dataset, where the black area denotes the unlabeled pixels.

Table 1. Ground truth of classes and number of their respective samples in the Indian Pines scene.

Number	Class		Samples		
	Classes	Total	Training	Candidate	Testing
1	Alfalfa	46	6	17	23
2	Corn-notill	1428	26	688	714
3	Corn-mintill	830	12	403	415
4	Corn	237	7	112	118
5	Grass-pasture	483	8	234	241
6	Grass-trees	730	16	349	365
7	Grass-pasture-mowed	28	5	9	14
8	Hay-windrowed	478	13	226	239
9	Oats	20	5	5	10
10	Soybean-notill	972	12	474	486
11	Soybean-mintill	2455	38	1190	1227
12	Soybean-clean	593	11	286	296
13	Wheat	205	7	96	102
14	Woods	1265	15	618	632
15	Building-Grass-Trees	386	12	181	193
16	Stone-Steel-Towers	93	7	40	46
	Total	10,249	200	4928	5121

The samples in training set could be regarded as limited samples with labels. The samples in candidate set were used for choosing core samples, and the core samples are added into the training samples as a new augmented training set. The samples in candidate set were also used for pre-training the DNNs with their pseudo-labels. The samples in testing set were used for evaluating the effect of the proposed method.

Dataset 2: The second dataset was the Salinas image [44], which was imaged in Salinas Valley in California through AVIRIS as well, as shown in Figure 7a. The ground truth is shown in Figure 7b. Differing from the Indian Pines image, whose spatial resolution is 20 m, its spatial resolution reached 3.7 m. As shown in Figure 6, the size of this image is 512×217 pixels, with 224 spectral bands. The number of bands was reduced to 204 after eliminating the low signal-to-noise-ratio (SNR) bands. Among them, 54,129 samples were used for training and testing in total. The details of each category of samples are given in Table 2. This dataset was used to test the feasibility and effectiveness of the proposed approach for classification.

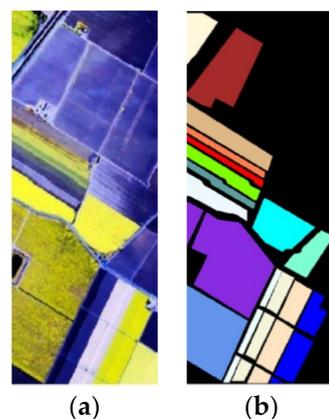


Figure 7. The Salinas scene in Dataset 2. (a) shows the composite image; (b) shows the ground truth of the Salinas Dataset, where the black area denotes the unlabeled pixels.

Table 2. Ground truth of classes and number of their respective samples in the Salinas scene.

Number	Class	Total	Samples		
	Classes		Training	Candidate	Testing
1	Broccoli_green_weeds_1	2009	11	994	1004
2	Broccoli_green_weeds_2	3726	16	1847	1863
3	Fallow	1976	12	976	988
4	Fallow_rough_plow	1394	10	687	697
5	Fallow_smooth	2678	11	1328	1339
6	Stubble	3959	19	1961	1979
7	Celery	3579	13	1777	1789
8	Grapes_untrained	11,271	14	5622	5635
9	Soil_vinyard_develop	6203	15	3087	3101
10	Corn_senesced_green_weeds	3278	10	1629	1639
11	Lettuce_romaine_4wk	1068	12	522	534
12	Lettuce_romaine_5wk	1927	13	951	963
13	Lettuce_romaine_6wk	916	10	448	458
14	Lettuce_romaine_7wk	1070	11	524	535
15	Vinyard_untrained	7268	15	3621	3634
16	Vinyard_vertical_trellis	1807	10	894	903
	Total	54,129	200	26,868	27,061

3.2. Experimental Parameter Settings

In the experiment, the samples were randomly selected from the HSI dataset. The training sample set includes 200 samples. For utilizing the effective cluster-inspired active learning method, the samples in candidate set were used to obtain the core samples through the MCFSFDP algorithm for manual labeling, and the pseudo-labels of the samples in candidate set were generated through the K-means algorithm for the DNN's pre-training. The number of cluster centers was set to 10, 20, . . . , 100.

In the experiment, as shown in Figure 5, the DNN framework used three fully connected layers and one soft-max layer. In our algorithm, three fully connected layers, namely, hidden layers, all adopted Leaky ReLU as the activation function. The number of neuron nodes in the three hidden layers was 512, 2048 and 1024, respectively. The learning rate was 0.0001. The batch size was designed as 256.

The code was run on a computer with Intel i9-11900K, NVIDIA 3060 GPU \times 2, 128 GB Memory, and 1TB SSD.

3.3. Experimental Results

3.3.1. Effectiveness of the Core Samples Actively Selected via MCFSFDP

The effectiveness of the core samples generated by the actively selected method is worthy to be verified. To verify the influence of core samples selected based active learning method in classification, we compared the accuracy of randomly selected samples based active learning method with the accuracy of actively selected core sample-based method, the number of randomly selected samples from candidate set being same as the core samples. The testing accuracy via the training samples with randomly selected samples and training samples with core samples via our proposed MCFSFDP in Dataset 1 is shown in Table 3. Additionally, the testing accuracy for Dataset 2 is shown in Table 4.

Table 3. The testing result of randomly selected samples and core samples via MCFSFDP in Dataset 1.

Class	The Adaptive Distance Threshold	The Number of Selected Core Samples	Testing Accuracy (%)	
			Randomly Selected Samples	Core Samples
1	0.15	55	39.1	65.2
2			51.8	57.1
3			47.0	56.1
4			41.5	47.5
5			78.4	66.0
6			94.8	93.2
7			71.4	85.7
8			95.0	90.0
9			20.0	20.0
10			59.5	52.3
11			73.1	77.0
12			32.8	40.2
13			100.0	99.0
14			75.5	80.1
15			35.2	32.6
16			95.7	89.1
OA (%)			65.9	67.8
AA (%)			63.2	65.7
Kappa			61.1	64.2

Table 4. The testing result of randomly selected samples and core samples via MCFSFDP in Dataset 2.

Class	The Adaptive Distance Threshold	The Number of Selected Core Samples	Testing Accuracy (%)	
			Randomly Selected Samples	Core Samples
1	0.12	40	99.0	95.0
2			97.0	99.4
3			45.1	66.5
4			99.7	99.6
5			78.3	94.2
6			99.6	99.1
7			99.2	98.3
8			88.2	82.3
9			94.5	96.8
10			67.5	74.6
11			91.6	99.1
12			97.0	97.0
13			99.0	99.0
14			90.8	90.5
15			45.8	55.8
16			88.0	84.6
OA (%)			83.1	85.6
AA (%)			86.3	89.5
Kappa			81.3	84.0

In the Indian Pines dataset, the adaptive distance threshold is calculated as 0.15, and we obtain 55 core samples via the MCFSFDP algorithm. The curve for determining the adaptive distance is shown in Figure 4. The adaptive distance is 0.12, and the number of core samples is 40 in Dataset 2. The curve for determining the adaptive distance is shown in Figure 8.

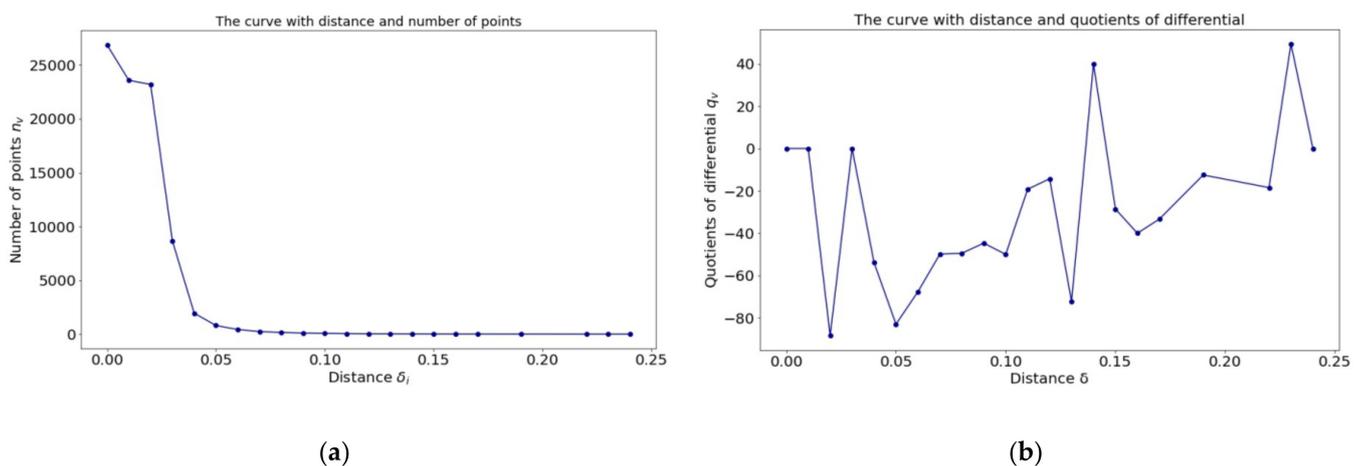


Figure 8. The curve for determining the adaptive distance in the Salinas scene dataset. (a) shows the curve of the point-number over distance δ_v ; (b) gives the curve of the quotients of differential over distance δ_v .

As can be seen from Tables 3 and 4, the testing result for small samples with core samples is higher than the result for small samples with randomly selected samples. Specifically in Table 4, the overall accuracy (OA) of small samples with core samples is shown to be more than 2% greater than the overall accuracy (OA) of randomly selected samples. Therefore, using the actively selected core samples via MCFSFDP to train the BP neural network can enhance the testing accuracy of the small sample HSI classification.

Additionally, the actively selected core sample-based method not only enhances the quantity but also the quality of the training samples.

The other testing results in the two datasets, i.e., the accuracy of each class, average accuracy (AA) and Kappa, which are also shown in Tables 3 and 4.

3.3.2. Effectiveness of the Proposed Method-Based on Actively Selected Core Samples

Through the above experiments, we have demonstrated the effectiveness of the actively selected core samples method in small sample HSI classification. The classification results prove the effectiveness of the proposed method based on actively selected samples on two datasets.

In the two datasets, the original training samples set, which has 200 samples with their labels, is used for training the BP neural network, while the testing samples set is used for testing the network. In the Indian Pines dataset, the adaptive distance threshold is calculated as 0.15, and we obtain 55 core samples via the MCFSFDP algorithm. These core samples are added into the training samples set and we utilize the new augmented training dataset to train the network. The testing result of the original training samples set and the augmented training samples set with core samples in Dataset 1 is shown in Table 5, while the curve for determining the adaptive distance is shown in Figure 4.

Table 5. The testing result of original training samples set and training samples set with core samples in Dataset 1.

Class	Testing Accuracy (%)	
	Original Training Samples Set	Training Samples Set with Core Samples
1	47.8	65.2
2	47.3	57.1
3	49.6	56.1
4	44.1	47.5
5	29.9	66.0
6	92.6	93.2
7	85.7	85.7
8	93.3	90.0
9	20.0	20.0
10	30.2	52.3
11	69.4	77.0
12	29.4	40.2
13	100.0	99.0
14	74.5	80.1
15	31.1	32.6
16	93.5	89.1
OA (%)	58.9	67.8
AA (%)	58.5	65.7
Kappa	52.8	64.2

The testing accuracy for the Salinas dataset is shown in Table 6, and the curve for determining the adaptive distance is shown in Figure 8. The adaptive distance is 0.12, and the number of core samples is 40 in dataset 2, which can also be seen in Tables 3 and 4.

Table 6. The testing result of original training samples set and training samples set with core samples in Dataset 2.

Class	Testing Accuracy (%)	
	Original Training Samples Set	Training Samples Set with Core Samples
1	99.1	95.0
2	97.5	99.4
3	55.9	66.5
4	99.7	99.6
5	73.0	94.2
6	99.6	99.1
7	99.2	98.3
8	90.1	82.3
9	96.0	96.8
10	64.2	74.6
11	94.2	99.1
12	98.4	97.0
13	98.7	99.0
14	90.5	90.5
15	30.0	55.8
16	83.5	84.6
OA (%)	81.7	85.6
AA (%)	56.6	89.5
Kappa	80.0	84.0

In Table 5, the testing accuracy (OA) with the original training samples set for Dataset 1 is 58.9% after 13,000 training epochs. In contrast to this, the testing accuracy (OA) with the augmented training samples set with core samples in Dataset 1 is 67.8% after 13,000 training epochs. According to the data, the testing accuracy (OA) with the original training samples set is lower than the testing accuracy with the augmented training samples set with core samples.

Additionally, in Table 6, the maximal testing accuracy (OA) with the original training samples set in Dataset 2 is 81.7% after 11,000 training epochs. In contrast to this, the testing accuracy (OA) with the augmented training samples set with core samples for Dataset 2 is 85.6% after 11,000 training epochs. According to the data, the testing accuracy (OA) with the original training samples set is also lower than the testing accuracy (OA) with the training samples set with core samples. Consequently, obtaining the core samples via MCFSFDP added to the training samples set, which is demonstrated to enhance the small sample HSI classification accuracy in Dataset 1 and Dataset 2.

The other testing results in the two datasets, i.e., the accuracy of each class, average accuracy (AA) and Kappa, which are also shown in Tables 5 and 6.

3.3.3. Effectiveness of Pre-Training by Testing Samples with Pseudo-Labels

Through the above experiments, we have proved the effectiveness of active learning in small sample HSI classification. In order to demonstrate the effectiveness of the proposed method of pre-training using candidate samples with pseudo-labels via clustering combined with adaptive active learning, we labeled the pseudo-labels for the candidate samples via the K-means algorithm and utilized these data to pre-train the BP neural network. Then, the training samples set with core samples is used for fine-tuning the network.

To determine the appropriate number of clusters for pseudo-labels, we observe the testing accuracy of the proposed method with a different number of clusters after 13,000 training epochs in Dataset 1. The testing accuracy of the proposed method with different numbers of clusters after 13,000 training epochs in Dataset 1 is shown in Table 7. The testing accuracy of the proposed method with different numbers of clusters after 11,000 training epochs in Dataset 2 is shown in Table 8.

Table 7. The testing accuracy of the proposed method with different numbers of clusters in Dataset 1.

The Number of Clusters	10	20	30	40	50	60	70	80	90	100
Testing Accuracy OA (%)	63.7	64.6	65.3	66.2	68.9	66.1	65.8	65.7	66.5	66.6

Table 8. The testing accuracy of the proposed method with different numbers of clusters in Dataset 2.

The Number of Clusters	10	20	30	40	50	60	70	80	90	100
Testing Accuracy OA (%)	85.5	85.9	86.0	85.9	85.9	85.8	85.9	86.8	86.1	85.4

In Table 7, the maximal testing accuracy (68.9%) of the proposed method for Dataset 1 shows that the number of cluster centers is 50 when using 13,000 training epochs. Compared with the value of Table 5, the testing accuracy of the proposed method is higher than that of the original training samples set (58.9%) and the training samples set with core samples (67.8%). According to the data, compared with the method of only adaptive active learning, the testing accuracy of the proposed method significantly improved. Additionally, in Table 8, the maximal testing accuracy (86.8%) of the proposed method for Dataset 2 shows that the number of cluster centers is 80 when using 11,000 training epochs. Compared with the value of Table 6, the testing accuracy of the proposed method is higher than that of the original training samples set (81.7%) and the training samples set with core samples (85.6%).

In Tables 7 and 8, due to the different distributions of samples in the two datasets, the number of clusters in the Indian Pines dataset and the Salinas dataset are different, which choose 50 clusters and 80 clusters, respectively. Consequently, the proposed cluster-inspired active learning method is demonstrated to enhance the small sample HSI classification accuracy and has a better effect than the above method in Tables 3–6 on Dataset 1 and Dataset 2.

3.3.4. The Proposed Method Compared with the Other Methods

In these experiments, our method is compared with other methods, including random based active learning method, K-means based active learning method, minimum probability-based active learning method, CFSFDP based active learning method [43] and our MCFSFDP based active learning method [28] and the proposed cluster inspired active learning method.

Specifically speaking, K-means selected sample-based method utilizes the K-means algorithm to extract samples. Minimum probability-based active learning method uses n minimum probabilities of predicted samples to choose samples. CFSFDP and MCFSFDP selected sample-based methods are used to increase the number of samples. The classification effect is different through Back-Propagation neural network. The testing accuracy (OA) of these methods compared with the proposed method for Dataset 1 is shown in Table 9. The testing accuracy (OA) for Dataset 2 is shown in Table 10.

Table 9. The testing accuracy of the proposed method compared with the other methods for Dataset 1.

Dataset 1	Testing Accuracy (%)					
	Random Selected Samples	K-Means Selected Samples	Minimum Probability Selected Samples	CFSFDP Selected Samples	MCFSFDP Selected Samples	Proposed Method
OA (%)	65.9	59.6	63.9	64.4	67.8	68.9

Table 10. The testing accuracy of the proposed method compared with the other methods for Dataset 2.

Dataset 2	Testing Accuracy (%)					Proposed Method
	Random Selected Samples	K-Means Selected Samples	Minimum Probability Selected Samples	CFSFDP Selected Samples	MCFSFDP Selected Samples	
OA (%)	83.1	82.9	83.8	85.1	85.6	86.8

Through the classification results of different methods for Dataset 1 and 2, it can be seen that the testing accuracy of the proposed cluster-inspired active learning method is better than the other methods. Among them, the testing accuracy of K-means-based active learning method is lowest, and our MCFSFDP based active learning method is the second-best method.

4. Discussion

4.1. Influence of the Network Training Iterations

The experimental results in Tables 11 and 12 show that adding the core samples into the training samples set for training the network can obtain better testing accuracy than using original small samples for Dataset 1 and Dataset 2.

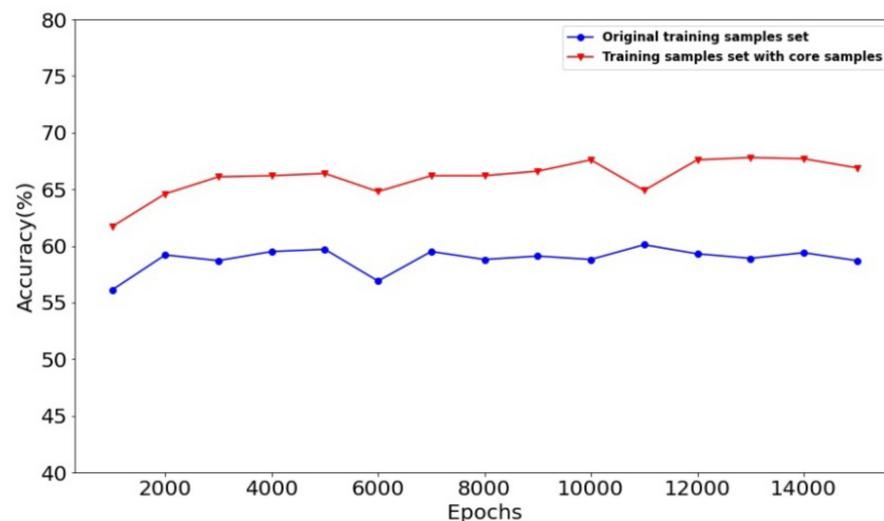
According to the data in Table 11, the number of epochs, which is 13,000, is confirmed as the best training iteration with core samples, as it obtains the testing accuracy (58.9%) in the original training samples set for Dataset 1. The testing accuracy of the training samples set with core samples is 67.8%, which is the best testing accuracy of training samples with core samples, subsequent experiments still use 13,000 epochs as the best training iteration. The best testing accuracy in the original training sample set is 60.1% with the 11,000 iterations. In addition, we choose 13,000 epochs as the iteration times in Dataset 1. The iteration influence curve is shown in Figure 9.

Table 11. The testing accuracy of original training samples set and training samples set with core samples for Dataset 1.

Dataset	Epochs	Testing Accuracy OA (%)	
		Original Training Set	Training Set with Core Samples
Indian Pines	1000	56.1	61.7
	2000	59.2	64.6
	3000	58.7	66.1
	4000	59.5	66.2
	5000	59.7	66.4
	6000	56.9	64.8
	7000	59.5	66.2
	8000	58.8	66.2
	9000	59.1	66.6
	10,000	58.8	67.6
	11,000	60.1	64.9
	12,000	59.3	67.6
	13,000	58.9	67.8
	14,000	59.4	67.7
	15,000	58.7	66.9

Table 12. The testing accuracy of original training samples set and training samples set with core samples for Dataset 2.

Dataset	Epochs	Testing Accuracy OA (%)	
		Original Training Set	Training Set with Core Samples
Salinas	1000	71.1	71.8
	2000	79.2	78.1
	3000	80.6	81.5
	4000	81.3	82.8
	5000	82.2	84.0
	6000	82.9	84.1
	7000	82.2	84.7
	8000	82.2	84.6
	9000	82.4	85.4
	10,000	81.4	85.5
	11,000	81.7	85.6
	12,000	80.9	85.5
	13,000	79.6	85.4
	14,000	79.1	85.1
	15,000	78.5	85.2

**Figure 9.** The classification accuracy with the training epochs for Dataset 1.

According to the data in Table 12, the number of epochs, which is 6000, is confirmed as the best training period for attaining the best testing accuracy (82.9%) in the original training samples set for Dataset 2. The testing accuracy of the training samples set with core samples is 84.1%, which is higher than that of the original samples set. However, the testing accuracy of training samples with core samples trained using 11,000 epochs is 85.6%, it is the best training result, and subsequent experiments use 11,000 epochs as the condition. The iteration influence curve is shown in Figure 10.

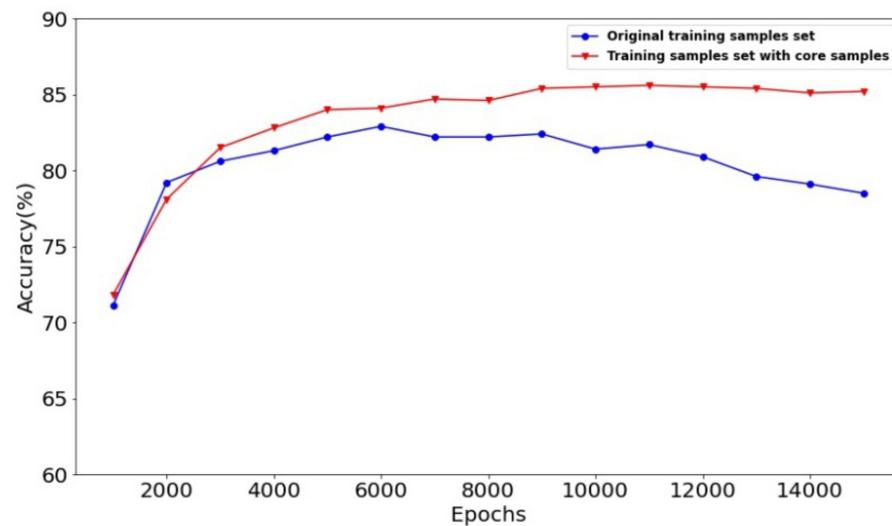


Figure 10. The classification accuracy with the training epochs for Dataset 2.

4.2. Influence of the Number of Clusters and Iterations

As can be seen from Tables 13 and 14, the testing accuracy of the proposed method is influenced by the number of clusters via K-means and the network training epochs.

In Table 13, the best accuracy is shown to be 68.9%, when we choose 13,000 iterations and 50 clusters for Dataset 1. The best testing accuracy, as shown in Table 14, is 86.8% with the best parameters, which are 11,000 iterations and 80 clusters. Therefore, Tables 9 and 10 demonstrate the two best accuracies as the final results for Dataset 1 and Dataset 2.

Table 13. The testing accuracy of the proposed method with different numbers of clusters and iterations for Dataset 1.

Dataset	The Number of Clusters		Testing Accuracy OA (%)									
	Epochs		10	20	30	40	50	60	70	80	90	100
Indian Pines	1000		60.8	61.8	63.3	64.3	66.6	63.8	62.2	62.6	62.6	63.7
	2000		62.8	62.7	63.3	62.9	67.0	65.4	64.6	64.4	64.3	65.7
	3000		62.4	63.3	62.8	63.6	68.1	63.7	64.7	65.1	66.1	65.7
	4000		61.1	64.7	64.2	65.6	67.9	63.7	64.1	65.1	66.2	66.6
	5000		62.2	64.6	64.1	64.9	67.7	64.8	62.9	65.4	66.2	66.6
	6000		62.5	62.9	64.2	63.2	67.6	65.3	65.1	65.3	66.7	66.3
	7000		63.7	64.3	64.4	64.8	67.3	64.8	65.4	65.9	66.8	67.5
	8000		65.5	63.4	64.5	65.4	67.8	66.5	65.8	64.9	65.5	66.0
	9000		63.2	62.9	57.4	65.6	67.9	64.8	64.9	64.9	65.7	67.9
	10,000		64.1	63.4	65.0	66.9	68.1	64.6	64.8	65.7	64.9	66.8
	11,000		63.0	65.2	61.9	63.8	68.4	65.9	65.7	66.3	65.9	67.2
	12,000		63.4	64.2	65.8	65.8	68.4	65.3	65.1	65.5	65.7	67.3
	13,000		63.7	64.6	65.3	66.2	68.9	66.1	65.8	65.7	66.5	66.6
	14,000		63.9	63.7	64.8	66.9	67.6	65.9	64.9	66.8	65.1	67.4
	15,000		63.4	64.4	66.1	66.8	68.3	65.3	65.2	64.8	64.1	66.8

Table 14. The testing accuracy of the proposed method with different numbers of clusters and iterations for Dataset 2.

Dataset	Epochs	The Number of Clusters		Testing Accuracy OA (%)							
		10	20	30	40	50	60	70	80	90	100
Salinas	1000	78.4	78.2	78.1	77.7	78.6	77.1	77.7	78.2	77.1	77.4
	2000	79.9	79.8	79.8	78.9	80.1	80.2	79.7	80.8	80.7	79.4
	3000	80.6	82.2	81.3	79.7	82.1	81.7	80.9	81.9	83.6	80.2
	4000	82.0	84.2	82.4	80.7	83.7	83.2	82.5	84.3	84.6	80.8
	5000	83.4	85.1	84.1	81.9	84.4	84.3	83.7	85.3	84.8	83.1
	6000	84.4	85.4	84.5	82.2	85.1	84.9	84.4	85.2	84.9	83.6
	7000	84.6	85.6	85.2	83.9	85.5	85.3	84.7	85.8	85.6	84.4
	8000	84.9	85.7	85.3	84.2	85.9	85.8	85.2	86.1	85.8	84.8
	9000	85.3	86.1	85.7	84.4	86.1	85.7	85.3	86.4	85.9	85.1
	10,000	85.6	86.2	85.8	85.1	86.3	85.8	85.4	86.7	85.9	84.8
	11,000	85.5	85.9	86.0	85.5	85.9	85.8	85.9	86.8	86.1	85.4
	12,000	85.7	86.2	85.5	85.2	86.3	85.6	86.0	86.1	86.2	85.0
	13,000	85.3	85.8	85.7	85.3	86.0	85.6	86.3	86.6	86.3	84.5
	14,000	85.6	85.5	85.8	84.7	85.9	85.2	86.3	86.4	85.9	85.8
	15,000	85.9	85.7	85.7	84.9	85.6	85.3	86.2	86.4	86.5	86.2

5. Conclusions

In this paper, we present a cluster-inspired active learning method for HSI classification, which mainly contributes to two aspects. On one hand, the modified clustering by fast search and find of peaks (MCFSFDP) clustering method is utilized to select highly informative and diverse samples from samples in candidate set for manual labeling, which empowers us to appropriately augment the limited training set (i.e., labeled samples) and thus improve the generalization capacity of the baseline DNNs model. On the other hand, another K-means clustering-based pseudo-labeling scheme is utilized to pre-train the DNN model with all samples candidate set. By doing this, the pre-trained model can be effectively generalized to testing samples after being fine-tuned based on the augmented training set. The experimental results demonstrate that the proposed method is useful in selecting core samples with high quality to expand the data and improve the small sample HSI classification accuracy effectively.

Author Contributions: Conceptualization, C.D. and L.Z.; methodology, C.D., L.Z. and W.W.; validation, Y.Z. (Yuankun Zhang), F.C., X.Z., E.F. and D.W.; formal analysis, L.Z. and W.W.; investigation, M.Z. and F.C.; resources, C.D. and D.W.; data curation, C.D. and M.Z.; writing—original draft preparation, C.D. and M.Z.; writing—review and editing, C.D., Y.Z. (Yanning Zhang), L.Z. and W.W.; supervision, Y.Z. (Yanning Zhang), W.W. and L.Z.; project administration, Y.Z. (Yuankun Zhang) and F.C.; funding acquisition, C.D., W.W. and L.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the National Natural Science Foundations of China (grant no.61901369, grant no.62071387 and grant no.62101454), the Foundation of National Engineering Laboratory for Integrated Aero-Space-Ground- Ocean Big Data Application Technology (grant no.20200203) and the National Key Research and Development Project of China (No. 2020AAA0104603).

Data Availability Statement: Not applicable.

Acknowledgments: We acknowledge AVIRIS sensor for gathering the Indian Pines test site in North-western Indian and Salinas Valley, California.

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

References

- Landgrebe, D. Hyperspectral image data analysis. *IEEE Signal Process. Mag.* **2002**, *19*, 17–28. [[CrossRef](#)]
- Shaw, G.; Manolakis, D. Signal processing for hyperspectral image exploitation. *IEEE Signal Process. Mag.* **2002**, *19*, 12–16. [[CrossRef](#)]

3. Myasnikov, E.V. Hyperspectral image segmentation using dimensionality reduction and classical segmentation approaches. *Samara Natl. Res.* **2017**, *41*, 564–572. [[CrossRef](#)]
4. Andriyanov, N.; Dementiev, V.; Gladkikh, A. Analysis of the Pattern Recognition Efficiency on Non-Optical Images. In Proceedings of the 2021 Ural Symposium on Biomedical Engineering, Radioelectronics and Information Technology (USBEREIT), Yekaterinburg, Russia, 13–14 May 2021; pp. 0319–0323.
5. Lazcano, R.; Madronal, D.; Florimbi, G.; Sancho, J.; Sanchez, S.; Leon, R.; Fabelo, H.; Ortega, S.; Torti, E.; Salvador, R.; et al. Parallel Implementations Assessment of a Spatial-Spectral Classifier for Hyperspectral Clinical Applications. *IEEE Access* **2019**, *7*, 152316–152333. [[CrossRef](#)]
6. Eismann, M.T.; Hardie, R.C. Application of the stochastic mixing model to hyperspectral resolution enhancement. *IEEE Trans. Geosci. Remote Sens.* **2004**, *42*, 1924–1933. [[CrossRef](#)]
7. Chang, C.-I. An information-theoretic approach to spectral variability, similarity, and discrimination for hyperspectral image analysis. *IEEE Trans. Inf. Theory* **2000**, *46*, 1927–1932. [[CrossRef](#)]
8. Jia, X.; Richards, J.A. Efficient maximum likelihood classification for imaging spectrometer data sets. *IEEE Trans. Geosci. Remote Sens.* **1994**, *32*, 274–281.
9. Chen, S.; Gunn, S.R.; Harris, C.J. The relevance vector machine technique for channel equalization application. *IEEE Trans. Neural Netw.* **2001**, *12*, 1529–1532. [[CrossRef](#)]
10. 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. [[CrossRef](#)]
11. Chen, Y.; Nasrabadi, N.M.; Tran, T.D. Hyperspectral Image Classification via Kernel Sparse Representation. *IEEE Trans. Geosci. Remote Sens.* **2013**, *51*, 217–231. [[CrossRef](#)]
12. Melgani, F.; Bruzzone, L. Classification of hyperspectral remote sensing images with support vector machines. *IEEE Trans. Geosci. Remote Sens.* **2004**, *42*, 1778–1790. [[CrossRef](#)]
13. Fang, L.; Li, S.; Kang, X.; Benediktsson, J.A. Spectral–Spatial Hyperspectral Image Classification via Multiscale Adaptive Sparse Representation. *IEEE Trans. Geosci. Remote Sens.* **2014**, *52*, 7738–7749. [[CrossRef](#)]
14. Baassou, B.; He, M.; Mei, S.; Zhang, Y. Unsupervised hyperspectral image classification algorithm by integrating spatial-spectral information. In Proceedings of the 2012 International Conference on Audio, Language and Image Processing, Shanghai, China, 16–18 July 2012; pp. 610–615.
15. Lary, D.J.; Alavi, A.H.; Gandomi, A.H.; Walker, A.L. Machine learning in geosciences and remote sensing. *Geosci. Front.* **2016**, *7*, 3–10. [[CrossRef](#)]
16. Chen, L.; Wei, Z.; Xu, Y. A Lightweight Spectral–Spatial Feature Extraction and Fusion Network for Hyperspectral Image Classification. *Remote Sens.* **2020**, *12*, 1395. [[CrossRef](#)]
17. Ma, W.; Ma, H.; Zhu, H.; Li, Y.; Li, L.; Jiao, L.; Hou, B. Hyperspectral Image Classification Based on Spatial and Spectral Kernels Generation Network. *Inf. Sci.* **2021**, *578*, 435–456. [[CrossRef](#)]
18. Hang, R.; Li, Z.; Liu, Q.; Ghamisi, P.; Bhattacharyya, S.S. Hyperspectral image classification with attention aided CNNs. *IEEE Trans. Geosci. Remote Sens.* **2020**, *59*, 2281–2293. [[CrossRef](#)]
19. Abdulsamad, T.; Chen, F.; Xue, Y.; Wang, Y.; Zeng, D. Hyperspectral image classification based on spectral and spatial information using resnet with channel attention. *Opt. Quantum Electron.* **2021**, *53*, 1–20. [[CrossRef](#)]
20. Pande, S.; Banerjee, B. Adaptive hybrid attention network for hyperspectral image classification. *Pattern Recognit. Lett.* **2021**, *144*, 6–12. [[CrossRef](#)]
21. Zhu, M.; Jiao, L.; Liu, F.; Yang, S.; Wang, J. Residual spectral-spatial attention network for hyperspectral image classification. *IEEE Trans. Geosci. Remote Sens.* **2020**, *59*, 449–462. [[CrossRef](#)]
22. 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](#)]
23. Chen, Y.; Zhao, X.; Jia, X. Spectral–Spatial Classification of Hyperspectral Data Based on Deep Belief Network. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2015**, *8*, 2381–2392. [[CrossRef](#)]
24. Hu, W.; Huang, Y.; Wei, L.; Zhang, F.; Li, H. Deep Convolutional Neural Networks for Hyperspectral Image Classification. *J. Sens.* **2015**, *2015*, 258619. [[CrossRef](#)]
25. Yang, J.; Zhao, Y.; Chan, J.C.-W.; Yi, C. Hyperspectral image classification using two-channel deep convolutional neural network. In Proceedings of the 2016 IEEE International Geoscience and Remote Sensing Symposium (IGARSS), Beijing, China, 10–15 July 2016; pp. 5079–5082.
26. 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. [[CrossRef](#)]
27. Chan, T.-H.; Jia, K.; Gao, S.; Lu, J.; Zeng, Z.; Ma, Y. PCANet: A Simple Deep Learning Baseline for Image Classification? *IEEE Trans. Image Process.* **2015**, *24*, 5017–5032. [[CrossRef](#)] [[PubMed](#)]
28. 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](#)]
29. Fahad, A.; Alshatri, N.; Tari, Z.; Alamri, A.; Khalil, I.; Zomaya, A.Y.; Fofou, S.; Bouras, A. A Survey of Clustering Algorithms for Big Data: Taxonomy and Empirical Analysis. *IEEE Trans. Emerg. Top. Comput.* **2014**, *2*, 267–279. [[CrossRef](#)]

30. Zhang, G.; Zhao, S.; Li, W.; Du, Q.; Ran, Q.; Tao, R. HTD-Net: A Deep Convolutional Neural Network for Target Detection in Hyperspectral Imagery. *Remote Sens.* **2020**, *12*, 1489. [[CrossRef](#)]
31. Wei, Y.; Zhou, Y. Spatial-Aware Network for Hyperspectral Image Classification. *Remote Sens.* **2021**, *13*, 3232. [[CrossRef](#)]
32. Li, W.; Wu, G.; Zhang, F.; Du, Q. Hyperspectral image classification using deep pixel-pair features. *IEEE Trans. Geosci. Remote Sens.* **2016**, *55*, 844–853. [[CrossRef](#)]
33. Kemker, R.; Kanan, C. Self-Taught Feature Learning for Hyperspectral Image Classification. *IEEE Trans. Geosci. Remote Sens.* **2017**, *55*, 2693–2705. [[CrossRef](#)]
34. Zhong, Z.; Li, J.; Luo, Z.; Chapman, M. Spectral-spatial residual network for hyperspectral image classification: A 3-d deep learning framework. *IEEE Trans. Geosci. Remote Sens.* **2017**, *56*, 847–858. [[CrossRef](#)]
35. Fang, B.; Li, Y.; Zhang, H.; Chan, J.C.-W. Hyperspectral Images Classification Based on Dense Convolutional Networks with Spectral-Wise Attention Mechanism. *Remote Sens.* **2019**, *11*, 159. [[CrossRef](#)]
36. Paoletti, M.E.; Haut, J.M.; Fernandez-Beltran, R.; Plaza, J.; Plaza, A.; Li, J.; Pla, F. Capsule networks for hyperspectral image classification. *IEEE Trans. Geosci. Remote Sens.* **2019**, *57*, 2145–2160. [[CrossRef](#)]
37. Li, W.; Wei, W.; Zhang, L.; Wang, C.; Zhang, Y. Unsupervised deep domain adaptation for hyperspectral image classification. In Proceedings of the IEEE International Geoscience and Remote Sensing Symposium, Yokohama, Japan, 28 July–2 August 2019; pp. 1–4. [[CrossRef](#)]
38. Wang, W.; Chen, Y.; He, X.; Li, Z. Soft Augmentation-Based Siamese CNN for Hyperspectral Image Classification with Limited Training Samples. *IEEE Geosci. Remote Sens. Lett.* **2021**, *19*, 1–5. [[CrossRef](#)]
39. Cui, Y.; Yu, Z.; Han, J.; Gao, S.; Wang, L. Dual-Triple Attention Network for Hyperspectral Image Classification Using Limited Training Samples. *IEEE Geosci. Remote Sens. Lett.* **2021**. [[CrossRef](#)]
40. Radford, A.; Metz, L.; Chintala, S. Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks. In Proceedings of the ICLR 2016: International Conference on Learning Representations, San Juan, PR, USA, 2–4 May 2016.
41. Creswell, A.; White, T.; Dumoulin, V.; Arulkumaran, K.; Sengupta, B.; Bharath, A.A. Generative Adversarial Networks: An Overview. *IEEE Signal Process. Mag.* **2018**, *35*, 53–65. [[CrossRef](#)]
42. Zhu, L.; Chen, Y.; Ghamisi, P.; Benediktsson, J.A. Generative Adversarial Networks for Hyperspectral Image Classification. *IEEE Trans. Geosci. Remote Sens.* **2018**, *56*, 5046–5063. [[CrossRef](#)]
43. Rodriguez, A.; Laio, A. Clustering by fast search and find of density peaks. *Science* **2014**, *344*, 1492–1496. [[CrossRef](#)]
44. Hyperspectral Remote Sensing Scenes. Available online: http://www.ehu.eus/ccwintco/index.php?title=Hyperspectral_Remote_Sensing_Scenes (accessed on 10 December 2021).
45. Olden, J.D.; Jackson, D.A. Illuminating the “black box”: A randomization approach for understanding variable contributions in artificial neural networks. *Ecol. Model.* **2002**, *154*, 135–150. [[CrossRef](#)]