International Journal of Emerging Technology in Computer Science & Electronics (IJETCSE) ISSN: 0976-1353 Volume 25 Issue 5 – APRIL 2018 (SPECIAL ISSUE).

DEEP LEARNING TO DIVERSIFY BELIEF NETWORKS FOR REMOTE SENSING IMAGE CLASSIFICATION

S.Dhanalakshmi[#1]

#PG Scholar, Department of Computer Science, Dr.Sivanthi Aditanar college of Engineering, Tiruchendur

dhanojade3@hotmail.com

***Abstract* – In remote sensing, deep models with numerous layers have exhibited their possibilities in learning the abstract and invariant features for better representation and classification. The standard supervised deep models, for example, convolutional neural systems, require large number of labeled training samples. . However, the real-world hyperspectral image classification task provides only a limited number of training samples. This paper adopts another famous deep model, i.e., deep belief networks (DBNs), to deal with this problem. The DBNs allow unsupervised pretraining over unlabeled samples at first and then a supervised fine-tuning over labeled samples. But the usual pretraining and fine-tuning method would make many hidden units in the learned DBNs tend to behave very similarly .These outcomes could negatively affect the ability and classification performance of DBNs. To additionally enhance DBN's execution, this paper builds up another diversifying DBN through regularizing pretraining and adjusting fine-tuning procedures by a diversity promoting prior over latent factors. In addition, the regularized pretraining and fine-tuning can be efficiently implemented through usual recursive greedy and back-propagation learning framework. The experiments over real-world hyperspectral images exhibited that the diversity promoting prior in both pretraining and fine tuning procedure lead to the learned DBNs with more diverse latent factors, which specifically make the diversified DBNs obtain much better outcomes over unique DBNs .**

***Index Terms*—**Deep belief network (DBN), diversity, hyper spectral image, image classification, diversity promoting prior

I.INTRODUCTION

Remote sensing is the acquisition of information about an object or phenomenon without making physical contact with the object and thus in contrast to on site observation. In modern usage, the term generally refers to the use of aerial sensor technologies to detect and classify objects on Earth (both on the surface, and in the atmosphere and oceans) by means of propagated signals (e.g. electromagnetic radiation)

Hyperspectral images are composed of hundreds of spectral bands that increase the possibility to capture the spectral characteristics of remotely sensed scenes.

For each pixel in an image, a hyperspectral camera acquires the light intensity (radiance) for a large number (typically a few tens to several hundred) of contiguous spectral bands.

Every pixel in the image thus contains a continuous spectrum (in radiance or reflectance) and can be used to characterize the objects in the scene with great precision.

AVIRIS is an acronym for the Airborne Visible Infrared Imaging Spectrometer. AVIRIS is a premier instrument in the realm of Earth Remote Sensing. It is a unique optical sensor that delivers calibrated images of the upwelling spectral radiance in 224 contiguous spectral channels (also called bands) with wavelengths from 400 to 2500 nanometers (nm).

The main objective of the AVIRIS project is to identify, measure, and monitor constituents of the Earth's surface and atmosphere based on molecular absorption and particle scattering .

In terms of model structure, most of the popular image classification models have only one or two processing layers. They are called “shallow” methods. Recently deep architectures with more than two layers have gained increasing attention as they could demonstrate their potential in image classification

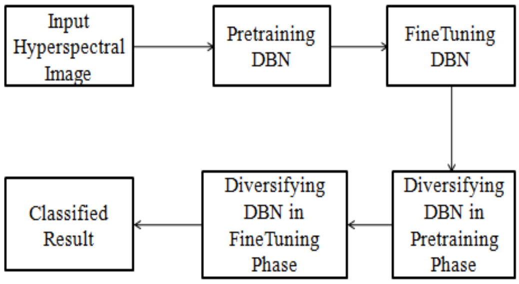


Fig1. Block Diagram

The DBN allow the unsupervised learning, and thus in this paper, we mainly investigate the DBN for its suitability to hyperspectral image classification. Generally the training of a DBN includes a unsupervised pretraining over the unlabeled samples at first and then a supervised fine-tuning over the labeled samples. Then all the diversified RBMs are stacked together to encode the input in a diversified hierarchical way, which will increase the model's description ability and indirectly benefits the later classification. However, since the pretraining step does not see any land cover labels that the hyperspectral image classification task aims at, the DBN diversified in pretraining cannot guarantee the optimal classification results. This paper is the first work that proposes to diversify a deep model to improve hyperspectral image classification. It should be mentioned that since a DBN is actually the stacking of multiple RBMs, the methods in proposed to diversify the RBMs can give us some basic theory on the layer-wise diversity of the DBN in pretraining step.

But the diversity in deep structures and the corresponding diversifying method still need to be investigated comprehensively. The rest of this paper is arranged as follows. The DBN for hyperspectral image representation and classification is depicted in Section II. Section III develops the method to diversify the DBN in both the pretraining and fine-tuning procedures. The proposed method will be evaluated over the real-world hyperspectral image data sets in Section IV. Finally, our technique is concluded and discussed in Section V.

II LEARNING DBN TO REPRESENT AND CLASSIFY HYPERSPECTRAL IMAGES

A DBN can be viewed as a stack of simple and unsupervised networks such as RBMs or auto encoders. This paper uses the RBMs to construct the DBN.

The remaining of this section is organized as follows: in Section II-A, we outline the structure of a DBN constructed from stacking multiple RBMs for the representation and classification of hyperspectral images; in Sections II-B and II-C, corresponding to the hyperspectral image representation and classification, respectively.

The pretraining and fine-tuning method of the DBN will be introduced.

Training all layers together in an unsupervised way is generally very difficult since both the definition of reasonable cost functions and corresponding optimizations are not easy works.

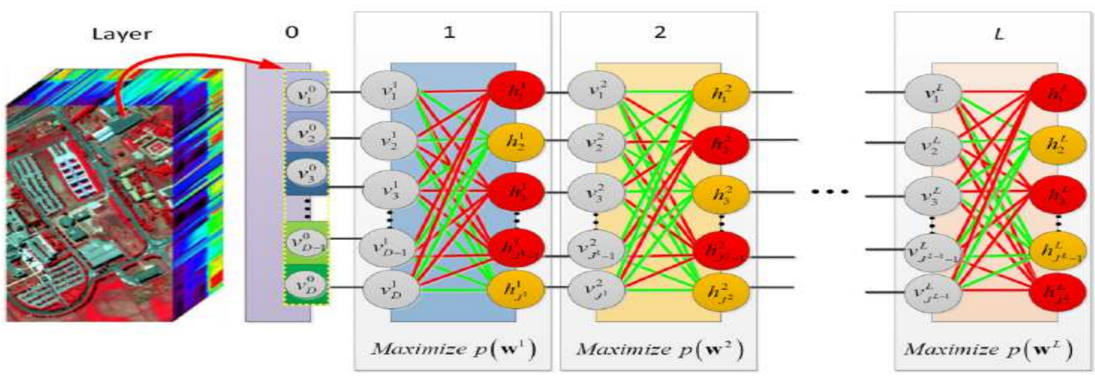


Fig2. Graphical representation of the DBN for hyperspectral image classification

A. DBN for Representation of Hyperspectral Images

A hyperspectral image usually has hundreds of spectral bands in a narrow bandwidth with fixed sample intervals. This abundant information equips the hyperspectral image with the potential to discriminate different land cover classes. However, the simple method using directly the spectral signature of each pixel cannot fully release the potential of the hyperspectral image. Hence, in order to get a better classification map, it is necessary to extract an informative representation of the original spectral signature. A DBN model constructed with a hierarchical series of RBMs could demonstrate the real power by capturing the important features in hyperspectral images.

The graphical representation of the DBN for hyperspectral image is shown in Fig. 1. The *D*-dimensional input $(v_1^0, v_2^0, ..., v_D^0)^T$ could be the spectral signature of a pixel or the feature extracted over a pixel and its neighbors. Then the series of RBMs are stacked hierarchically: using the output of the previous RBM as the input of the current RBM. In the DBN, two adjacent layers have a full set of connections between them, but no two units in the same layer are connected.

The structure of the DBN demonstrates that the first hidden layer merges the input spectral information and subsequently one or more hidden layers learn the spectral features elaborately.

Every layer can output a representation or feature of the input data, and the higher is the layer, the more abstract is the feature. Since the final task is to classify the hyper spectral images, a softmax layer is added as the last layer of the DBN to classify the extracted features from the last RBM.

Therefore, to formulate the DBN, we need only to give the mathematical details of the RBM. An RBM at the l layer of a DBN is an energy-based generative model that consists of one layer with binary visible units = { , ,…., }and one layer with binary hidden units ={ h_1^l ,…, }For all layers of DBN we have that $I^l - J^l$. The energy of the joint configuration of visible and hidden units (v^l, h^l) of the l layer is

(1)

$$E(v^l, h^l) = -\sum_{i=1}^{I^l} a_i^l v_i^l - \sum_{j=1}^{J^l} b_j^l h_j^l - \sum_{i=1}^{I^l} \sum_{j=1}^{J^l} w_{ij}^l v_i^l h_j^l$$

where ={ , , $w_{ij}^l, i = \{1, 2, \ldots, I^l\}, j = \{1, 2, \ldots, J\}$ is the set of model parameters. The and are the bias parameters of the visible and hidden units, respectively. The is the weight parameter between visible unit and hidden unit .The matrix of parameters is ={ w_1^l, ,…, } where each column $w^l_j = [w_1{}^l{}_j, w_2{}^l{}_j, \ldots, w^l{}_I l{}_j]^T$ corresponds to one hidden unit. to one hidden unit.

The RBM defines a joint probability over the units as

$$p(\mathbf{v}^l, \mathbf{h}^l; \theta^l) = \frac{\exp(-E(\mathbf{v}^l, \mathbf{h}^l; \theta^l))}{Z(\theta^l)} \quad \textbf{(2)}$$

Where *Z* is the partition function

$$Z(\theta^l) = \sum_{\mathbf{v}^l} \sum_{\mathbf{h}^l} \exp(-E(\mathbf{v}^l, \mathbf{h}^l; \theta^l)). \quad \textbf{(3)}$$

Then the conditional distribution of input vector **h***l* given hidden unit **v***l* is given by logistic function

$$p(h_j^l = 1 | \mathbf{v}^l) = \sigma\left(b_j^l + \sum_{i=1}^{I^l} w_{ij}^l v_i^l\right) \quad \text{for all } j = 1, 2, \ldots, J^l \quad \textbf{(4)}$$

while the conditional distribution of hidden unit **v***l* given input vector **h***l* is

$$p(v^l = 1 | \mathbf{h}^l) = \sigma\left(a_i^l + \sum_{i=1}^{J^l} w_{ij}^l h_j^l\right) \quad \textbf{(5)}$$

where *σ(•)* is the logistic function

$$\sigma(x) = \frac{1}{1+\exp(-x)}. \tag{6}$$

As shown in Fig. 1, the DBN training can be generally divided into two phases: the pretraining for representation and the fine-tuning for classification. The details about these two phases will be presented in the following sections.

B. Pretraining DBN

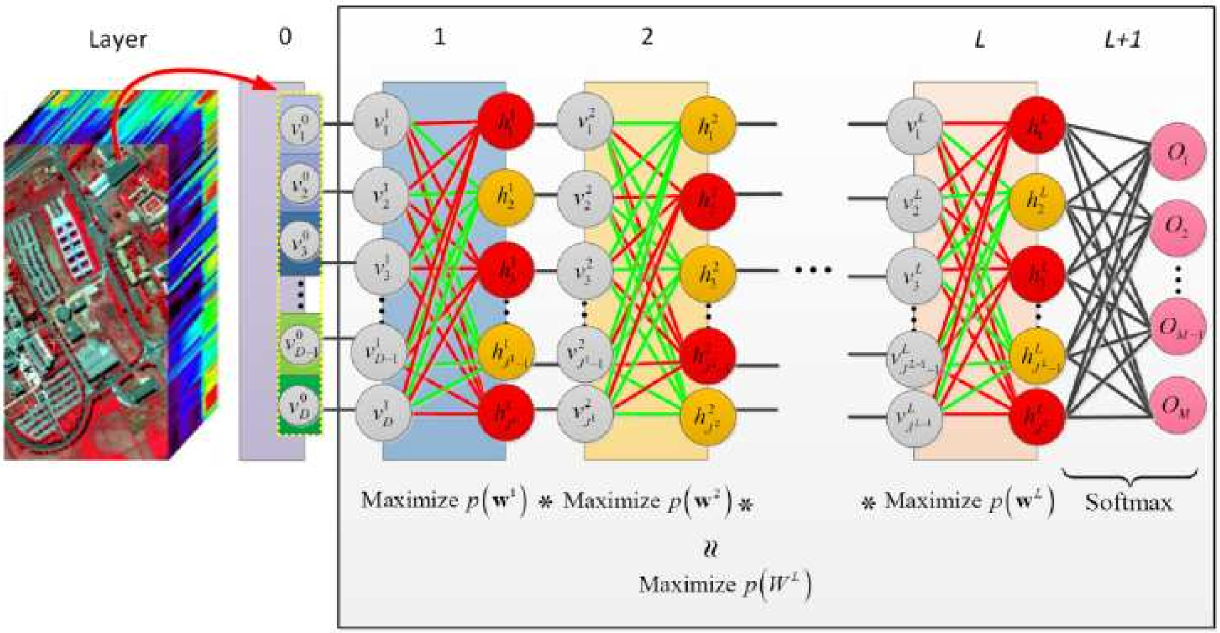
The pretraining phase learns the DBN to represent the spectral signature or features. The DBN for representation (without the last softmax layer) is a generative model. At the pretraining phase, the DBN is trained to reconstruct the unlabeled training data, and thus the pretraining can be performed in a unsupervised way. Corresponding to the hierarchical structure of the DBN, the pretraining can be implemented through a recursive greedy learning procedure. The main idea is to train the RBMs, which are stacked to form the DBN, layer by layer using the Contrastive Divergence algorithm

$$C(X) = \log p(X|\mathbf{w}^l) = \log \prod_{n=1}^{N} p(\mathbf{v}_n^l|\mathbf{w}^l) = \sum_{n=1}^{N} \log\left(\sum_{\mathbf{h}^l} p(\mathbf{v}_n^l, \mathbf{h}^l; \mathbf{w}^l)\right). \tag{7}$$

Each sample is a *D*-dimensional vector v_j^l = [$v_{n1}^l, v, \ldots,$ v $]^T$. Maximizing the cost function can be implemented by the gradient-based methods. The exact computation of with respect to the model's distribution. The *n*-step Contrastive Divergence method is used to approximate the gradient, and then the weights are updated as where (•) recons represents the expectation with respect to the distribution after *n* steps of block Gibbs sampling starting at the data

C. Fine-Tuning DBN

The pretraining phase fits the DBN to effectively represent the hyperspectral images. But for the hyperspectral image classification task at hand, the pretraining phase did not see any semantic information that the final task focuses on. Thus
the fine-tuning phase is used to further adjust the pretrained DBN to fit for the classification task. Fine-tuning the DB can be implemented through the back-propagation (BP) algorithm,which propagates the cost defined over the predictions and true labels of training samples from the last to first layer.



International Journal of Emerging Technology in Computer Science & Electronics (IJETCSE) ISSN: 0976-1353 Volume 25 Issue 5 – APRIL 2018 (SPECIAL ISSUE).

Fig3. Graphical representation of the diversified DBN in the unsupervised pretraining phase

III DIVERSIFYING DBN TO IMPROVE REPRESENTATION AND CLASSIFICATION

No matter the pretraining or fine-tuning could produce many similar hidden units and many "dead" (never responding) or "potential over-tolerant" (always responding) latent factors (neurons) (see Fig. 1 for illustration). As a consequence most of the computations are performed for redundant latent factors, which will decrease the DBN's description ability. To improve the performances of usual DBN, we propose to diversify the DBN and obtain a new DBN in both the pretraining and fine tuning phases.

A. Diversifying DBN in the Pretraining Phase

Following the pretraining framework, we develop a new training method to diversify DBN model. The diversity in this context means that the responses of latent units should be diverse. The diversification is obtained through diversifying RBMs, which are stacked to form the DBN, layer by layer. Therefore, the proposed training method is implemented through a recursive greedy learning procedure, which diversifies the weight parameters and indirectly the corresponding latent units. The diversifying method can be summarized in the following steps: 1) train the first layer of RBM with initial input to diversify the parameters to fit $\mathbf{w}1$ to training data;2) use the learned diversified parameters $\mathbf{w}1$ to compute the diversified hidden $\mathbf{h}1$; 3) use the hidden $\mathbf{h}1$ as input data to train next layer of RBM; and 4) repeat the procedure until all subsequent hidden layers $\mathbf{h}l$ are diversified for $l = 1, 2, \ldots, L$, where L is the number of hidden layers in DBN. We now describe how RBM is trained and meanwhile the method to diversify the weight parameters $\{\mathbf{w}l , l = 1, 2. . . L\}$. It should be mentioned that only weight parameters $\{\mathbf{w}l , l = 1, 2. . . L\}$ are diversified since they directly corresponds to the hidden units we aim to diversify. Fig. 2 shows the graphical representation of our method to add a diversity promoting prior to diversify the latent units in pretraining step. It should be noted that the training data contains only the input unlabeled observations and no semantic labels. Thus the diversifying learning at pretraining stage is unsupervised. Moreover the diversifying method is performed for every layer independently, and thus cannot use the possible dependence between the parameters of different layers to further improve DBN's representation ability. Training all layers together in an unsupervised way is generally very difficult since both the definition of reasonable cost functions and corresponding optimizations are not easy works. We will deal with this problem through performing the diversifying method in a supervised fine-tuning procedure, which propagates classification errors defined over the labeled training samples throughout the whole network and thus uses semantic information to adjust all parameters. Therefore the fine tuning phase can incorporate the labeled training samples to adjust the pretrained model for optimal representation and classification task in the sense of the given cost function. Same as that in pretraining phase, only weight parameters will be diversified. Moreover, it is reasonable to assume the weight parameters of different layers are independent

B. Diversifying DBN in the Fine-Tuning Phase

The previous section presented an unsupervised learning process to diversify a DBN. We propose to further diversify the pretrained model in a fine-tuning phase, which optimizes classification performance in the sense of a given loss function for labeled training data regularized by a diversity prior. To do so, a softmax layer is added to the end of the feature learning system. The layer acts as a classifier, and its output results are combined with the ground truth of the labeled training data to formulate the loss function on model parameters. A diversity promoting prior for model parameters is further incorporated in the loss function to formulate a final diversity promoting regularized objective. This objective can be efficiently optimized through BP. Fig. 3 shows the graphical representation of the diversified DBN in fine-tuning procedure, where the $p(\mathbf{w}l)$ is a diversity promoting prior of the parameters $\mathbf{w}l$. The layer acts as a classifier, and its output results are combined with the ground truth of the labeled training data to formulate the loss function on model parameters. A diversity promoting prior for model parameters is Moreover the diversifying method is performed for every layer independently, and thus cannot use the possible dependence between A DBN model constructed with a hierarchical series of RBMs could demonstrate the real power by capturing the important features in hyperspectral images. . It should be noted that the training data contains only the input unlabeled observations and no semantic labels. Thus the diversifying learning at pretraining stage is unsupervised. Moreover the diversifying method is performed for every layer independently, and thus cannot use the possible dependence between the parameters of different layers to further improve DBN's representation ability. Training all layers

International Journal of Emerging Technology in Computer Science & Electronics (IJETCSE) ISSN: 0976-1353 Volume 25 Issue 5 – APRIL 2018 (SPECIAL ISSUE).

together in an unsupervised way is generally very corresponding optimizations are not easy works. We will deal with this problem through performing the diversifying method in a supervised classification.

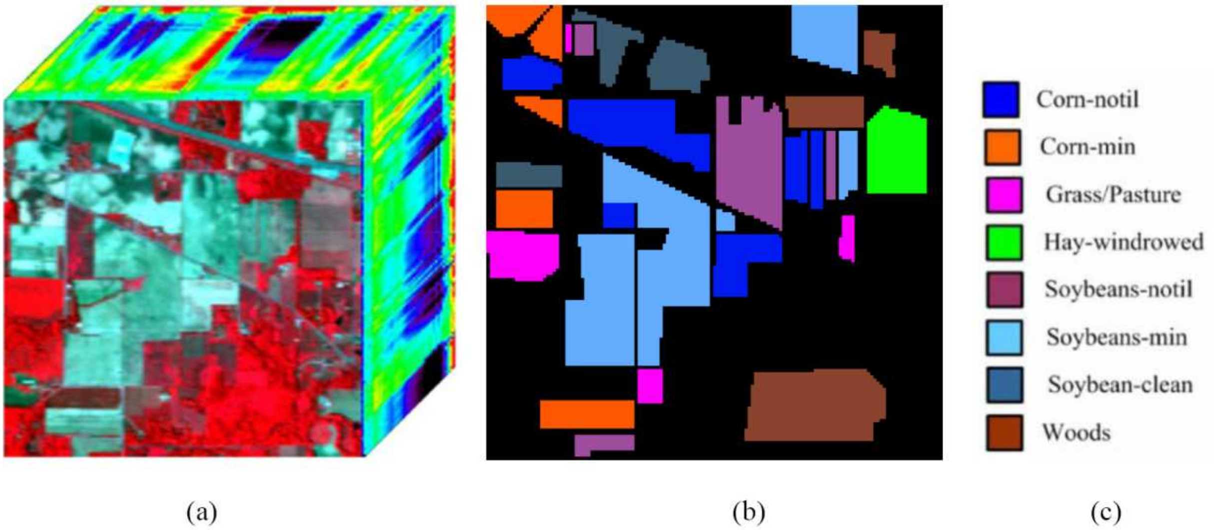


Fig4. Indian Pines data set. (a) Original image produced by the mixture of three bands. (b) Ground truth with eight classes. (c) Map color.

Fig. 3 see that the training data set contains both input observations and corresponding semantic labels. Thus, the diversifying learning at fine-tuning stage is supervised. Moreover, the BP algorithm can propagate the information from semantic labels, which are the main objective of hyperspectral image classification, back to the previous all layers, making the parameters optimally fit the training data. Meanwhile the proposed diversifying method implemented through the BP algorithm can jointly learn and diversify the parameters of different layers. Then the diversifying method in fine-tuning procedure has the potential to sufficiently use their dependences, and possibly further improve DBN's representation ability.

IV. EXPERIMENTAL RESULTS

A. *Experimental Setup*

The data set was gathered by Airborne Visible/Infrared Imaging Spectrometer sensor in northwestern Indiana. The original data set has 220 spectral channels in 0.4 to 2.45 μm region of the visible and infrared spectrum with a spatial resolution of 20 m × 20 m. As usual setup,
the 20 spectral bands were removed due to noise and water absorption, and the data set contains 200 bands of size 145 ×145 pixels. A three-band false color image and the ground truth data are presented in Fig. 4. The Indiana Pines data set has the original ground truth of sixteen different landcover classes. Eight classes have only few labeled training samples and thus were discarded to make the experimental analysis more statistically significant .The remaining eight classes were distributed with 8598 samples including both the spectral signals and their semantic labels. The pretraining is an unsupervised procedure and thus any spectrum (no need of its semantic label) can be used as the training samples. However, the fine-tuning is a supervised phase needing labeled samples. Therefore, to implement fine-tuning phase we randomly selected 200 samples as training samples for each class, and the remaining samples were used as test samples. The structure of the DBN for the Indian Pines data set is set as $200 - 50 - \cdot\ \cdot\ \cdot - 50 - 8$, which means the input layer has 200 nodes corresponding to the dimension of input data, the output layer has eight nodes corresponding to the number of classes, and all the middle layers (the number of the layers is specific for a given experimental setup) have 50 nodes. Details about the effects of different model structures on the DBN's performance.

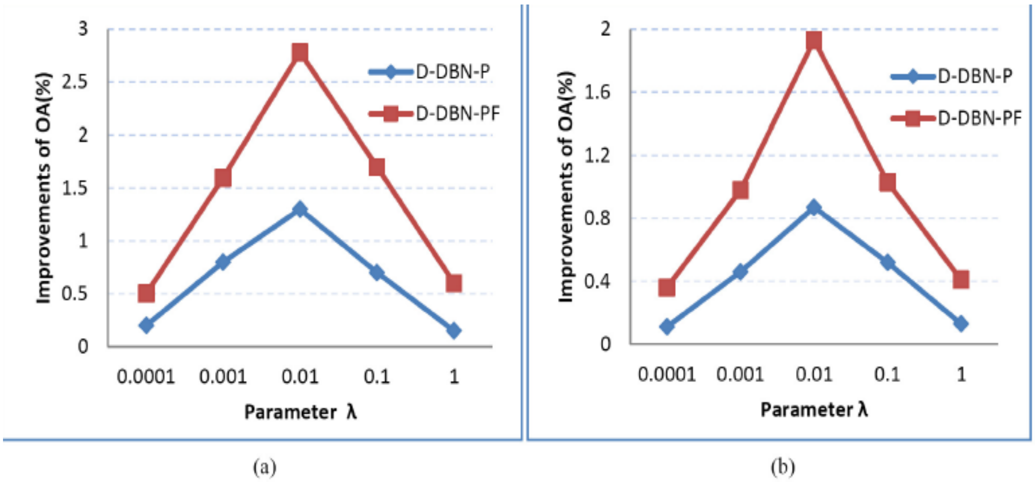


Fig. 7. Effects of parameter λ on the classification performances of the proposed diversified DBNs. (a) and (b) show the improvements of classification results (OA(%)) of D-DBN-P and D-DBN-PF compared with the original DBN over Indian Pines and Pavia University, respectively.

V. DISCUSSION AND CONCLUSION

This paper proposed a new diversified DBN model through introducing a diversity promoting prior into the pretraining and fine-tuning procedure of DBN. The introduced prior encouraged latent factors to be diverse and thus improved the performance of DBN on representation and classification of hyperspectral images. In addition, the proposed diversifying method can be efficiently implemented under the available recursive greedy and BP learning framework. The experiments over the real-world hyperspectral images demonstrated that the learned DBNs through the diversity regularization in both the pretraining and fine-tuning procedure have more diverse latent factors, which directly made the diversified DBNs obtain much better results than original DBNs and comparable or even better performances compared with other recent hyperspectral image classification methods.

REFERENCES

[1] U. Shaham et al., “A deep learning approach to unsupervised ensemble learning,” in Proc. Int. Conf. Mach. Learn., 2016, pp. 30–39.

[2] H. Xiong, A. J. Rodríguez-Sánchez, S. Szedmak, and J. Piater, “Diversity priors for learning early visual features,” Frontiers Comput. Neurosci., vol. 9, pp. 1–11, Aug. 2015.

[3] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, “Dropout: A simple way to prevent neural networks from overfitting,” J. Mach. Learn. Res., vol. 15, no. 1, pp. 1929–1958, 2014.

[4] P. Xie, Y. Deng, and E. P. Xing, “Diversifying restricted Boltzmann machine for document modeling,” in Proc. ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining, 2015, pp. 1315–1324.

[5] G. E. Hinton, S. Osindero, and Y.-W. Teh, “A fast learning algorithm for deep belief nets,” Neural Comput., vol. 18, no. 7, pp. 1527–1554,
2006.