remote sensing

MDPI

Article

Hyperspectral Imagery Classification Using Sparse Representations of Convolutional Neural Network Features

Heming Liang and Qi Li *

Institute of Remote Sensing and GIS, Peking University, Beijing 100871, China; heming126liang@126.com
* Correspondence: qi.lee009@gmail.com; Tel.: +86-10-6275-1964

Academic Editors: Xiaofeng Li and Prasad Thenkabail
Received: 19 October 2015; Accepted: 30 December 2015; Published: 27 January 2016

Abstract: In recent years, deep learning has been widely studied for remote sensing image analysis. In this paper, we propose a method for remotely-sensed image classification by using sparse representation of deep learning features. Specifically, we use convolutional neural networks (CNN) to extract deep features from high levels of the image data. Deep features provide high level spatial information created by hierarchical structures. Although the deep features may have high dimensionality, they lie in class-dependent sub-spaces or sub-manifolds. We investigate the characteristics of deep features by using a sparse representation classification framework. The experimental results reveal that the proposed method exploits the inherent low-dimensional structure of the deep features to provide better classification results as compared to the results obtained by widely-used feature exploration algorithms, such as the extended morphological attribute profiles (EMAPs) and sparse coding (SC).

Keywords: deep learning; deep features; sparse representation; remote sensing image classification

1. Introduction

Hyperspectral images can provide rich information both from the spectral and spatial domain simultaneously. For this reason, hyperspectral images are widely used in agriculture, environmental management and urban planning. Classification of each pixel in hyperspectral imagery is a common method used in these applications. However, hyperspectral sensors generally have more than 100 spectral bands for each pixel (e.g., AVIRIS, Reflective Optics System Imaging Spectrometer (ROSIS)), and the interpretation of such high dimensionality imagery with good accuracy is rather difficult.

Recently, sparse representation [1] has been demonstrated as a useful tool for high dimensional data processing. It is also widely applied in hyperspectral imagery classification [2–4]. Sparse models intend to represent most observations with linear combinations of a small set of elementary samples, often referred to as atoms, chosen from an over-complete training dictionary. In this way, hyperspectral pixels, which lie in a high dimension space, can be approximately represented by a low dimension subspace structured by dictionary atoms from the same class. Therefore, given the entire training dictionary, an unlabeled pixel can be sparsely represented by a specific linear combination of atoms. Finally, according to the positions and values of the sparse coefficients of the unlabeled pixel, the class label can be determined.

Spatial information is an important aspect in sparse representations of hyperspectral images. It is widely accepted that a combination of spatial and spectral information provides significant advantages in terms of improving the performance of hyperspectral image representation and classification (e.g., [5,6]). To explore effective spatial features, several methods have been developed in this direction. In [3],

two kinds of spatial-based sparse representation are proposed for hyperspectral image processing. Among them, one is a local contextual-based method. In this method, it adds a spatial smoothing term in the optimization formulation during the sparse reconstruction process of the original data. The second one jointly utilizes the sparse constraints of neighboring pixels, around the pixel of interest. The experimental results show that these two strategies perform better, in terms of classification results. However, both spatial smoothing and the joint sparsity model lay emphasis only on local consistency in the spectral domain, whereas spatial features (e.g., shapes and textures) also need to be explored for better representation of hyperspectral imagery. Recently, mathematical morphology (MM) methods [7] have been commonly used for modeling the spatial characteristics of the objects in hyperspectral images. For panchromatic images, derivative morphological profiles (DMPs) [8] have been successfully used for image classification. In the field of hyperspectral image interpretation, spatial features are commonly extracted by building extended morphological profiles (EMPs) [9] on the first few principal components. Moreover, extended morphological attribute profiles (EMAPs) [10], similar to EMPs, have been introduced as an advanced algorithm to obtain detailed multilevel spatial features of high resolution images generated by the sequential application of various spatial attribute filters that can be used to model different kinds of structural information. Such morphological spatial features, which are generated from the pixel level (low level), suffer heavily from redundancy and great variations in feature representation. To reduce the redundancy in morphological feature space, several studies have been set to find more representative spatial features by using a sparse coding technique, such as [11,12]. However, due to the variability of low-level morphological features, which limited the power of sparse representation, it is necessary to find higher level and more robust spatial features.

To explore higher level and more effective spatial features, [13] defines sparse contextual properties based on over-segmentation results, which greatly reduce computational cost. However, objects seldom belong to only one superpixel because of the spectral variations, and this is particularly so in high resolution images. Moreover, the spatial features, defined at the superpixel level, are commonly merged and linearly transformed from low level (pixel-level) ones; therefore, they probably would not significantly increase the representation power of spatial features in remote sensing images. After all, both MM and object-level spatial features require prior knowledge of setting proper parameters for feature extraction. The process of parameter setting always produces inefficient and redundant spatial features [14,15]. Therefore, in this paper, instead of setting spatial features, we explore high level spatial features by using a deep learning strategy [16,17]. Deep learning, as one of the state-of-the-art algorithms in the computer vision field, shifts the human-engineered feature extraction process to automatic feature learning and highly application-dependent feature exploration [18–20]. Furthermore, due to the deep structure in such learning strategies (e.g., stacked autoencoder (SAE) [21], convolutional neural network (CNN) [22]), one can extract higher level spatial features by using non-linear activation functions, layer by layer, which are much more robust and effective than low level ones. Recently, some efforts have been made in deep learning for hyperspectral image classification. Chen [23] probably is the first one to explore the SAE framework for hyperspectral classification. In his work, SAE was used for spectral and spatial feature extraction in the hierarchical structure. However, SAE can only extract higher level features from one-dimensional data, while it overlooked the two-dimensional spatial characteristics (although an adjacent effect has been considered). Unlike SAE, CNN takes a fixed size image patch, called the "receptive field", for deep spatial feature extraction; thus, it can keep spatial information intact. In the work of Chen [24], wherein the vehicles on the roads are detected by deep CNN (DCNN), the results show that CNN is effective for object detection in high resolution images. Instead of object detection, Yue [14,25] explored both spatial and spectral features in higher levels by using a deep CNN framework for the possible classification of hyperspectral images. However, the extracted deep features still remain in a high dimensional space, which involves a rather high computational cost and may lead to lower classification accuracies.

In this paper, we follow a different strategy, exploit the low dimensional structure of high level spatial features and perform sparse representation using both spectral and spatial information for

hyperspectral image classification. Specifically, we focus on CNN, which offers the potential to describe the structural characteristics in high levels according to the hierarchical feature extraction procedure. At the same time, we also exploit the fact that the deep spatial features of the same class lie in a low-dimensional subspace or manifold and can be expressed by linear sparse regression. Thus, it would be worthwhile to combine sparse representation with high dimensional deep features, which may provide better representation in terms of the characterization of spatial and spectral features and for better discrimination between different classes. Therefore, the method proposed in this paper for hyperspectral image classification combines the merits of deep learning and sparse representations. In this work, we tested our method on two well-known hyperspectral datasets: an Airborne Visible Infra-Red Imaging Spectrometer (AVIRIS) scene over Indian Pines, IN, USA, and a Reflective Optics System Imaging Spectrometer (ROSIS) scene over Pavia University. The experimental results show that the proposed method can effectively exploit the sparsity that lies at a higher level spatial feature subspace and also provides better classification performance. The merits of the proposed method are as follows: (1) instead of manually setting spatial features, we use CNN to learn such features automatically, which is more effective for hyperspectral image representation; (2) the hierarchical network strategy is applied to explore higher level spatial features, which are more robust and effective for classification compared to the low level spatial features; (3) the sparse representation method is introduced to exploit a suitable subspace for high dimension spatial features, which reduces the computational cost and increases feature discrimination between classes.

The remainder of the paper is structured as follows. Section 2 presents the proposed methodology in two parts: CNN deep feature extraction and sparse classification and describes the datasets used for experiments and compares the performance of the proposed method with that of other well-known approaches. Finally, Section 3 concludes with some suggestions for future works.

2. Proposed Methodology

The proposed method can be divided mainly into three parts, as shown in Figure 1. First, the high level spatial features are extracted by the CNN framework. Then, the sparse representation technique was applied to reduce the dimensionality of the high level spatial features generated by the previous step. Finally, with the learned sparse dictionary, classification results can be obtained.

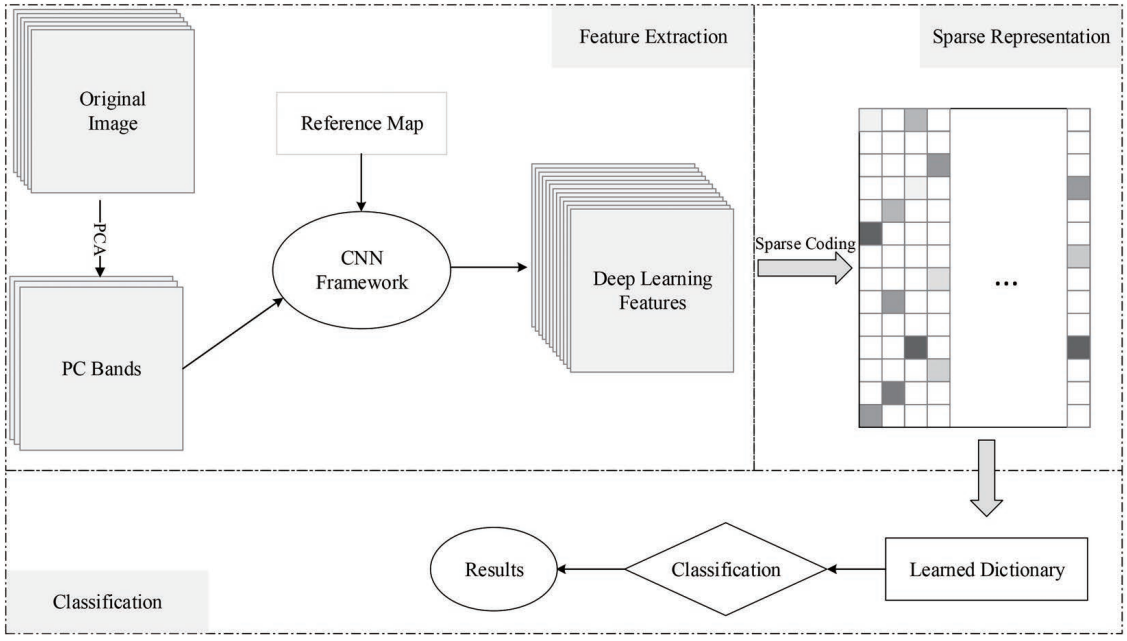

Figure 1. Graphical illustration of the convolutional neural networks (CNN)-based spatial feature extraction, sparse representation and classification of hyperspectral images.

2.1. CNN-Based Deep Feature Extraction

Recently, deep learning, one of the state-of-the-art techniques in the field of computer vision, has demonstrated impressive performances in recognition and classification of several famous image datasets [26,27]. Instead of setting image features, CNN can automatically learn higher level features from a hierarchical neural network in a way similar to the process of human cognition. To explore such spatial information, a fixed size of the neighborhood area (receptive field) should be first given. For a PC band of hyperspectral images, given a training sample p_i and its pixel neighbors in $P_s(p_i)$, a local neighborhood area forms with the size of $\mathcal{P} \times \mathcal{P}$; the patch-based training sample can be denoted as X_i. Additionally, the label of patch sample X_i can be denoted as t_i. CNN works like a black box; given the input patches and its labels, the hierarchical spatial features can be generated by a layer-wise activation structure, shown in Figure 2. Conventionally, two kinds of layers are stacked together in the CNN framework $f(k, b|X)$; the convolution layer and the sub-sampling layer [28]. Here, $f(x) = (1 + e^{-x})^1$ is the non-linear activation function. The convolution layer generates spatial features by activating the output value of previous layers with spatial filters. Then, the sub-sampling layer generates more general and abstract features, which greatly reduces the computational cost and increases the generalization power for image classification. Learning a CNN network with L layers involves learning the trainable parameters in each layer of the framework. The feature maps of the previous layer are convolved by the convolution layer with learnable kernel k and bias term b through the activation function to form feature maps of the current layer. For l-th convolution layer $l \in (1, 2, ..., L)$, we have that:

$$F^l = f(F^{l-1} * k^l + b^l) \tag{1}$$

where F^l represents the feature maps of the current layer and F^{l-1} means the feature map lies in the previous layer. k and b are trainable parameters in the convolution layer. Commonly, sub-sampling layers are interspersed with convolution layers for computational cost reduction and feature generalization. Specifically, a subsampling layer produces downsampled versions of the input feature maps for feature abstraction. For example, for the q-th sub-sampling layer $q \in (1, 2, ..., L)$, we have:

$$F^q = f(\text{down}(F^{q-1}) + b^q) \tag{2}$$

where down$(\cdot)$ represents the sub-sampling function that shrinks a feature map by using a mean value pooling operation and b is the bias term of the sub-sampling layer. The final output layer can be defined as:

$$y(k, b) = f^L(k^L h^{L-1} + b^L) \tag{3}$$

where $y(k, b)$ is the predicted value of the entire CNN and h^{L-1} means the output feature map of the $(L-1)$-th hidden layer in the CNN, which could be either a convolution layer or a sub-sampling layer. During the training process, the squared loss function is applied to measure the deviation from target labels and predicted labels. If there are N training samples, the optimization problem is to minimize the loss function E^N as follows:

$$\min E^N = \frac{1}{2} \sum_{i=1}^{N} ||t_i - y_i(k, b)||_2^2 \tag{4}$$

where $a^L = k^L h^{L-1} + b^L$ denotes a single activation unit. To minimize the loss function, a backward propagation algorithm is a common choice. Specifically, the stochastic gradient descent algorithm (SGD) is applied to optimize the parameters k and b.

The parameter of the entire network could be updated according to the derivatives. Once the back propagation process is finished, k and b are determined. Then, a feed-forward step is applied to generate new error derivatives, which can be used for another round for parameter updating. These feed-forward and back-propagation processes are repeated until convergence is achieved, and thus, optimal k and b are obtained. High level spatial features D_i can thus be extracted by using such learned parameters and a hierarchical framework.

$$O = f^L(\mathbf{k}X_i + \mathbf{b}) \tag{5}$$

Once the output feature map of the last layer is obtained, it is important to flatten the feature map into a one-dimension vector for pixel-based classification. Therefore, the flattened deep feature can be represented as $\mathbf{D_i} = \text{vectorize}(O)$, where O is the final output feature map.

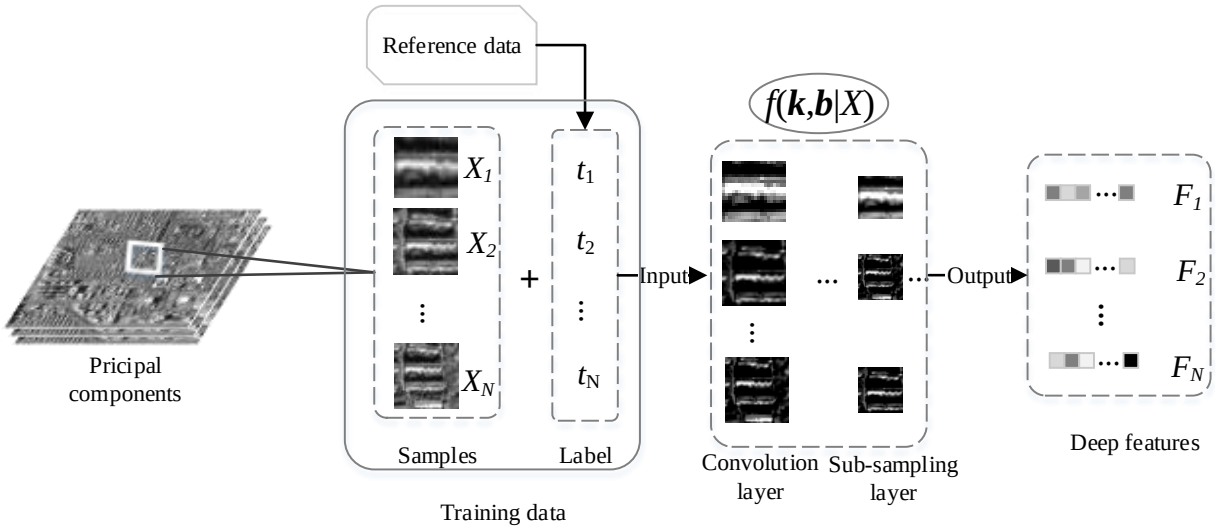

Figure 2. The process of CNN-based spatial feature extraction. The training samples are squared patches. The convolution layer and sub-sampling layer are interspersed in the framework of CNN.

2.2. Deep Feature-Based Sparse Representation

Deep spatial features generated by the CNN framework are usually of high dimensionality, which are ineffective for classification. Therefore, we introduce sparse coding as one of the-state-of-art techniques to find a subspace for deep feature representation and to possibly improve the classification performances, as shown in Figure 3. The sparse representation classification (SRC) framework was first introduced for face recognition [29]. Similarly, in hyperspectral images, a particular class with high dimensional features both in the spectral and spatial domain should lie in a low dimensional subspace spanned by dictionary atoms (training pixels) of the same class. Specifically, an unknown test pixel can be represented as a linear combination of training pixels from all classes. As a concrete example, let $\mathbf{x_i} \in \mathbb{R}^{M\times 1}$ be the pixel with M denoting the dimension of deep features in D and $\mathbf{A} = [\mathbf{A}_1, ..., \mathbf{A}_c, ..., \mathbf{A}_C]$ the structural dictionary, where $\mathbf{A}_c \in \mathbb{R}^{M\times n_c}, c = 1, ..., C$ holds the samples of class c in its columns; C is the number of classes, n_c is the number of samples in $\mathbf{A}_c$; and $\sum_{c=1}^{C} N_c = N$ is the total number of atoms in $\mathbf{A}$. Therefore, a pixel $\mathbf{x}_i$, whose class identity is unknown, can be represented as a linear combination of atoms from the dictionary $\mathbf{A}$:

$$\mathbf{x}_i = \mathbf{A}\boldsymbol{\alpha} \tag{6}$$

where $\mathbf{ff} \in \mathbb{R}^n$ is a sparse coefficient for the unknown pixel $\mathbf{x}_i$. Given the structural dictionary $\mathbf{A}$, the sparse coefficient $\boldsymbol{\alpha}$ can be obtained by solving the following optimization problem:

$$\hat{\boldsymbol{\alpha}} = \arg\min||\boldsymbol{\alpha}||_0 \quad \text{subject to} \quad ||\mathbf{x}_i - \mathbf{A}\boldsymbol{\alpha}||_2 \leq \delta \tag{7}$$

where $||\boldsymbol{\alpha}||_0$ denotes the ℓ_0-norm of $\boldsymbol{\alpha}$, which counts the number of nonzero components on the coefficient vector, and δ is the error tolerance, which represents noise and possible modeling error. However, the aforementioned problems make it hard to solve this optimization problem because of its nondeterministic and NP-hard characteristic. To tackle this problem, therefore, greedy algorithms, such as basis pursuit (BP) [30] and orthogonal matching pursuit (OMP) [31], have been proposed. In the BP algorithm, the ℓ_1 norm replaces the ℓ_0 norm. The optimization problem is transferred into:

$$\hat{\boldsymbol{\alpha}} = \arg\min||\boldsymbol{\alpha}||_1 \quad \text{subject to} \quad ||\boldsymbol{\alpha}||_0 \leq K \tag{8}$$

where K is the sparsity level, representing the number of selected atoms in the dictionary. $||\boldsymbol{\alpha}||_1 = \sum_i |\alpha_i|$, for $i = 1, ..., n$. On the other hand, the OMP algorithm incorporates the following steps at each iteration based on the correlation between the dictionary $\mathbf{A}$ and the residual vector $\mathbf{R}$, where $\mathbf{R} = x - \mathbf{A}\boldsymbol{\alpha}$. Specifically, at each iteration, the OMP finds the index of the atom that best approximates the residual, adds this member to the matrix of atoms, updates the residual and computes the estimate of $\boldsymbol{\alpha}$ using the newly-obtained atoms. Once the approximation error falls below a certain prescribed limit, then OMP finds the sparse coefficient vector $\hat{\boldsymbol{\alpha}}$. The class label for $\mathbf{x}_i$ can be determined by the minimal representation error between $\mathbf{x}_i$ and its approximation from the sub-dictionary of each class:

$$\hat{c} = \arg\min_c ||\mathbf{x}_i - \mathbf{A}_c\hat{\boldsymbol{\alpha}}_c||_2, \quad c = 1, ..., C \tag{9}$$

Therefore, we proposed the deep feature-based OMP algorithm to explore the low dimension subspace for deep feature representation and for image classification.

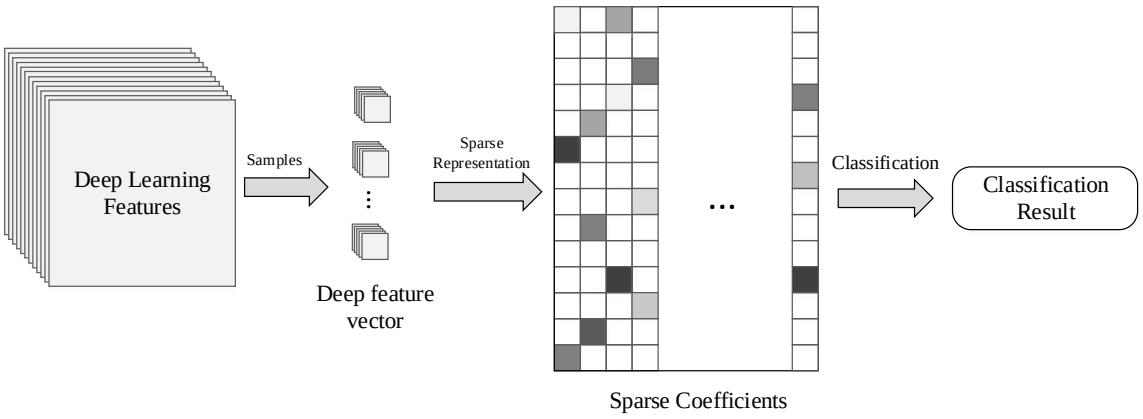

Figure 3. The process of deep feature-based sparse representation classification.

Algorithm 1 Framework of the deep feature-based sparse coding.

Require:

$\mathbf{x}_i$, training pixels from all classes with deep features; A, structural dictionary; C, number of classes; K, sparsity level;

Ensure:

$\hat{\alpha}$, sparse coefficients matrix;

1: Initialization: set the index matrix $\mathbf{I^{iter=1}} = \emptyset$, residual matrix $\mathbf{R^{iter=1}} = \mathbf{x}_i$, the iteration counter iter = 1;
2: Compute residual correlation matrix $\mathbf{E}^{iter}$: $\mathbf{E}^{iter} = \mathbf{A}^T\mathbf{R}^{iter}$;
3: Select a new adaptive set based on $\mathbf{E}^{iter}$
4: Find the best representation atoms' indexes $\mathbf{i}_c^{iter}$ and the corresponding coefficient values $\mathbf{v_c^{iter}}$ for each class c.
5: Combine the best representative atoms for each class into a cluster $\mathbf{W}_c^{iter}$, and obtain the corresponding coefficients $\mathbf{V}_c^{iter}$ in that cluster.
6: Select the adaptive set $\mathbf{L}^{iter}$ from the best atoms out of $\mathbf{W}_c^{iter}$ according to the indexes in $\mathbf{V}_c^{iter}$.
7: Combine the newly-selected adaptive set with the previously-selected adaptive sets: $\mathbf{I^{iter}} = \mathbf{I^{iter}} \bigcup \mathbf{L^{iter}}$
8: Calculate the sparse representation coefficients $\hat{\alpha}$.
9: Update the residual matrix: $\mathbf{R}^{iter} = \mathbf{x} - \mathbf{DA}$.
10: Check if sparsity coefficient $\hat{\alpha}^{iter} > K$; stop the procedures; and output the final sparse coefficient matrix; otherwise, set $iter = iter + 1$ and go to Step 2.

2.3. Datasets

In this section, we evaluate the performance of the proposed deep feature-based sparse classification algorithm on two hyperspectral image datasets, *i.e.*, the Reflective Optics System Imaging Spectrometer (ROSIS-03) University of Pavia data and the Airborne Visible/Infrared Imaging Spectrometer (AVIRIS) Indian Pines data.

The AVIRIS Indian Pines image was captured over the agricultural Indian Pine test site located in the northwest of Indiana. Two-thirds of the site contain agricultural crops and one-third forest or other natural, perennial vegetation. The spectral range includes 220 spectral bands from 0.2 to 2.4 μm, and each band measures 145 × 145 with a spatial resolution of 20 m. Prior to commencing the experiments, the water absorption bands were removed. There are 16 different classes in Indian Pines reference map, and most of them can be related to different types of crops.

The University of Pavia image was acquired by the ROSIS-03 sensor over the University of Pavia, Italy. The image measures 610 × 340 with a spatial resolution of 1.3 m per pixel. There are 115 channels whose coverage ranges from 0.43 to 0.86 μm. Prior to commencing the experiment, 12 absorption bands were discarded because of noise. Nine information classes were considered for this scene.

2.4. Configuration of CNN

During the deep feature extraction process, it is important to address the configuration of the deep learning framework. The receptive field ($\mathcal{P}$), the kernel (k), the number of layers (n_l) and the number of feature maps of each layer (n_f) are primary variables that affect the quality of deep features. We empirically set the size of the receptive field to $28 * 28$, which offered enough contextual information. The kernel sizes recommended in recent studies for CNN framework are 5×5, 7×7 or 9×9 [32]. For 7×7 and 9×9 kernels, there were 49 and 81 trainable parameters, which significantly increased the computational cost during training of such a framework, as compared to the cost for 5×5 kernels. Therefore, for our CNN framework, we adopted 5×5 kernels to accelerate the training process. Once the sizes of the receptive field and kernels are determined, the main structure of the CNN framework can be considered established. A training patch X_i (receptive field) can generate four levels of feature maps (two convolutional layers and two subsampling layers), and the size of the final output map is

4×4 $(((28 - (5 - 1))/2 - 4)/2)$. However, the number of feature maps (n_f) of each layer still remains unsolved. To solve this problem, we constrained the number of feature maps to be equal at each layer. With this configuration, CNN works, like the deep Boltzmann machine (DBM) [33], which should not significantly affect the quality of the output of deep features. To illustrate the impact of different CNN configurations on classification accuracy, we conducted a series of experiments, as will be explained in the following section. The experiment was conducted on the spatially independent training sets and the remaining as the test datasets.

2.4.1. CNN Depth Effect

The depth parameter of CNN plays an important role in classification accuracy, because it controls the deep feature quality in terms of the level of abstraction. To measure the effectiveness of the depth parameter, a series of experiments were conducted on the Pavia University dataset. We set four different depths of CNNs from 1 to 4, and the feature number was fixed to 50. Overall accuracy was used to measure the classification performance with different depth configurations. The experimental results are shown in Figures 4 and 5.

As can be seen from the figure, the classification accuracies can be obtained as increased with the increase in the depth configuration. The shallow layers contain low-level spatial features, but they vary greatly because of the constrained representation power. However, in deeper layers, the deep features are more robust and representative than those of lower ones. In addition, the shallow CNNs seem to have suffered more from overfitting, as presented in Figure 5.

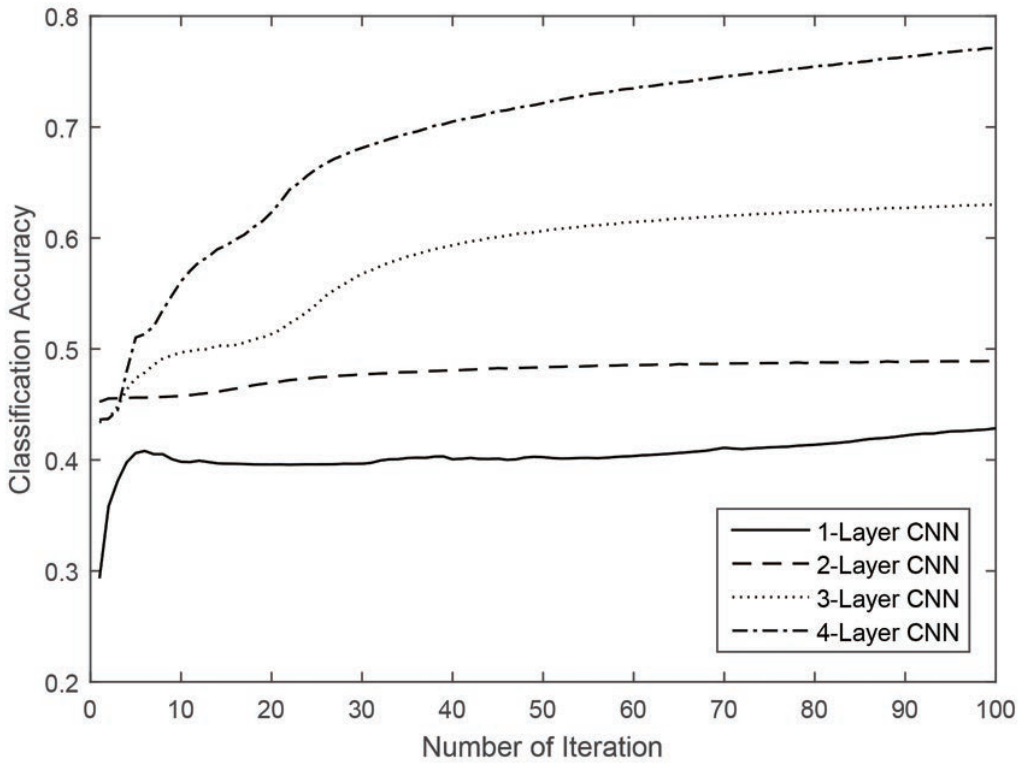

Figure 4. Overall accuracies of the university of Pavia dataset classified by CNN under different depths.

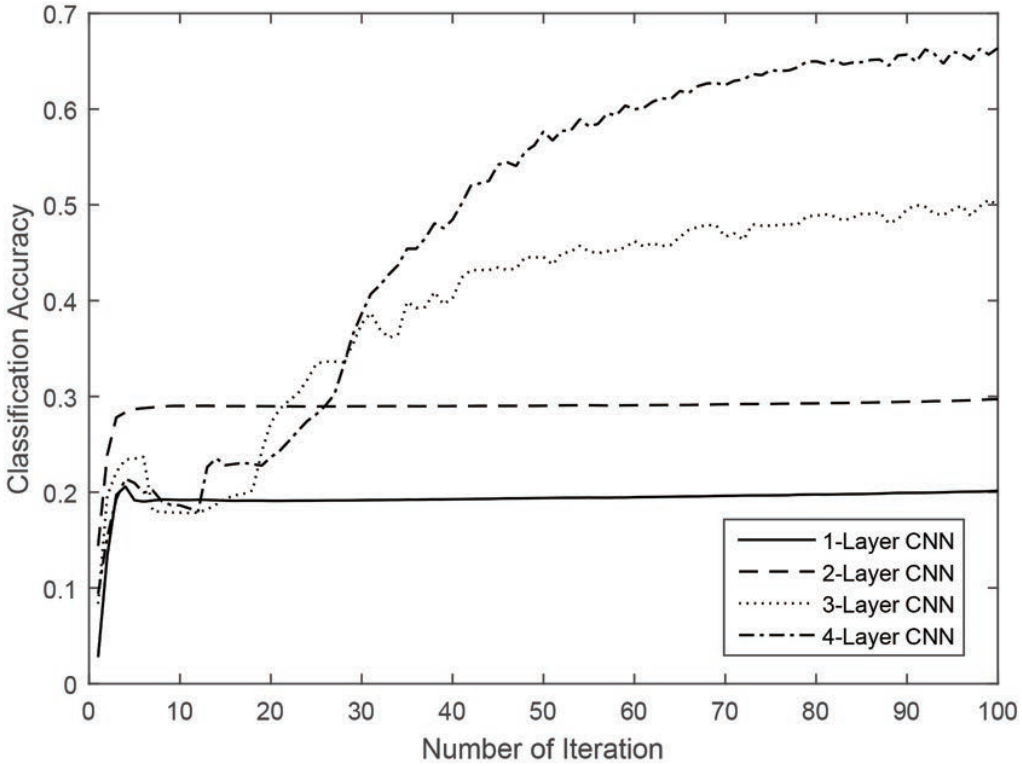

Figure 5. Overall accuracies of Indian Pines dataset classified by CNN under different depths.

2.4.2. CNN Feature Number Effect

In the CNN framework, the number of features can determine the dimensionality of the extracted spatial features. To measure the effect of spatial number on classification accuracy, a series of experiments were conducted. The feature number was varied from 10 to 100, and the whole CNN framework was constructed with a four-layer structure. Overall accuracy was used to measure the performance of the CNN-based classification algorithm. The classification results are reported in Figure 6.

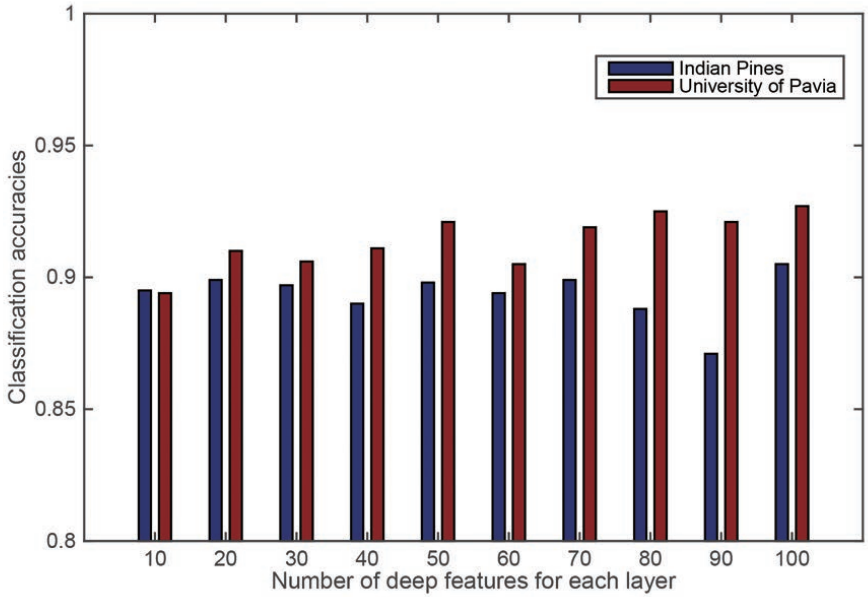

Figure 6. Classification results by setting different deep feature numbers for the CNN framework.

From the results for the University scene, it can be seen that the classification accuracy increased with the increase in feature number. However, no appreciable change in accuracy was noticed beyond the number of 50 deep features. A similar pattern can be seen in the Indian Pines dataset as well. Unlike in the University scene, the classification accuracy in the Indian Pines data dropped significantly after 50, reaching the lowest point at 90. This indicates that the classification accuracy becomes unstable when the number of deep features increases beyond a limit. Therefore, for our experiments, we set the number of deep features to 50.

2.5. Analysis of Sparse Representation

To conclude the effectiveness of sparse representation, we analyzed the relationship between the size of the training dictionary in both EMAP space and the deep feature space and the classification accuracies obtained by the OMP algorithm in this work. In Figure 7, the obtained classification accuracies are plotted as a function of the size of the training dictionary. The best classification accuracies are obtained by exploring the sparse representation of deep features for both the Indian Pines and Pavia University datasets. Generally, as the number of training samples increase, the uncertainty of classes decreases.

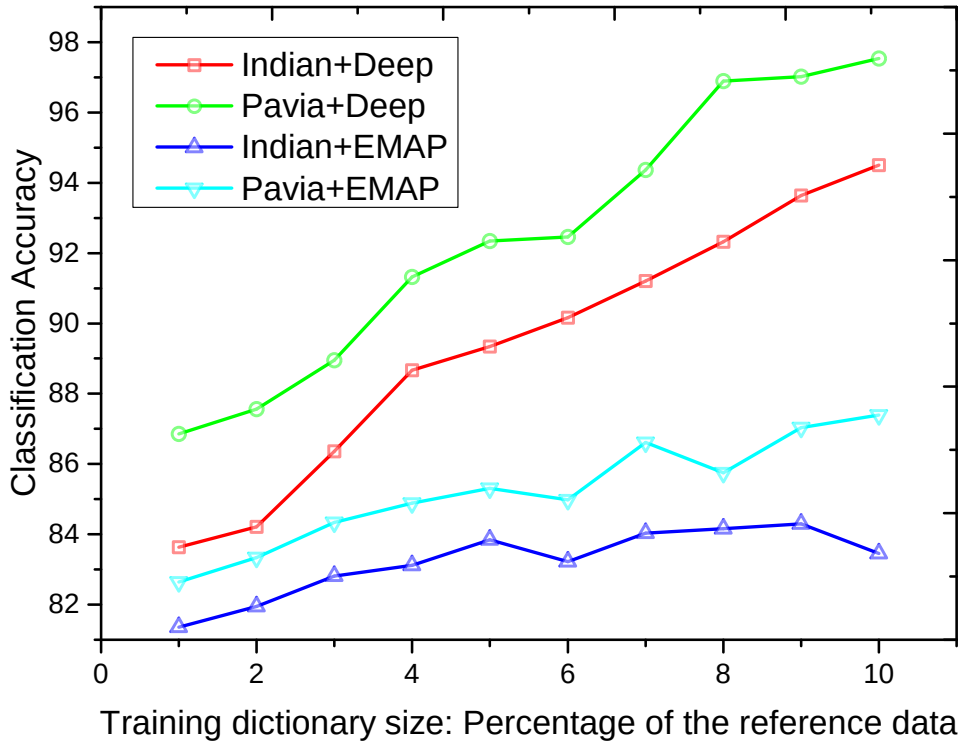

Figure 7. Classification OAs as a function of training dictionary size (expressed as a percentage of training samples for each class) for the Indian Pines and Pavia University datasets.

The following experiments illustrate the advantage of using a sparse representation in deep feature space for image classification over using the EMAP-based sparse coding. We considered a training dictionary made up of 1043 atoms and labeled the remaining samples as the test set. After constructing the dictionary, we randomly selected a pixel (belonging to Class 3) for sparse representation analysis, and the sparse coefficients are shown as bars in Figure 8. From these figures, it can seen that in the original spectral space, the sparse coefficients appear so mixed up that it is hard to distinguish one class from the other. In the EMAP space, the differences between classes are becoming clear, but they are also hard to classify in the highly mixed-up pixel. In the deep feature space, the unknown pixel can be seen as belonging to Class 3, because it is more discriminative than that in the spectral or EMAP

space. The reason behind this phenomenon is that the redundancy in spectral information and EMAP features greatly reduced the representativeness of the pixels. However, in the space of deep features, the correlation between different features is rather poor, and thus, it is more discriminative than the spectral and EMAP space.

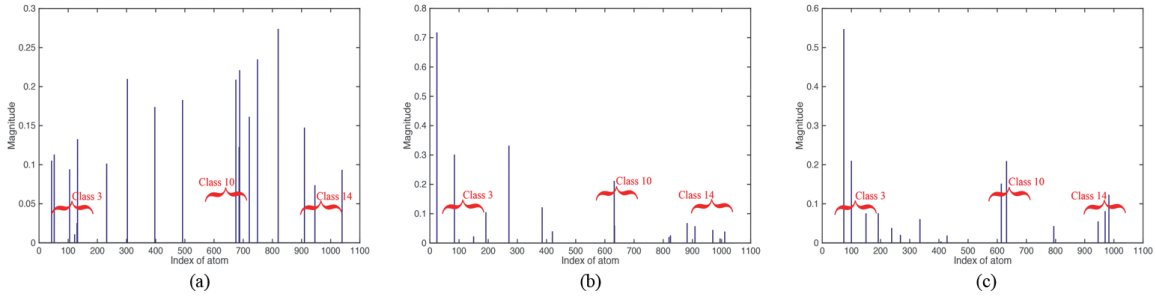

Figure 8. Estimated sparse coefficients (spectral space) for one pixel (belonging to Class 3) in the Indian Pines image. (**a**) Spectral space; (**b**) extended morphological attribute profile (EMAP) space; (**c**) deep feature space.

2.6. Comparison of Different Methods

The main purpose of the experiments with such remote sensing datasets is to compare the performances of different state-of-the-art algorithms in terms of classification results. Prior to feature extraction and classification, all of the datasets were whitened with the PCA algorithm, preserving the first several bands that contained more than 98% information.

To assess the effect of deep features, the well-known spatial feature extended morphological attribute profiles (EMAP) were introduced to classify the images in the spatial domain. Specifically, the EMAPs were built by using the attributes of area and standard deviation. Following the work in [34], threshold values were chosen for the area in the range of {50,500} with a stepwise increment of 50 and for a standard deviation in the range of 2.5% to 20% with a stepwise increment of 2.5%. However, both EMAP and deep features are commonly shown in great redundancy and also with high dimensionality. To address the importance of the sparsity constraint in such spectral and spatial features, we added the original spectral information to our EMAP and deep feature-based sparse representation classification experiments. It should be noted that, in all of the experiments, the OMP algorithm was used to approximately solve the sparse problem for the original spectral information, EMAP and deep features, which can be denoted respectively as Spe_o, EMAP_o and Deep_o. We also compared the proposed method with the nonlocal weighting sparse representation (NLW-SR) and the spectral-spatial deep convolutional neural network (SSDCNN) [25] in terms of classification accuracy.

In addition, we compared the sparse-based classification accuracy of OMP with the accuracies obtained by several state-of-the-art methods. Recently, some novel classification strategies have proposed for classifying hyperspectral images, such as random forest [35]. However, to evaluate the robustness of the deep features, the widely-used SVM classifier is considered as the benchmark in the experiment. It should be noted that the parameters of SVM were determined by five-fold cross-validation, and we selected the polynomial kernel for the rest of the experiments. The polynomial kernel can easily reveal the effectiveness of deep features, for future comparison. We denote the SVM-based classifications of spectral information, EMAP and deep features respectively as Spe_s, EMAP_s and Deep_s.

For our research, a series of experiments were conducted to extract deep features with different numbers of feature map settings. Furthermore, the training dictionary was constituted of randomly-selected samples from a reference map. The remaining samples were used for evaluating the classification performances. Overall accuracy (OA), average accuracy (AA) and the Kappa coefficient

were used to quantitatively measure the performance of the proposed method. The classification results are shown in Figure 9 and Figure 10.

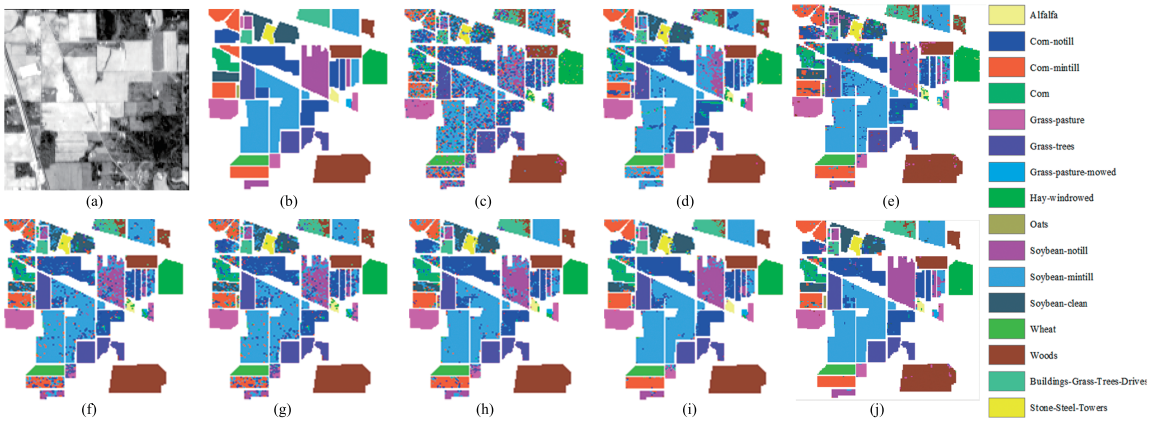

Figure 9. Classification results obtained by different classifiers for the AVIRIS Indian Pines scene. (**a**) Original map; (**b**) Reference map; (**c**) Spe_s classification map; (**d**) EMAP_s classification map; (**e**) Deep_s classification map; (**f**) Spe_o classification map; (**g**) EMAP_o classification map; (**h**) Nonlocal weighting sparse representation (NLW-SR) classification map; (**i**) Spectral-spatial deep convolutional neural network (SSDCNN) classification map; (**j**) Deep_o classification map.

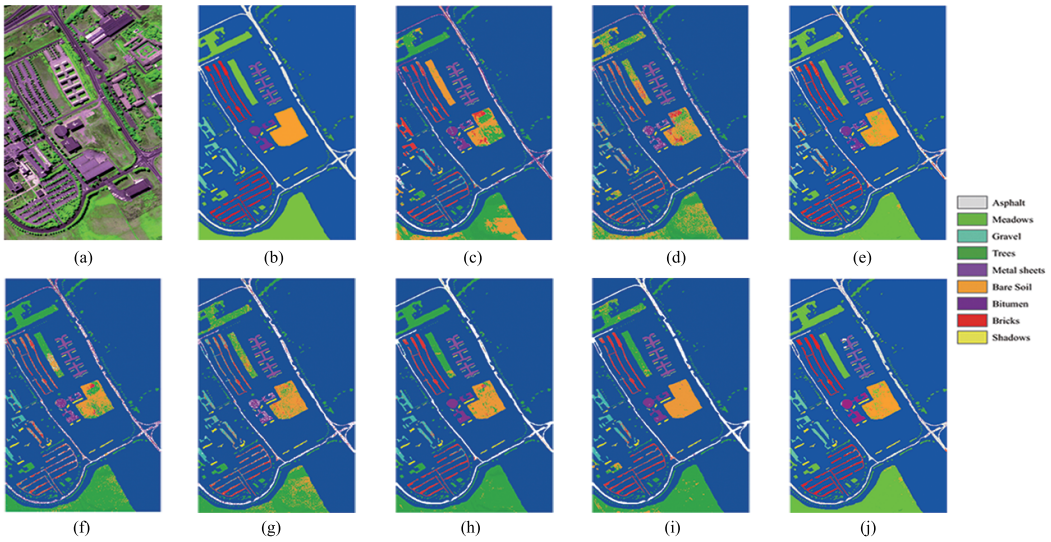

Figure 10. Classification results obtained by different classifiers for the Reflective Optics System Imaging Spectrometer (ROSIS) Pavia University Scene; (**b**) Reference map; (**c**) Spe_s classification map; (**d**) EMAP_s classification map; (**e**) Deep_s classification map; (**f**) Spe_o classification map; (**g**) EMAP_o classification map; (**h**) NLW-SR classification map; (**i**) SSDCNN classification map; (**j**) Deep_o classification map.

2.7. Experiments with the AVIRIS Indian Pines Scene

In our first experiment with AVIRIS Indian Pines dataset, we investigate the characteristic of CNN-based deep features. Specifically, we considered the four-layer CNN with 50 features at each layer as the default configurations for deep feature generation.

We compare the classification accuracies obtained for the Indian Pines dataset by the proposed method with those obtained by the other state-of-the-art classification methods. To illustrate the classification accuracies obtained with a limited number of training samples in a better way, the

individual class accuracies obtained for the case of 10% training samples are presented in Table 1. As can be seen in this table, in most cases, the proposed deep feature-based sparse classification ($Deep_o$) method provided the best results, in terms of individual class accuracies, as compared to the results obtained by other methods. When only the spectral information is considered, it is difficult to classify the Indian Pines image, because of the spectral mixture phenomenon. However, by introducing spatial information (EMAP), higher classification accuracies can be obtained in comparison to the accuracies obtained with the methods using only spectral information.

Table 1. OA, average accuracy (AA) and Kappa statistic obtained after executing 10 Monte Carlo runs for the AVIRIS Indian Pines data.

Class	Train	Test	Spe_s	$EMAP_s$	$Deep_s$	Spe_o	$EMAP_o$	NLW-SR	SSDCNN	$Deep_o$
1	3	43	54.88	94.88	85.42	33.33	**96.51**	96.34	90.34	95.83
2	14	1414	42.63	68.06	87.67	52.24	72.23	95.30	95.18	**96.08**
3	8	822	29.53	60.22	74.40	32.66	66.34	93.50	95.03	**95.46**
4	3	234	17.65	35.73	89.52	32.38	51.92	87.88	93.52	**97.28**
5	5	478	57.51	74.10	87.25	71.58	74.29	95.70	93.92	**98.01**
6	7	723	81.02	91.59	98.98	83.63	88.73	99.31	99.71	**99.80**
7	3	25	86.04	95.02	56.52	21.73	**98.00**	56.40	96.12	97.06
8	5	473	62.37	94.52	99.35	93.18	**99.98**	99.86	94.61	99.77
9	3	17	70.59	85.88	33.33	5.56	97.06	50.55	96.34	**99.55**
10	10	962	39.73	75.60	71.52	36.28	83.37	92.68	90.36	**93.66**
11	25	2430	73.31	87.37	94.32	63.21	88.61	96.43	95.67	**96.53**
12	6	587	21.72	57.68	77.17	39.31	70.49	91.14	88.34	**91.52**
13	3	202	87.28	96.44	99.36	93.15	98.61	91.52	96.65	**100**
14	13	1252	84.03	97.32	**99.92**	93.81	95.67	99.63	98.36	99.91
15	4	382	17.38	65.16	79.53	42.69	75.68	89.97	95.13	**97.91**
16	3	90	70.67	86.89	95.31	85.88	95.11	98.09	97.84	**98.29**
	OA		56.73	79.07	83.18	61.42	82.70	95.38	96.02	**97.45**
	AA		56.04	79.17	88.83	55.04	84.54	89.64	93.59	**95.91**
	Kappa		49.88	76.08	87.18	55.68	80.26	95.26	94.67	**96.36**
	time (s)		2.13	13.26	86.32	31.32	40.23	35.36	124.32	92.61

As regards the sparse representation effects, some important observations can be made from the content of Table 1. After introducing the sparse coding technique, both EMAP and deep feature-based classification methods show a significant improvement in terms of classification accuracy. This reveals the importance of using sparse representation techniques, particularly in EMAP and the deep feature space.

2.8. Experiments with the ROSIS Pavia University Scene

In the second experiment with the ROSIS Pavia University scene, we investigated the characteristic of CNN-based deep features. As in the case of the AVIRIS Indian Pines dataset, here also, we considered, for deep feature generation, a CNN with four layers and 50 features at each layer as the default configuration. Unlike the image of the Indian Pines dataset, the image of the Pavia University dataset is of high spatial resolution, which is even more complicated in terms of classification. Nine thematic land cover classes were identified in the university campus: trees, asphalt, bitumen, gravel, metal sheets, shadows, self-blocking bricks, meadows and bare soil. There are 42,776 reference datasets. From the training dataset, we randomly selected 300 samples per class to obtain the classification results by six different classification methods, and the results are presented in Table 2. The table shows the OA, AA, Kappa and individual class accuracies obtained with different classification algorithms. It can be seen that the proposed classification method $Deep_o$ provided the best results in terms of OA, AA and most of individual class accuracies.

In comparison, the classification accuracies obtained by using the SVM classifier tend to be lower than those obtained by sparse coding-based methods. With the introduction of EMAP features, the classification accuracies increased significantly, indicating thereby that spatial features are important, especially for high spatial resolution images. However, great redundancy lies in the EMAP space. Therefore, sparse coding-based OMP_{EMAP} can give better performance in terms of classification accuracy. Compared to EMAP features, deep features are more effective and representative. Therefore, deep feature-based classification methods (both Deep_s and Deep_o) provide higher classification accuracy.

Table 2. OA, AA and Kappa statistic obtained after executing 10 Monte Carlo runs for the ROSIS Pavia University scene.

Class	Train	Test	Spe_s	EMAP_s	Deep_s	Spe_o	EMAP_o	NLW-SR	SSDCNN	Deep_o
1	300	6631	84.87	86.96	**96.78**	65.66	89.31	90.25	84.56	94.78
2	300	18,649	64.53	78.99	97.83	60.38	84.74	97.13	98.95	**99.29**
3	300	2099	74.54	81.24	77.87	52.35	82.99	**99.80**	96.62	98.65
4	300	3064	94.37	95.02	87.74	92.57	76.92	97.42	95.33	**97.53**
5	300	1345	99.61	99.41	97.74	98.93	99.32	99.97	76.65	**100**
6	300	5029	72.77	84.98	81.48	47.89	88.62	99.50	98.45	**99.91**
7	300	1330	96.77	98.90	69.12	84.71	99.71	98.85	99.62	**99.80**
8	300	3682	77.59	90.88	86.92	74.59	87.88	98.24	94.57	**99.09**
9	300	947	99.44	99.21	**99.89**	96.19	92.08	86.70	96.83	98.29
	OA		75.28	84.92	91.80	65.62	86.61	96.50	95.18	**98.35**
	AA		84.94	90.62	88.39	74.81	89.06	96.43	93.51	**98.61**
	Kappa		69.14	80.90	90.12	57.15	82.85	95.34	93.64	**97.86**
	time (s)		2.51	34.92	93.19	86.82	97.63	57.31	134.75	104.82

3. Conclusions

In this paper, we investigated a new classification method that integrates sparse representations and deep learning techniques for spatial-spectral classification of hyperspectral remote sensing images. The classification results indicate that the proposed method can effectively classify the hyperspectral images. Furthermore, it can appropriately exploit the inherent sparsity present in deep features to provide state-of-the-art classification results. We also investigated the characteristics of deep learning features, which are more discriminative than the low-level hand-crafted spatial features. In comparison to the the state-of-the-art classifiers, the proposed method gives very promising results, particularly when the number of available training samples is very small. Future work will be focused on the development of computationally-efficient implementation of the proposed method.

Acknowledgments: The authors would like to thank David A. Landgrebe for making the AVIRIS Indian Pines hyperspectral dataset available to the community and Paolo Gamba for providing the ROSIS data over Pavia, Italy, along with the training and test sets. Additionally, we also should thank Mauro Dalla Mura for providing the EMAP code. Last, but not least, the authors would like to thank the Associate Editor and the three anonymous reviewers for their detailed and highly constructive criticisms, which greatly helped us to improve the quality and presentation of our manuscript.

Author Contributions: Heming Liang designed the study, developed the methodology, performed the experiments, analyzed the experimental results and wrote this paper. Qi Li supervised the study and revised this paper.

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

References

1. Bruckstein, A.M.; Donoho, D.L.; Elad, M. From sparse solutions of systems of equations to sparse modeling of signals and images. *SIAM Rev.* **2009**, *51*, 34–81.
2. Zhang, H.; Li, J.; Huang, Y.; Zhang, L. A nonlocal weighted joint sparse representation classification method for hyperspectral imagery. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2014**, *7*, 2056–2065.

3. Chen, Y.; Nasrabadi, N.; Tran, T. Hyperspectral image classification using dictionary-based sparse representation. *IEEE Trans. Geosci. Remote Sens.* **2011**, *49*, 3973–3985.
4. Tang, Y.Y.; Yuan, H.; Li, L. Manifold-based sparse representation for hyperspectral image classification. *IEEE Trans. Geosci. Remote Sens.* **2014**, *52*, 7606–7618.
5. Fauvel, M.; Benediktsson, J.; Chanussot, J.; Sveinsson, J. Spectral and spatial classification of hyperspectral data using SVMs and morphological profiles. *IEEE Trans. Geosci. Remote Sens.* **2008**, *46*, 3804–3814.
6. Bernabe, S.; Reddy Marpu, P.; Plaza, A.; Dalla Mura, M.; Atli Benediktsson, J. Spatial classification of multispectral images using kernel feature space representation. *IEEE Geosci. Remote Sens. Lett.* **2014**, *11*, 288–292.
7. Benediktsson, J.; Pesaresi, M.; Amason, K. Classification and feature extraction for remote sensing images from urban areas based on morphological transformations. *IEEE Trans. Geosci. Remote Sens.* **2003**, *41*, 1940–1949.
8. Plaza, A.; Martinez, P.; Plaza, J.; Perez, R. Dimensionality reduction and classification of hyperspectral image data using sequences of extended morphological transformations. *IEEE Trans. Geosci. Remote Sens.* **2005**, *43*, 466–479.
9. Benediktsson, J.; Palmason, J.; Sveinsson, J. Classification of hyperspectral data from urban areas based on extended morphological profiles. *IEEE Trans. Geosci. Remote Sens.* **2005**, *43*, 480–491.
10. Dalla Mura, M.; Benediktsson, J.; Waske, B.; Bruzzone, L. Morphological attribute profiles for the analysis of very high resolution images. *IEEE Trans. Geosci. Remote Sens.* **2010**, *48*, 3747–3762.
11. Song, B.; Li, J.; Dalla Mura, M.; Li, P.; Plaza, A.; Bioucas-Dias, J.; Atli Benediktsson, J.; Chanussot, J. Remotely sensed image classification using sparse representations of morphological attribute profiles. *IEEE Trans. Geosci. Remote Sens.* **2014**, *52*, 5122–5136.
12. Li, J.; Zhang, H.; Zhang, L. Supervised segmentation of very high resolution images by the use of extended morphological attribute profiles and a sparse transform. *IEEE Geosci. Remote Sens. Lett.* **2014**, *11*, 1409–1413.
13. Fang, L.; Li, S.; Kang, X.; Benediktsson, J. Spatial hyperspectral image classification via multiscale adaptive sparse representation. *IEEE Trans. Geosci. Remote Sens.* **2014**, *52*, 7738–7749.
14. Zhao, W.; Guo, Z.; Yue, J.; Zhang, X.; Luo, L. On combining multiscale deep learning features for the classification of hyperspectral remote sensing imagery. *Int. J. Remote Sens.* **2015**, *36*, 3368–3379.
15. Ji, R.; Gao, Y.; Hong, R.; Liu, Q.; Tao, D.; Li, X. Spectral-spatial constraint hyperspectral image classification. *IEEE Trans. Geosci. Remote Sens.* **2014**, *52*, 1811–1824.
16. Hinton, G.E.; Osindero, S.; Teh, Y.W. A fast learning algorithm for deep belief nets. *Neural Comput.* **2006**, *18*, 1527–1554.
17. Le, Q.V. Building high-level features using large scale unsupervised learning. In Proceedings of the 2013 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Vancouver, BC, Canada, 26–31 May 2013; pp. 8595–8598.
18. Rifai, S.; Vincent, P.; Muller, X.; Glorot, X.; Bengio, Y. Contractive auto-encoders: Explicit invariance during feature extraction. In Proceedings of the 28th International Conference on Machine Learning (ICML-11), Bellevue, WA, USA, 28 June–2 July 2011; pp. 833–840.
19. Razavian, A.S.; Azizpour, H.; Sullivan, J.; Carlsson, S. CNN features off-the-shelf: An astounding baseline for recognition. In Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Columbus, OH, USA, 23–28 June 2014; pp. 512–519.
20. Oquab, M.; Bottou, L.; Laptev, I.; Sivic, J. Learning and transferring mid-level image representations using convolutional neural networks. In Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Columbus, OH, USA, 23–28 June 2014; pp. 1717–1724.
21. Vincent, P.; Larochelle, H.; Lajoie, I.; Bengio, Y.; Manzagol, P.A. Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion. *J. Mach. Learn. Res.* **2010**, *11*, 3371–3408.
22. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. Imagenet classification with deep convolutional neural networks. In Proceedings of the Advances in Neural Information Processing Systems, Lake Tahoe, NV, USA, 3–8 December 2012; pp. 1097–1105.
23. 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.

24. Chen, X.; Xiang, S.; Liu, C.L.; Pan, C.H. Vehicle detection in satellite images by hybrid deep convolutional neural networks. *IEEE Geosci. Remote Sens. Lett.* **2014**, *11*, 1797–1801.
25. Yue, J.; Zhao, W.; Mao, S.; Liu, H. Spectral–spatial classification of hyperspectral images using deep convolutional neural networks. *Remote Sens. Lett.* **2015**, *6*, 468–477.
26. LeCun, Y.; Boser, B.; Denker, J.S.; Henderson, D.; Howard, R.E.; Hubbard, W.; Jackel, L.D. Backpropagation applied to handwritten zip code recognition. *Neural Comput.* **1989**, *1*, 541–551.
27. Arel, I.; Rose, D.; Karnowski, T. Deep machine learning—A new frontier in artificial intelligence research [research frontier]. *IEEE Comput. Int. Mag.* **2010**, *5*, 13–18.
28. Nebauer, C. Evaluation of convolutional neural networks for visual recognition. *IEEE Trans. Neural Netw.* **1998**, *9*, 685–696.
29. Wright, J.; Yang, A.; Ganesh, A.; Sastry, S.; Ma, Y. Robust face recognition via sparse representation. *IEEE Trans. Pattern Anal. Mach. Intell.* **2009**, *31*, 210–227.
30. Tropp, J. Greed is good: Algorithmic results for sparse approximation. *IEEE Trans. Inf. Theory* **2004**, *50*, 2231–2242.
31. Elad, M.; Aharon, M. Image denoising via sparse and redundant representations over learned dictionaries. *IEEE Trans. Image Process.* **2006**, *15*, 3736–3745.
32. Egmont-Petersen, M.; de Ridder, D.; Handels, H. Image processing with neural networks—A review. *Pattern Recognit.* **2002**, *35*, 2279–2301.
33. Salakhutdinov, R.; Hinton, G.E. Deep boltzmann machines. In Proceedings of the International Conference on Artificial Intelligence and Statistics, Clearwater Beach, FL, USA, 16–18 April 2009; pp. 448–455.
34. Pedergnana, M.; Marpu, P.; Mura, M.; Benediktsson, J.; Bruzzone, L. A novel technique for optimal feature selection in attribute profiles based on genetic algorithms. *IEEE Trans. Geosci. Remote Sens.* **2013**, *51*, 3514–3528.
35. Marpu, P.; Pedergnana, M.; Mura, M.; Peeters, S.; Benediktsson, J.; Bruzzone, L. Classification of hyperspectral data using extended attribute profiles based on supervised and unsupervised feature extraction techniques. *Int. J. Image Data Fusion* **2012**, *3*, 269–298.