

## Research Article

# Deep Belief Network for Feature Extraction of Urban Artificial Targets

Xiaoai Dai <sup>1</sup>, Junying Cheng,<sup>1</sup> Yu Gao,<sup>1</sup> Shouheng Guo,<sup>1</sup> Xingping Yang,<sup>1</sup> Xiaoqian Xu,<sup>2</sup> and Yi Cen<sup>3</sup>

<sup>1</sup>School of Earth Science Chengdu University of Technology, Chengdu 610059, China

<sup>2</sup>The First Scientific Research Institute of Wuxi, Wuxi 214000, China

<sup>3</sup>Institute of Remote Sensing and Digital Earth, Chinese Academy of Sciences, Beijing 100049, China

Correspondence should be addressed to Xiaoai Dai; daixiaoai@163.com

Received 21 February 2020; Accepted 1 May 2020; Published 30 May 2020

Academic Editor: Volodymyr Ponomaryov

Copyright © 2020 Xiaoai Dai et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Reducing the dimension of the hyperspectral image data can directly reduce the redundancy of the data, thus improving the accuracy of hyperspectral image classification. In this paper, the deep belief network algorithm in the theory of deep learning is introduced to extract the in-depth features of the imaging spectral image data. Firstly, the original data is mapped to feature space by unsupervised learning methods through the Restricted Boltzmann Machine (RBM). Then, a deep belief network will be formed by superimposed multiple Restricted Boltzmann Machines and training the model parameters by using the greedy algorithm layer by layer. At the same time, as the objective of data dimensionality reduction is achieved, the underground feature construction of the original data will be formed. The final step is to connect the depth features of the output to the Softmax regression classifier to complete the fine-tuning (FT) of the model and the final classification. Experiments using imaging spectral data showing the in-depth features extracted by the profound belief network algorithm have better robustness and separability. It can significantly improve the classification accuracy and has a good application prospect in hyperspectral image information extraction.

## 1. Introduction

Hyperspectral image classification is one of the most advanced techniques to understand the remote sensing image scene [1]. However, the classification constitutes many challenges because of the high dimensionality of images, high correlations among bands, and spectral mixing. Hyperspectral images are generally composed of hundreds or even thousands of relatively narrow bandwidth bands, which provide sufficient spectral and spatial information [2]. Combined with the spectral approach and optical approach, each space of the target pixel is dispersed during the process of spatial imaging, and then covering the spectrum. Characteristics of hyperspectral images and classification approaches, which are based on hyperspectral imaging, provide possibilities to classify the land surface objects high accurately [3, 4].

Hyperspectral datasets are composed of hundreds of bands and combine images with spectrum. They provide rich

surface spectral information and have incomparable advantages in the exceptional identification and classification of surface materials [5]. In terms of increasing the data dimension and selecting more data samples, band information is extended to increase the redundancy of the model. Although this improves the spectral resolution of hyperspectral remote sensing images, it dramatically affects the processing speed of the model data and also reduces the accuracy of the model and affects the target recognition. But high spectral resolution can affect the occurrence of Hughes effects [6]. The high correlation and information redundancy between the bands, as well as the foreign matter in the image and the same-spectrum foreign matter problem, result in a highly nonlinear data structure, which also makes extract information from the imaging spectral data difficult [7]. Therefore, the dimension reduction method is used to extract more productive and stable low-dimensional features to express the original high-dimensional data. While

reducing the computational complexity, the improvement of the classification accuracy on imaging spectral images has become one of the leading research questions in information extraction of spectral images [8].

Commonly used dimensionality reduction methods include linear dimensionality reduction and nonlinear dimensionality reduction. Linear dimensionality reduction methods mainly include principal component analysis (PCA) [9], independent components analysis (ICA) [10], and minimum noise fraction (MNF) [11]. However, the hyperspectral data have nonlinear structures, and the traditional linear dimensionality reduction method cannot reveal the nonlinear structure contained in the datasets. In recent years, the nonlinear manifold learning algorithm (NMLA) has been introduced to reduce the dimensionality of hyperspectral data. Bachmann et al. [12, 13] applied the improved isometric mapping algorithm (ISOMAP) [14] and locally linear embedding (LLE) [15] algorithm to the dimensionality reduction of hyperspectral images, further improving the accuracy of image classification. However, the current dimension reduction methods are often limited to extracting the shallow features of pixels, which may restrict the performance of classifiers. Deep learning extracts the pattern of features from the data according to the pre-designed feature extraction rules, from which the in-depth features of the pixels can be extracted, achieving the purpose of dimensionality reduction.

Deep learning can be regarded as the continuation and sublimation of neural networks. Deep learning gradually extracts the features from the low level to the advanced input by stimulating the learning process of the brain and finally forms the ideal characteristics of pattern classification to improve the classification accuracy. In 2006, Hinton and Salakhutdino proposed using a deep belief network (DBN) [16] to achieve data dimensionality reduction and classification. It is essentially the feature extraction of data using deep neural networks, called the deep learning algorithm. Typical methods for deep learning include Restricted Boltzmann Machines (RBM), deep belief network (DBN), convolution neural network (CNN), and auto encoder (AE) [17]. New deep learning approaches include the recurrent neural network (RNN), long short-term memory (LSTM), and generative adversarial nets (GAN). At present, deep learning has become the focus of attention in the field of machine learning and artificial intelligence. It has been widely used in image classification [18], target detection [19], speech image recognition [20], and natural language processing [21].

In this paper, based on the study of the dimensionality reduction method of traditional imaging spectral data, we introduce deep belief network based on the theory of deep learning to using a dimensionality reduction of hyperspectral images. The conventional dimension reduction method and deep belief network are compared to extract hyperspectral image information, and the robustness and separability of abstract features are considered. Finally, the optimal way of classification accuracy is verified.

## 2. Methods

**2.1. Restricted Boltzmann Machine (RBM).** The Restricted Boltzmann Machine is a typical energy-based model, as shown in Figure 1. Suppose that there is a bipartite graph. One is the visible layer (that is the data input layer), and the other one is the hidden layer. There are connections between all the visible layer and the hidden layer, but there is no connection between the hidden layer and the visible layer; that is, there is full connection between layers and no connection within layers. All nodes can only be taken as 0 or 1; that is, all nodes are random binary variables. At the same time, the full probability distribution  $p(v, h)$  satisfies the Boltzmann distribution. This model is a Restricted Boltzmann Machine (RBM). The energy function of the RBM model, such as equation (1), can be directly converted into a free energy form. The configured joint probability distribution can be determined by the Boltzmann distribution and the configured energy, as shown in equation (2). For example, equations (3) and (4) are conditionally independent because there is no connection between nodes in each layer, i.e., equations (5) and (6), so it is known that  $v, h$ , can be obtained by  $p(h | v)$ . Similarly,  $v'$  of the visible layer can be obtained by  $h$ , by adjusting the parameters to make  $v$  and  $v'$  equal. At this time,  $h$  can be expressed as a feature of the input data. There is a probability that the  $j^{\text{th}}$  node of the hidden layer is 1 or 0 when the visible layer  $v$  is given, such as equation (7). Similarly, there is a probability that the  $i^{\text{th}}$  node of the visible layer is 1 or 0 when the hidden layer is provided, such as equation (8). In this case, the free energy function can be expressed as equation (9) presenting a set of samples satisfying independent and identical distribution:  $D = \{v_1, v_2, v_3, \dots, v_n\}$ . It requires learning parameters  $\theta = \{W, a, b\}$ , and the RBM Log-likelihood gradient is equations (10)–(12). The contrastive divergence (CD) algorithm is used to update the weight, such as equations (13)–(15).

$v$  and  $h$  are the units of visible and hidden units, respectively.  $W$  is the connection weight between the visible layer and the hidden layer,  $b'$  is the bias of the neurons in the visible layer, and  $c'$  is the respective bias of the hidden layer. For a given set of states  $(v, h)$ , the energy possessed by RBM as a system is defined as

$$E(v, h) = -b'v - c'h - h'Wv, \quad (1)$$

$$F(v) = -b'v - \sum_i \log \sum_{h_i} e^{h_i(c_i + W_i v)}. \quad (2)$$

$\theta = \{W, b', c'\}$  is a parameter of RBM, all of which are real. When the parameters are determined, based on the energy function, we can get the joint probability density distribution of  $(v, h)$ :

$$\begin{aligned} P(v, h) &= \frac{1}{Z(\theta)} \exp(-E(v, h)) \\ &= \frac{1}{Z(\theta)} \prod_{ij} e^{W_{ij} v_i h_j} \prod_i e^{b_i v_i} \prod_j e^{c_j h_j}. \end{aligned} \quad (3)$$

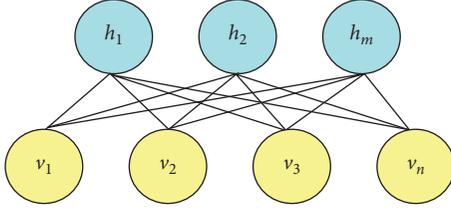


FIGURE 1: RBM model schematic.

$Z(\theta)$  is the normalization factor, which is the energy sum in all possible cases. The concept is formed by dividing the energy of a certain state by the total energy sum of possible states as follows:

$$Z(\theta) = \sum_{h,v} \exp(-E(v, h)), \quad (4)$$

$$p(h|v) = \prod_i p(h_i|v), \quad (5)$$

$$p(v|h) = \prod_j p(v_j|h). \quad (6)$$

When the state of a visible unit is given, the activation states of each hidden unit are independent of the conditions. At this point, the probability of activation of the  $j^{\text{th}}$  hidden unit is

$$P(h_i = 1|v) = \frac{e^{c_i + W_i v}}{1 + e^{c_i + W_i v}} = \sigma(c_i + W_i v). \quad (7)$$

Since the structure of the RBM is symmetrical, when given the state of the hidden unit, the activation states of each visible unit are also conditionally independent; that is, activation probability of the  $j^{\text{th}}$  visible unit is

$$P(v_j = 1|h) = \sigma(b_j + W'_j h). \quad (8)$$

It should be noted that, in both formulas (7) and (8),  $c_i$  and  $b_j$  are corresponding bias values.

$$F(v) = -b'v - \sum_i \log(1 + e^{(c_i + W_i v)}). \quad (9)$$

The following assumes only one training cost, and we use  $p(h|v)$  and  $p(v|h)$  probability distributions, respectively, then the logarithmic deterministic function is about the connection weight  $W_{ij}$ , the bias  $c_i$  of the visible layer unit, and the hidden layer unit. The partial derivatives of offset  $b_j$  are

$$-\frac{\partial \log p(v)}{\partial W_{ij}} = E_v [p(h_i|v) \cdot v_j] - v_j^{(i)} \cdot \sigma(W_i \cdot v^{(i)} + c_i), \quad (10)$$

$$-\frac{\partial \log p(v)}{\partial c_i} = E_v [p(h_i|v)] - \sigma(W_i \cdot v^{(i)}), \quad (11)$$

$$\frac{\partial \log p(v)}{\partial b_j} = E_v [p(v_j|v)] - v_j^{(i)}. \quad (12)$$

At the beginning of the CD algorithm, the state of the visible unit is set to a training sample and the binary state of

the hidden unit is calculated. After determining the status of all hidden units, the probability that the  $i^{\text{th}}$  visible unit takes a value of 1 is determined according to the formula, resulting in a reconstruction of the visible layer, so that when values are on the training data, the criteria for updating for each parameter are

$$W = W + \epsilon (h_1 x'_1 - Q(h_2 = 1 | x_2) x'_2), \quad (13)$$

$$b = b + \epsilon (x_1 - x_2), \quad (14)$$

$$c = c + \epsilon (h_1 - Q(h_2 = 1 | x_2)), \quad (15)$$

where  $\epsilon$  denotes the learning rate in the CD algorithm,  $Q(h_2 = 1 | x_2)$ ; it is a vector consisting of  $Q(h_{2i} = 1 | x_2)$ .

**2.2. Deep Belief Network Is Constructed Using Training Restricted Boltzmann Machine by Layer.** The deep belief network is a superposition of a multilayer of Restricted Boltzmann Machines, which can extract the indepth features of the original data. Figure 2 declares the model. The joint probability distribution between the input data  $v$  and the  $l$ -layer hidden layer  $h^k$  in the visible layer is shown in equation (16). The weight is obtained by using the unsupervised Greedy algorithm (GA). First, the first layer of the Restricted Boltzmann Machine is trained to fix the training parameters of the first layer. Then, the hidden layer output of the first layer of Restricted Boltzmann Machine takes as the input to the second layer of Restricted Boltzmann Machine, and the parameters of the first layer are successively trained layer by layer. The last hidden layer is connected to the Softmax regression classifier, and the fine-tuning (FT) is completed by the supervised Gradient descent (GD) algorithm.

$$P(v, h^1, h^2, \dots, h^l) = \left( \prod_{k=0}^{l-2} P(h^k | h^{k+1}) \right) P(h^{l-1}, h^l), \quad (16)$$

$P(h^{l-1}, h^l)$  is the joint probability distribution between the visible and hidden layers of the topmost RBM model.

**2.3. Deep Belief Network Algorithm Workflow.** The training process is shown in Figure 3. Give the training sample set  $X$  after initialization, there are  $k$  visible layer ( $v$ ) and  $j^{\text{th}}$  hidden layer ( $h$ ) in the RBM network structure, where the visible layer is only affected by  $j^{\text{th}}$  hidden layer. At the same time given the training period and learning rate, after the initialization of each parameter, the comparison dispersion algorithm is used to update the training parameters. If the algorithm converges, then the output, otherwise, continues with the parameter training as equations (13)–(15).

The training process is shown in Figure 4. Given the parameters and the number of hidden layers in Figure 3 after initialization, use Figure 3 to train the first layer of RBM. In this way, the hidden layer of the first layer of RBM is used as the input layer of the second layer of RBM, and the training is performed layer by layer until the last layer of RBM. Output the last layer and connect to the Softmax regression classifier, which is the output after fine-tuning (FT) (Figure 4).

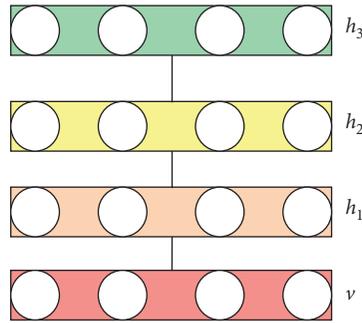


FIGURE 2: DBN model schematic.

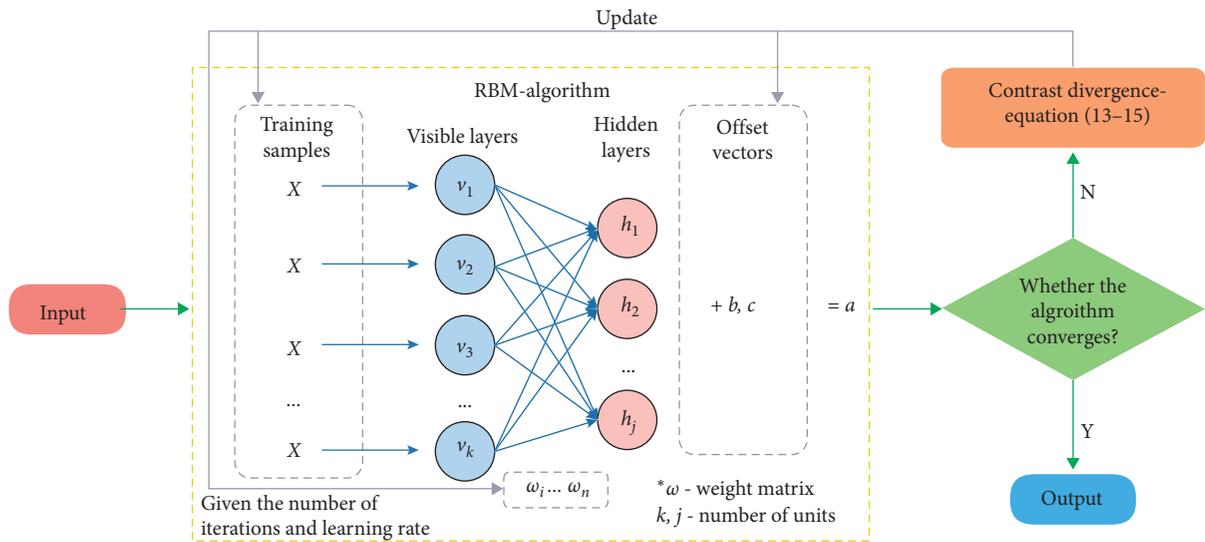


FIGURE 3: RBM training algorithm.

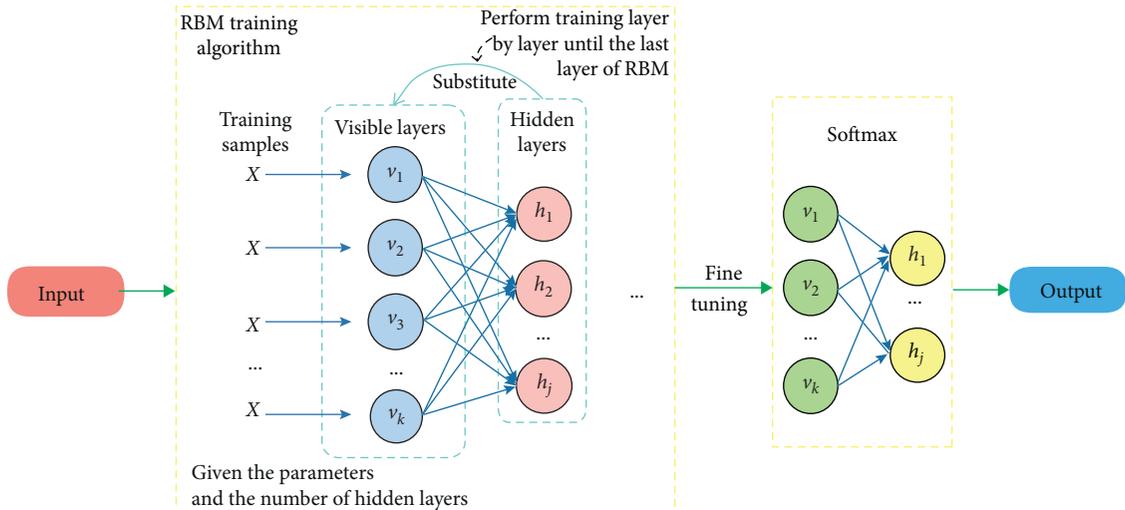


FIGURE 4: DBN training algorithm.

### 3. Data and Preprocessing

To verify the robustness of the model, two different hyperspectral image data types were simultaneously tested in

this section. The experiment has been carried out on two publicly available and widely used hyperspectral images: airborne hyperspectral image data and near-ground data captured by Hypspx imaging spectrometer data in Pavia

City, Italy. After preprocessing, such as radiation correction and reflectance inversion, the image pixel samples are connected to train the deep belief network model. Adjust and test the hyperparameters and training parameters at the same time, then compare them with the traditional dimensionality reduction method and analyze the optimal classification accuracy of the model to get the optimal classification model.

**3.1. Datasets.** The Pavia City image was gathered by the Reflective Optics System Imaging Spectrometer (ROSIS-3) optical sensor over the Pavia City, Italy. This image is  $610 \times 340$  pixels, as shown in Figure 5. The ROSIS-3 sensor generates 115 bands in the range of 430–860 nm, of which 103 bands except for the noisy band are selected for classification. Only the data without noisy band can effectively extract the characteristic bands. There are eight categories in the Pavia City image, as shown in Figure 6, and Table 1 shows the selection of sample data. According to the proportion of 3:1:1, the sample ratio is divided into training, verification, and test samples. The training samples are used to adjust the model's trainable parameters, the verification samples are used to improve the hyperparameters, and the test samples are used to test the classification accuracy of the model.

The near-ground image acquired by the Hypspec Imaging Spectrometer (Hypspec) uses the ground imaging method. This image is  $400 \times 600$  pixels, as shown in Figure 7. The Hypspec sensor generates 1600 spatial pixels and 108 bands in the range of 400–1000 nm. In the experiment, the Hypspec sensor selected 103 bands except the noisy band for classification to effectively extract the characteristic band. There are six categories in the sample spectrum curve, as shown in Figure 8, and the sample data selection is shown in Table 2.

The main advantages of the Hypspec imaging spectrometer are proper matching between point spread function and pixel size, low stray light, moderate spectral trapezoidal distortion effect, low polarization correlation, high sensitivity and low noise, high acquisition speed and data rate, real-time responsiveness, and dark compensation correction.

**3.2. Preprocessing.** ROSIS-3 has been preprocessed. The Hypspec uses a radiation correction of the original image obtained by the imaging calibration spectrometer. The reflectance inversion was performed by the Flat Field (FF) method based on the statistical model, and the large cement floor was selected as the Flat Field.

## 4. Results and Discussion

**4.1. RBM Model Analysis.** The deep belief network is composed of multilayered Restricted Boltzmann Machines, so the single Restricted Boltzmann Machine model is analyzed first. According to the hyperparameters selection opinion given in reference [16], the design learning rate is 0.1, and the Batch size is 20. The number of layers and hidden layer nodes of RBM model need to be determined by



FIGURE 5: ROSIS-3 data, true-color composite of the University of Pavia image (bands 56, 28, and 5).

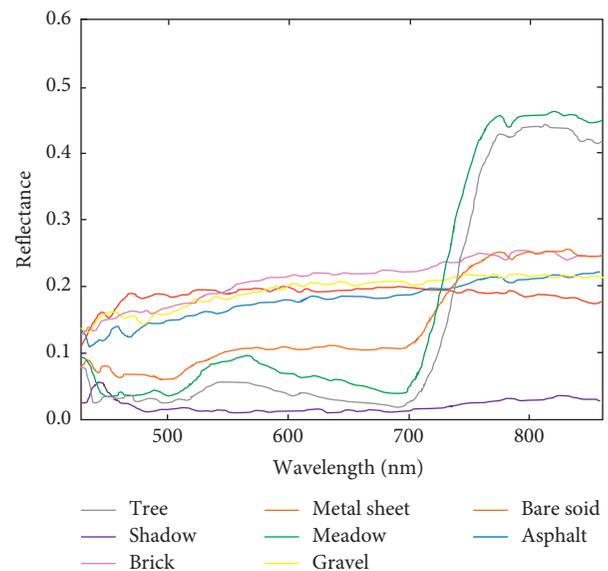


FIGURE 6: ROSIS-3 data, sample spectrum curve.

repeated experiments. By fixing other parameters and successively changing the number of layers, the optimal number of layers is obtained. The same training samples are used to train the RBM with different numbers of hidden layer neurons until the algorithm converges. The spectral curve of the original sample is compared with the reconstructed spectral curve under different experimental parameters, and the performance of the RBM under different numbers of neurons in the hidden layer is intuitively compared. In the following we try to discuss the influence of the number of hidden layer units and training optimization iterations of the model on the input data reconstruction ability. A representative water sample is selected to make the

TABLE 1: ROSIS-3 data classification category sample.

Category	Numbers			
	Training samples	Validation samples	Testing samples	Total
Asphalt	745	250	252	1247
Bare soil	231	82	85	398
Gravel	593	200	208	1001
Meadows	402	133	135	670
Metal sheet	1250	425	512	2187
Brick	352	133	139	624
Shadow	1647	518	509	2674
Tree	780	259	260	1299



FIGURE 7: Hypspx data, true-color composite of the near-ground image (bands 36, 28, and 10).

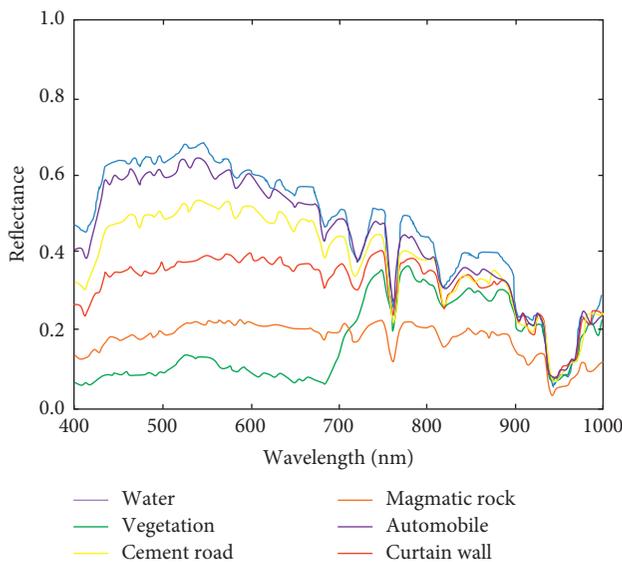


FIGURE 8: Hypspx data category sample spectroscopy curves.

experimental results more convincing. Figure 9(a) shows the original spectral curve. The number of fixed iterations is 100, and the number of hidden layer units is [20, 40, 60, 80, 100]. After training the model until convergence, the output reconstructed spectral curves are shown in Figures 9(b)–9(f). Therefore, when the number of hidden layer units is 60, the model is the most robust. The effect of the number of iterations on the performance of the Restricted Boltzmann Machine is obtained. Thus, selecting representative vegetation samples, the original spectral curve is shown in Figure 10(a). The number of fixed hidden layer neurons in the sample is 60, and the number of iterations is [100, 150, 200, 250, 300]. In this way, we discuss the effect of the number of iterations on the reconstruction capability of the input data. The reconstructed spectral curves of the output are as shown in Figures 10(b)–10(f). It can be obtained that when the number of iterations is 250, the reconstruction capability of the model starts to stabilize.

*4.2. Comparison with Traditional Dimensionality Reduction Methods.* To test the classification performance of the data after the dimension reduction and the deep belief network, the accuracy of this method is compared with the conventional dimension reduction method. In the experiment, we select a deep belief network model including two layers of Restricted Boltzmann Machines. The first layer parameter setting is 60 from the previous section, and the second layer is the final extracted feature number set to [4, 8, 12, 16, 20]. Meanwhile, conventional dimension reduction methods are principal component analysis, minimum noise separation, factor analysis (FA), and independent component analysis. All the dimension reduction methods have the same feature number extraction settings, and they are connected to the same classifier. In the experiment, we select the commonly used Support Vector Machine (SVM) and Softmax regression classifier for comparison analysis. SVM is a supervised classification method based on structural risk minimization. Its goal is to maximize the interval. It uses a limited number of boundary pixels to create a decision surface. By this way, the optimization problem becomes a convex quadratic programming problem. While the kernel function selects the radial basis kernel function suitable for hyperspectral image classification, the hyperparameter  $\sigma$  of the kernel function is

TABLE 2: Hypspx data classification category sample.

Category	Numbers			Total
	Training samples	Validation samples	Testing samples	
Water	515	168	179	862
Vegetation	528	177	172	877
Cement road	688	235	238	1161
Magmatic rock	374	140	145	659
Automobile	452	162	166	780
Curtain wall	415	138	145	698

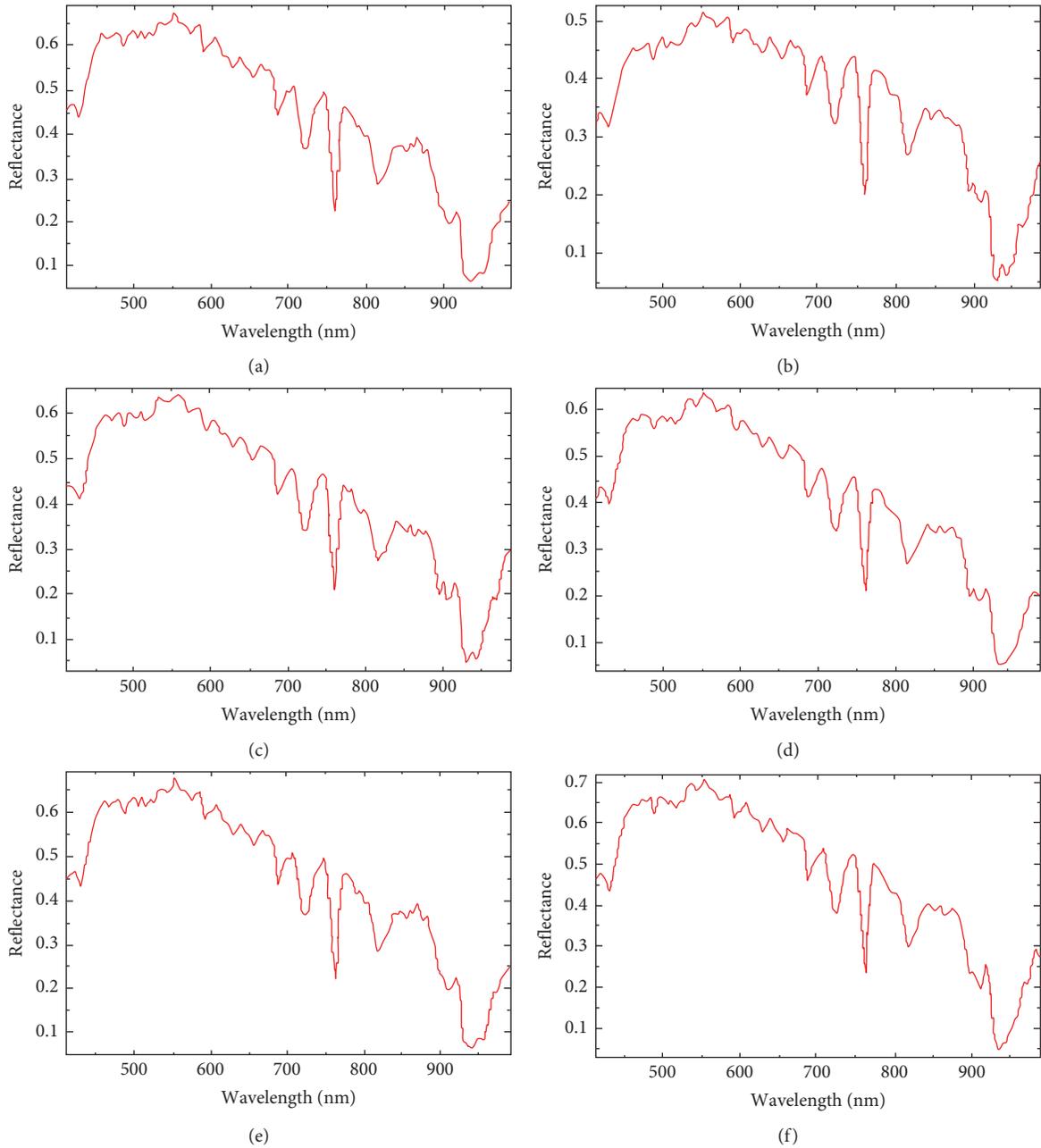


FIGURE 9: Original spectrum curve of water body and reconstructed spectrum curve. (a) Original spectrum curve of water. (b) Number of hidden layer units 20, reconstructed water body spectrum curve. (c) Number of hidden layer units 40, reconstructed water body spectrum curve. (d) Number of hidden layer units 60, reconstructed water body spectrum curve. (e) Number of hidden layer units 80, reconstructed water body spectrum curve. (f) Number of hidden layer units 100, reconstructed water body spectrum curve.

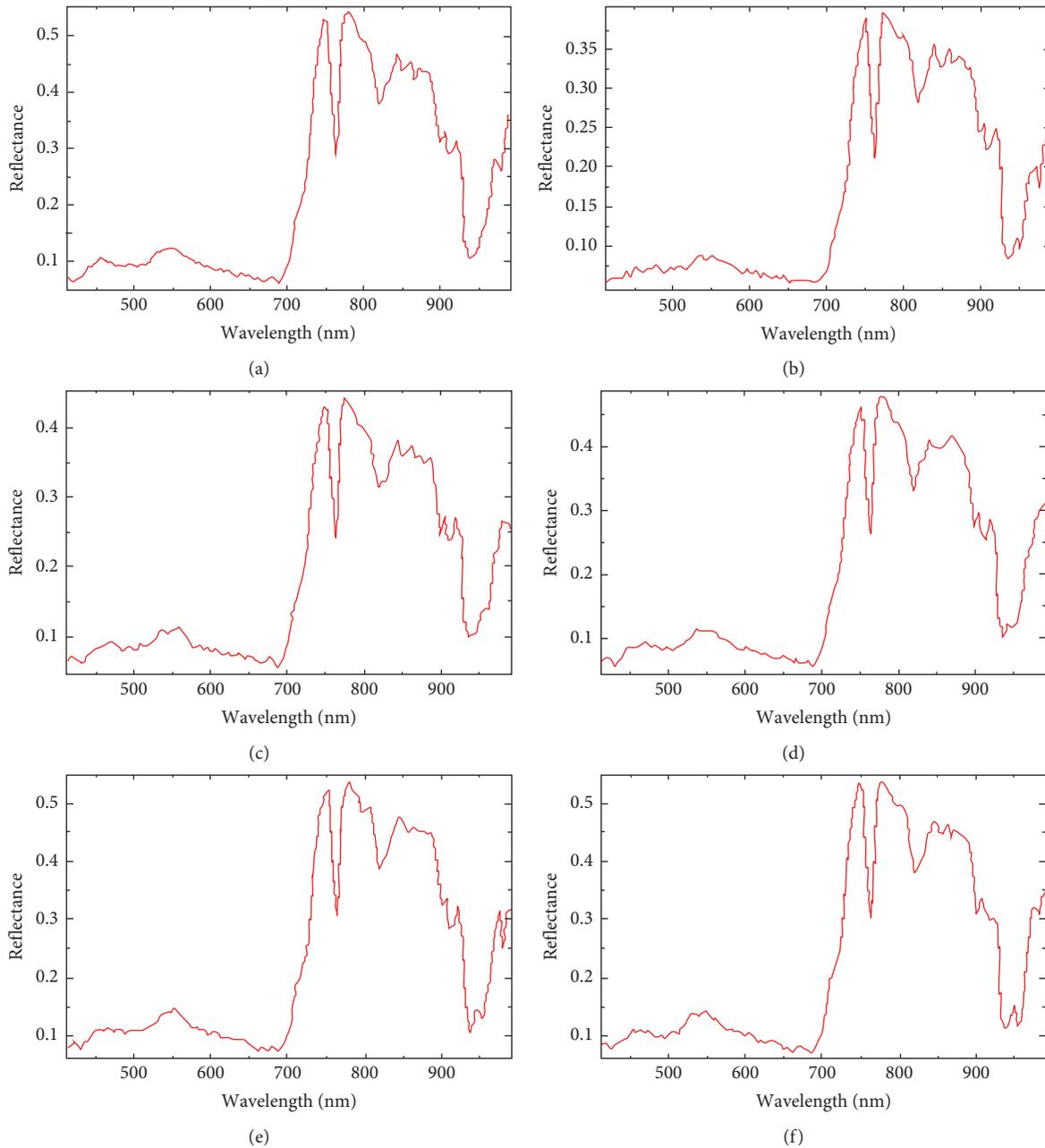


FIGURE 10: Original spectrum curve of vegetation body and reconstructed spectrum curve. (a) Original spectrum curve of vegetation. (b) Number of iterations 100, the reconstructed vegetation spectrum curve. (c) Number of iterations 150, the reconstructed vegetation spectrum curve. (d) Number of iterations 200, the reconstructed vegetation spectrum curve. (e) Number of iterations 250, the reconstructed vegetation spectrum curve. (f) Number of iterations 300, the reconstructed vegetation spectrum curve.

0.009, and the penalty parameter is 100. Softmax regression is a general form of Logistic regression for multiobject classification. For multiclassification problems, if  $C$  is used to indicate the number of categories, then the output layer of neural network, that is, the number of units in the  $L$  layer,  $n[L] = C$ . The output of each neuron corresponds to the probability of belonging to the class in turn. The learning rate of the Softmax regression selection is 0.1, and the optimization iteration number is 500. The experiment results of the ROSIS-3 data are shown in Figures 11 and 12, and the experiment results of the Hypspx data are shown in

Figures 13 and 14. Under experimental results, the Softmax regression classifier is more conducive to the feature classification after the dimension reduction, while other feature extraction methods are less accurate than the deep belief network.

*4.3. Analysis of the Influence of the Number of Implicit Layers in Deep Belief Network.* The selection of the number of hidden layers in the deep belief network determines whether it can extract appropriate features, which plays a essential

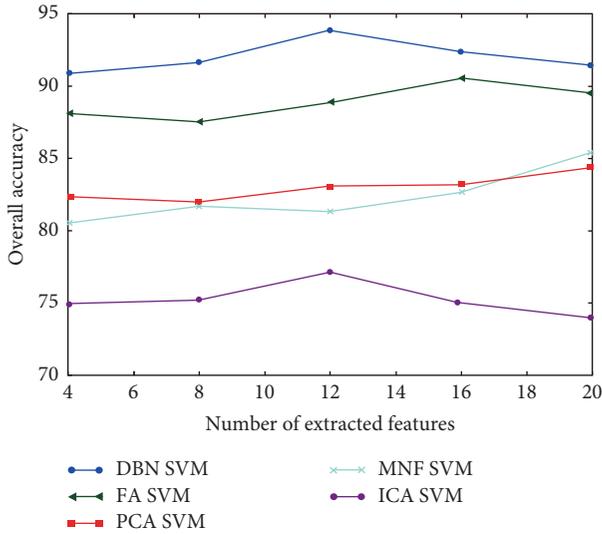


FIGURE 11: Comparison of classification accuracy of ROSIS-3 data based on different dimensionality reduction methods of SR classifier.

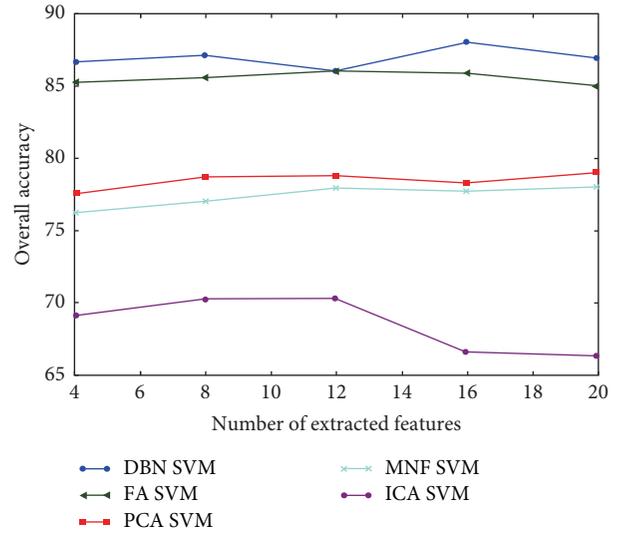


FIGURE 13: Comparison of classification accuracy of Hypspec data based on different dimensionality reduction methods of SR classifier.

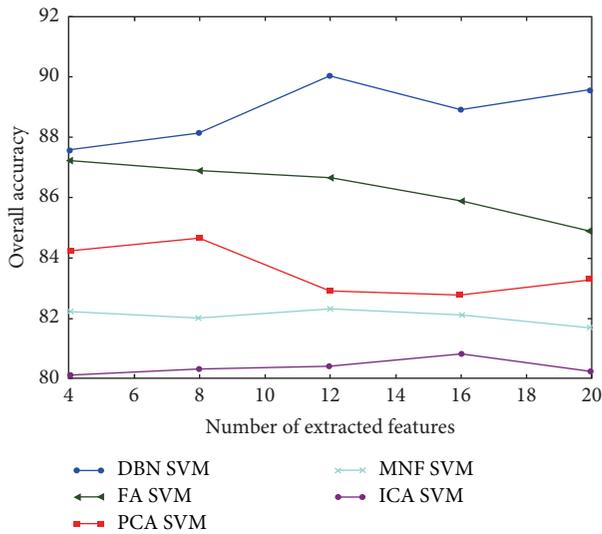


FIGURE 12: Comparison of classification accuracy of ROSIS-3 data based on different dimensionality reduction methods.

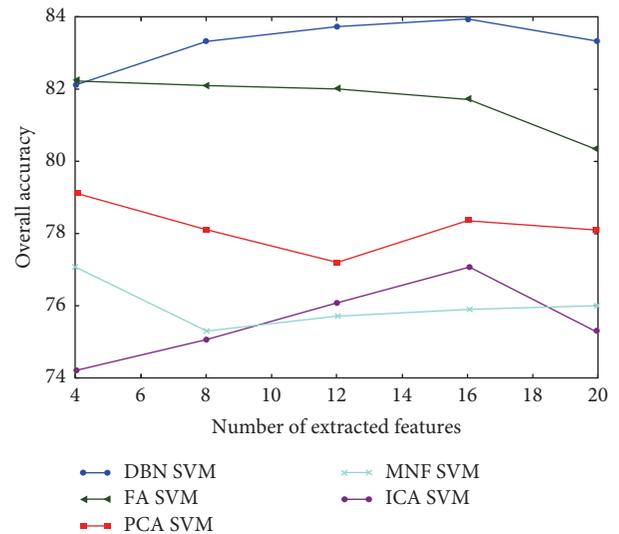


FIGURE 14: Comparison of classification accuracy of Hypspec data based on different dimensionality reduction methods of SVM classifier.

role in the accuracy of the final classification. When the number of layers is too few, only the features of the shallow layer can be extracted, which affects the classification accuracy. When the number of hidden layers increases, the abstract features obtained have better separability, which can improve the robustness of the classification model. But when there are too many layers, it is easy to make the model fall into overfitting. The number of hidden layers is set to [2–6]. To achieve the purpose of full dimensionality reduction, the number of top-level units is set to 4. The setting of the other parameters is consistent with the analysis of the Restricted Boltzmann model. The results are shown in Figure 15.

#### 4.4. Accuracy Evaluation and Image Classification Effect

4.4.1. *Optimal Model Accuracy Evaluation.* Compared with the traditional dimensionality reduction method, the deep belief network has a higher classification accuracy after dimensionality reduction. Moreover, the optimal model can be obtained when the number of hidden layers is 4; the number of units in each hidden layer is set to 60-60-60-4. Use confusion matrix to evaluate the classification accuracy of the optimal model and a ROSIS-3 confusion matrix shown in Table 3, with Hypspec confusion matrix being shown in Table 4.

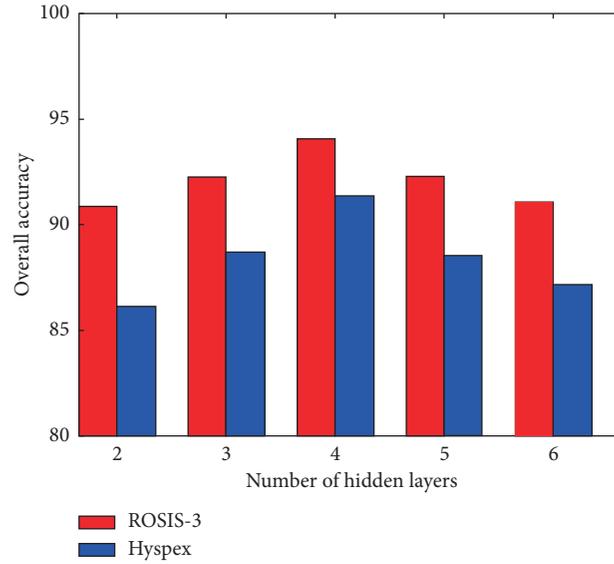


FIGURE 15: Comparative analysis of the precision of different DBN implicit layers.

TABLE 3: Confusion matrix of ROSIS-3 data DBN-SR classification accuracy.

Class no.	1	2	3	4	5	6	7	8	Total	Accuracy (%)
1	246	0	3	0	3	0	0	0	252	97.62
2	0	81	0	3	0	0	0	1	85	95.29
3	0	0	171	0	2	35	0	0	208	82.2
4	0	0	0	127	0	0	0	8	135	94.07
5	39	0	1	0	472	0	0	0	512	92.19
6	2	0	13	0	0	124	0	0	139	89.21
7	0	0	0	0	0	0	503	6	509	98.82
8	0	0	0	9	0	0	0	251	260	96.54
Total	287	81	188	139	477	159	503	266	2100	
Accuracy (%)	85.71	100	90.96	91.37	98.95	77.99	100	94.36		

$$\text{Overall accuracy} = 1975/2100 = 94.05\%$$

$$\text{Kappa} = (4147500 - 728590)/(4410000 - 728590) = 0.9287$$

(Class no.1: asphalt; 2: bare soil; 3: gravel; 4: meadows; 5: metal sheet; 6: brick; 7: shadow; 8: tree).

TABLE 4: Confusion matrix of Hypspx data DBN-SR classification accuracy.

Class no.	1	2	3	4	5	6	Total	Accuracy (%)
1	155	11	0	0	13	0	179	86.59
2	0	164	0	0	0	8	172	95.35
3	0	0	238	0	0	0	238	1
4	0	0	0	131	0	14	145	90.34
5	0	0	3	6	136	21	166	81.93
6	0	0	0	0	14	131	145	90.34
Total	155	175	241	137	163	174	1045	
Accuracy (%)	1	93.71	98.76	95.62	83.44	75.29		

$$\text{Overall accuracy} = 955/1045 = 91.39\%$$

$$\text{Kappa} = (997975 - 173163)/(1092025 - 173163) = 0.8976$$

(Class no.1: water; 2: vegetation; 3: cement road; 4: magmatic; 5: automobile; 6: curtain wall).

4.4.2. *Classification Effects.* Use the optimal model obtained by experimental analysis to classify the entire image. The effect of classification of test area I is shown in Figure 16(b),

and that of test area II is shown in Figure 17(b). Compared with the most commonly used support vector machine methods in hyperspectral image classification, the parameter

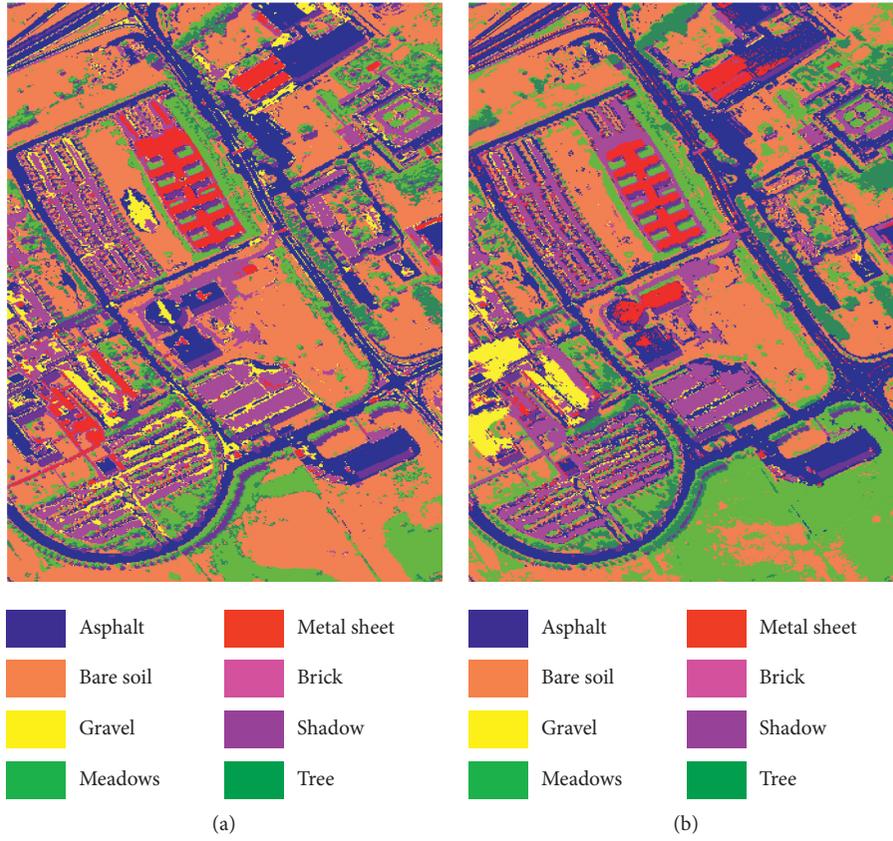


FIGURE 16: ROSIS-3 data classification renderings. (a) SVM. (b) DBN-SR.

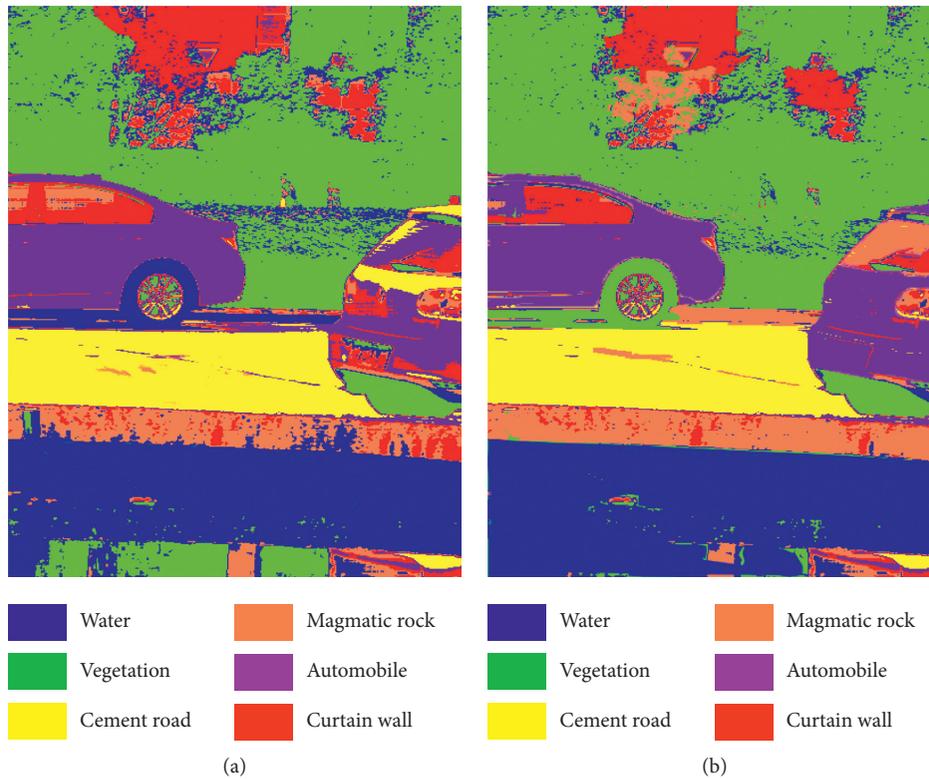


FIGURE 17: Hypex data classification renderings. (a) SVM. (b) DBN-SR.

setting is consistent with the previous text. The classification effect of ROSIS-3 is shown in Figure 16(a) and that of Hypspx is shown in Figure 17(a).

## 5. Conclusions

This paper presents new research based on the deep belief network to deal with the extraction of artificial target features in cities, as hyperspectral image. Based on the study of traditional imaging spectral data dimensionality reduction methods, it thoroughly considered that the conventional imaging spectral data dimensionality reduction methods only extract the shallow features of the pixels that tend to be unstable in the feature space, which limits the improvement of classification accuracy. Therefore, this paper introduces the deep belief network algorithm in the theory of deep learning that can not only reduce the dimension of data but also extract the depth features of pixels. Through the experimental analysis of the deep belief network model, it found that when using four hidden layers, the number of hidden layer units is 60-60-60-4, and connected to the Softmax regression classifier, the best classification accuracy can be obtained. Compared to traditional shallow feature extraction based on the Principal dimension analysis, minimum noise separation, factor analysis, independent component analysis, and other dimensionality reduction methods are obtained; the abstract features extracted by the deep belief network have better robustness and separability, which can lead to better classification accuracy and facilitate the phenotype of classifier performance. Besides, when using two different data types to test, the deep belief network has the best classification performance, which sufficiently proves that the model has broad applicability in imaging spectral data classification and information extraction. The next study will focus on the adjustment and selection of model parameters to obtain better classification results, as well as the introduction of the theory of deep learning into imaging spectral data processing.

## Data Availability

The data used to support the findings of this research are included within the article.

## Conflicts of Interest

The authors declare that they have no conflicts of interest.

## Acknowledgments

This research was funded by the Technology Foundation for Selected Overseas Chinese Scholar in Sichuan Province (no. 10900-19BZ08-014) and the International Science and Technology Innovation Cooperation between Governments Project of National Key Research and Development Program (no. 2017YFE9124900).

## References

- [1] B. Pan, Z. Shi, and X. Xu, "Hierarchical guidance filtering-based ensemble classification for hyperspectral images," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 55, no. 7, pp. 4177–4189, 2017.

- [2] D. Lu, P. Mausel, E. Brondizio, and E. Moran, "Change detection techniques," *International Journal of Remote Sensing*, vol. 25, no. 12, pp. 2365–2401, 2004.
- [3] L. Liang, "Hyperspectral remote sensing image classification based on ICA and SVM algorithm," *Spectroscopy and Spectral Analysis*, vol. 30, no. 10, pp. 2724–2728, 2010.
- [4] G. Camps-Valls, N. Shervashidze, and K. M. Borgwardt, "Spatio-spectral remote sensing image classification with graph kernels," *IEEE Geoscience and Remote Sensing Letters*, vol. 7, no. 4, pp. 741–745, 2010.
- [5] C.-I. Chang, *Hyperspectral Imaging: Techniques for Spectral Detection and Classification*, vol. 1, Springer Science & Business Media, Berlin, Germany, 2003..
- [6] G. Hughes, "On the mean accuracy of statistical pattern recognizers," *IEEE Transactions on Information Theory*, vol. 14, no. 1, pp. 55–63, 1968.
- [7] C.-I. Chang, *Hyperspectral Imaging: Signal Processing Algorithm Design and Analysis*, John Wiley & Sons, Hoboken, NJ, USA, 2007.
- [8] L. O. Jimenez, J. L. Rivera-Medina, E. Rodriguez-Diaz, E. Arzuaga-Cruz, and M. Ramirez-Velez, "Integration of spatial and spectral information by means of unsupervised extraction and classification for homogenous objects applied to multispectral and hyperspectral data," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 43, no. 4, pp. 844–851, 2005.
- [9] I. T. Jolliffe, "Principal component analysis," *Technometrics*, vol. 45, no. 3, p. 276, 2003.
- [10] A. Hyvarinen, J. Karhunen, and E. Oja, "Independent component analysis," *Studies in Informatics and Control*, vol. 11, no. 2, pp. 205–207, 2002.
- [11] A. A. Green, M. Berman, P. Switzer, and M. D. Craig, "A transformation for ordering multispectral data in terms of image quality with implications for noise removal," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 26, no. 1, pp. 65–74, 1988.
- [12] C. M. Bachmann, T. L. Ainsworth, and R. A. Fusina, "Improved Manifold coordinate representations of large-scale hyperspectral scenes," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 44, no. 10, pp. 2786–2803, 2006.
- [13] C. M. Bachmann, T. L. Ainsworth, and R. A. Fusina, "Exploiting Manifold geometry in hyperspectral imagery," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 43, no. 3, pp. 441–454, 2005.
- [14] J. B. Tenenbaum, V. De Silva, and J. C. Langford, "A global geometric framework for nonlinear dimensionality reduction," *Science*, vol. 290, no. 5500, pp. 2319–2323, 2000.
- [15] S. T. Roweis and K. L. Saul, "Nonlinear dimensionality reduction by locally linear embedding," *Science*, vol. 290, no. 5500, pp. 2323–2326, 2000.
- [16] G. E. Hinton and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *Science*, vol. 313, no. 5786, pp. 504–507, 2006.
- [17] K. Yu, L. Jia, Y. Chen, and W. Xu, "Deep learning: yesterday, today, and tomorrow," *Journal of Computer Research and Development*, vol. 50, no. 9, pp. 1799–1804, 2013.
- [18] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Proceedings of the Advances in Neural Information Processing Systems*, Lake Tahoe, NV, USA, 2012.
- [19] X. Chen, S. Xiang, C.-L. Liu, and C.-H. Pan, "Vehicle detection in satellite images by hybrid deep convolutional neural

- networks,” *IEEE Geoscience and Remote Sensing Letters*, vol. 11, no. 10, pp. 1797–1801, 2014.
- [20] D. C. Cireşan, U. Meier, L. M. Gambardella, and J. Schmidhuber, “Deep, big, simple neural nets for handwritten digit recognition,” *Neural Computation*, vol. 22, no. 12, pp. 3207–3220, 2010.
- [21] I. Sutskever, O. Vinyals, and Q. V. Le, “Sequence to sequence learning with neural networks,” in *Proceedings of the Advances in NIPS*, Montreal, Canada, 2014.