STATISTICS, OPTIMIZATION AND INFORMATION COMPUTING
Stat., Optim. Inf. Comput., Vol. **5**, June 2017, pp 75–98.
Published online in International Academic Press (www.IAPress.org)

A Hybrid DBN and CRF Model for Spectral-Spatial Classification of Hyperspectral Images

Ping Zhong *, Zhiqiang Gong

National University of Defense Technology, China

(Received: 5 May 2017; Accepted: 16 May 2017)

Abstract Hyperspectral image classification plays an important role in remote sensing image analysis. Recent techniques have attempted to investigate the capabilities of deep learning approaches to tackle the hyperspectral image classification. This work shows how to further improve the hyperspectral image classification through using both a deep representation and contextual information. To implement this objective, this work proposes a new Conditional Random Field (CRF) model (named DBN-CRF) with the potentials defined over the deep features produced by a Deep Belief Network (DBN). The newly formulated DBN-CRF model takes advantage of the strength of DBNs in learning a good representation and the ability of CRFs to model contextual (spatial) information in both the observations and labels. Within a piecewise training framework, an efficient training method is proposed to train the whole DBN-CRF model end-to-end. This means that the parameters in DBN and CRF can be jointly trained and thus the proposed method can fully use the strength of both DBN and CRF. Moreover, in the proposed training method, the end-to-end training can be implemented with a standard back-propagation algorithm, avoiding the repeated inference usually involved in CRF training and thus is computationally efficient. Experiments on real-world hyperspectral data show that our method outperforms the most recent approaches in hyperspectral image classification.

Keywords Deep learning, Conditional random field, Deep belief network, Spectral-spatial classification, Hyperspectral image.

AMS 2010 subject classifications 68T05,68T45

DOI: 10.19139/soic.v5i2.309

1. Introduction

Over the past decades, hyperspectral imaging has experienced a significant success since it can get spatially and spectrally continuous data simultaneously. The hyperspectral images are used in a wide range of real-world applications, such as in psychology, urban planning, surveillance, agriculture, and disaster prevention and monitoring [1, 2, 3, 4, 5]. In these applications, the main techniques can be finally transformed into the classification tasks, which are equivalent to the assignment of each pixel with a land-cover label. For this reason, hyperspectral image classification is of particular interest in land-cover analysis research. Many popular methods have been developed for the hyperspectral image classification in the past several decades . One of the main approaches in this context is the use of only spectral information within each pixel within a popular classifier, such as multinomial logistic regression (MLR) [6, 7, 8], neural networks[9, 10], support vector machines (SVMs)[11, 12], graph method [13, 14], AdaBoost [15], Gaussian process approach [16] and random forest[17].

Such hyperspectral image classification methods take into consideration only spectral variations of pixels, ignoring important spatial correlations. However, the spatial correlations have been proved to be very useful for

*Correspondence to: National University of Defense Technology, China. E-mail: zhongping@nudt.edu.cn

ISSN 2310-5070 (online) ISSN 2311-004X (print)
Copyright © 2017 International Academic Press

image analysis in both remote sensing and computer vision communities [18]. Thus in recent years, also spectral-spatial classification methods have been proposed capturing both spectral and spatial information in a pixel and its neighboring pixels. The spectral-spatial methods shown significant advantages in terms of improving classification performance, and were developed following two main strategies. One is to incorporate contextual information into extracted features [19, 20, 21, 22, 23, 24, 25, 26], such as the morphical features [21, 22], tensor features [23], wavelet features [24], LBP features[25], and cooccurrence texture features [26].

The other is to model contextual information by intrinsic structures of the classifiers [27, 28, 29, 30, 31, 32, 33, 34, 35, 36]. The probabilistic graphical models have been developed as effective methods to incorporate the spectral and spatial information in an unified framework. In particular, Markov Random Fields (MRFs) and its variant Conditional Random Fields (CRFs) have obtained widespread success and have become one of the successful graphical models used in the literature of remote sensing [29, 30, 31, 32, 33, 34, 35, 36]. The MRF framework is effective to model contextual information in label image [29, 30]. But for computational tractability, the observed spectral vectors are assumed to be conditional independent, neglecting the contextual information in the observed data of a given class. As a variant of MRF, the CRF has intrinsic ability to incorporate the contextual information in both label and observed data in a principled manner. Morover, the contextual information is captured through intrinsic CRF structure, no need of complex modeling of the dependencies between the observations of neighboring sites. With these merits, CRFs usually show significant advantages over MRFs in terms of improving classification performance [31, 32, 33, 34, 35, 36]. This work will develop a new CRF model to further improve the spectral-spatial hyperspectral image classification.

On a separate track to this progress in the hyperspectral image classification methods, deep learning methods have been developed as effective methods to represent and classify hyperspectral images [37, 38, 39, 40, 41]. In fact, most of the popular methods mentioned previously can be deemed as shallow methods with only one or two processing layers. The researches in computer vision, however, demonstrated that deep architectures with more layers can potentially extract abstract and invariant features for better image classification [42]. Moreover, the deep learning methods have been immensely successful in many vision tasks such as image recognition [43], object detection[44] and semantic segmentation [45, 46, 47]. This motivates exploring the use of deep learning for the hyperspectral image representation and classification.

There are significant challenges, however, in adapting deep learning to the hyperspectral image classification. The standard approach to real-world hyperspectral image classification is to select some samples from a given image for classifier training, and then use the learned classifier to classify the remaining test samples in the same image [31]. This means that we usually do not have enough training image patches for a supervised contextual deep model, such as convolutional neural network (CNN). A few methods have been proposed to partially deal with the problem and make the deep learning fit to the hyperspectral image classification. In work [41], a fully unsupervised learning method of CNN has been developed to extract the deep sparse features based on the highly efficient enforcing population and lifetime sparsity algorithm. The work [40] relieved the problem through defining a 1-D CNN, which is performed over the set of spatially independent spectral vectors, not over the image patches as used in 2-D CNN. Thus the method cannot use the important spatial information. Another kind of methods to deal with the problem utilize the deep models with special structures, which naturally support the unsupervised training. The typical models include the multi-layer stacked autoencoder (SAE) [37, 38]and deep belief networks (DBN)[39]. They can be pre-trained through unsupervised ways, and have the ability to use the spatial information in observation through using a hand-crafting contextual features as the input. To sum up, the available deep learning methods usually focus on the feature representation, and some of them can model the spatial information in the observations. Most of them, however, are unaware of the spatial information lied in the labels (the final objective of the classification task), no saying of using the spatial information in labels to improve the previous deep learning.

As mentioned previously, the CRFs naturally have the ability to model the spatial information in both the observations and labels. Therefore, intuitively, CRFs can be used to improve the hyperspectral image classification results produced by a deep model. One approach to combine the CRFs and deep models is using the result of deep model as the input to a CRF, and thus the CRF inference is essentially used as a post-processing step [46]. In this setup, the two procedures, i.e., the CRF and deep model, are independent of each other, and thus the previous deep model cannot be trained to optimally fit to the later CRF. Work [45] and [47] proposed different end-to-end training

methods of joint CNN and CRF models for natural image semantic segmentation. For their method to hyperspectral image classification, however, these methods still face the problem of a lack of enough training samples for the CNN. Considering the merit of DBN on the unsupervised training, we turn to develop a new model to combine the DBN and CRF to deal with the problems simultaneously. Especifically, this work will propose a new hybrid DBN and CRF model (named DBN-CRF) with the CRFs potentials defined over the deep features from the DBN.

The remaining topic is how to train the proposed DBN-CRF model. In this work, we will develop an end-to-end learning solution to jointly train DBN and CRF to use the spectral and spatial information in both the observations and labels. Since CRFs for image analysis are large graphical models with loops, the computing can be expensive. In addition, the deep structure of DBN model and the characteristic of the standard real-world hyperspectral image classification make the end-to-end training even much harder. Motivated by the work in [31, 47], we will develop an efficient method within the piecewise framework to jointly train the DBN and CRF model. We will demonstrate that the training of the proposed DBN-CRF model can be finally implemented as the fine-tunings of two DBN models, corresponding to the unary and pairwise potentials respectively.

The proposed method demonstrates the following main merits. Firstly, the method avoids repeated inference typically needed in CRF training. Secondly, it can use the efficient back-propagation method directly to fine-tune the DBN so that it optimally fits for CRF inference. Finally, the method allows using only spatially separated spectral vectors as training samples and thus is feasible in real-world hyperspectral image classification tasks. To the authors knowledge, this work is the first that proposes end-to-end training of a joint DBN and CRF model to improve the hyperspectral image classification based on spectral-spatial information. The rest of the paper is arranged as follows. Section 2 proposes the hybrid DBN and CRF model for spectral-spatial hyperspectral image classification. The efficient end-to-end learning algorithm of the proposed DBN-CRF model is developed in Section 3. Section 4 utilizes the real-world hyperspectral image data sets to evaluate the proposed method. Finally our technique is concluded and discussed in Section 5.

2. A hybrid DBN and CRF model

Our goal is to develop a systematic approach to assign a label for each terrain site based on cube of observations, hoping that the labels are as close to the ground truth as possible. This problem can be formulated naturally under the statistical framework. Especially, we will incorporate the deep features from DBN into the CRF to formulate a new hybrid DBN and CRF (DBN-CRF) model, which takes advantage of the strength of DBN in deep learning representation and CRF in contextual (spatial) modeling in both the observations and labels.

2.1. DBNs for Deep Representation

A hyperspectral image usually has hundreds of spectral bands in a narrow bandwidth with fixed sample intervals. The abundant information presents the hyperspectral image the potential to discriminate the different land cover classes. However, the simple method using directly the spectral signature cannot release fully the potential of the hyperspectral image. In order to get a better classification map, it is necessary to extract an informative representation of the original spectral signature. The single hidden layer model is usually limited in capturing the features in the hyperspectral data, while multiple layers together could demonstrate the real power.

A DBN is such a model built of stack of a series of Restricted Boltzmann Machines (RBMs). The graphical representation of a DBN is shown in Fig. 1. In a DBN, the output of the previous RBM is used as the input data for the next RBM. Two adjacent layers have a full set of connections between them, but no two units in the same layer are connected. In theory, the output of every layer can be used as the extracted deep features. The output of the j -th hidden unit of the l -th layer of the network with the input x is

$$h_j^l(x, W^l, B^l) = \frac{1}{1 + \exp\left(-b_j^l - \sum_{i=1}^{J^{l-1}} w_{ij}^l h_i^{l-1}(x, W^{l-1}, B^{l-1})\right)} \tag{1}$$

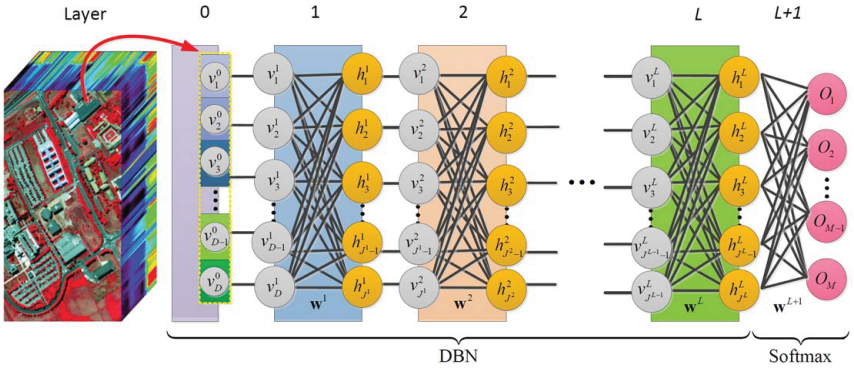

Figure 1. Illustration of the graphical representation of the DBN for hyperspectral image representation. A softmax layer is added to the DBN to use the semantic information of the training samples and the back-propagation to fine-tune the parameters of the DBN.

where $W^l = \left\{\mathbf{w}^1, \mathbf{w}^2, ..., \mathbf{w}^l\right\}$ and $B^l = \left\{\mathbf{b}^1, \mathbf{b}^2, ..., \mathbf{b}^l\right\}$ are the weight and bias parameters from the first to the l -th layer of the network,$\mathbf{w}^l = \left\{w_{ij}^l; i = 1, 2, ..., J^{l-1}, j = 1, 2, ..., J^l\right\}$ and $\mathbf{b}^l = \left\{b_j^l; j = 1, 2, ..., J^l\right\}$ are the weight and bias parameters of the l -th layer with J^l units.

To extract effectively the deep features, the parameters $W^L = \left\{\mathbf{w}^1, \mathbf{w}^2, ..., \mathbf{w}^L\right\}$ and $B^L = \left\{\mathbf{b}^1, \mathbf{b}^2, ..., \mathbf{b}^L\right\}$ of the DBN with L layers should be trained at first. The DBN training can be implemented by the unsupervised pre-training procedure [48]. Moreover, the pre-trained model can be further fine-tuned by a supervised training procedure. To implement the fine-tuning procedure, a softmax layer is usually incorporated into the DBN as the last layer (See Fig. 1). Since the softmax perform similar as a classifier, and thus the fine-tuning procedure can use the semantic labels of the training samples to fine-tune the model's parameters to fit for the final hyperspectral image classification. The output of the m -th unit (class) of the softmax layer is

$$O_m\left(\mathbf{x}, W^{L+1}\right) = \frac{\exp\left\{-\sum_{i=1}^{J^L} w_{im}^{L+1} h_i^L\left(\mathbf{x}, W^L, B^L\right)\right\}}{\sum_{n=1}^{M} \exp\left\{-\sum_{i=1}^{J^L} w_{in}^{L+1} h_i^L\left(\mathbf{x}, W^L, B^L\right)\right\}} = \frac{\exp\left\{-\left(\mathbf{w}_m^{L+1}\right)^T \mathbf{h}^L\left(\mathbf{x}, W^L, B^L\right)\right\}}{\sum_{n=1}^{M} \exp\left\{-\left(\mathbf{w}_n^{L+1}\right)^T \mathbf{h}^L\left(\mathbf{x}, W^L, B^L\right)\right\}} \tag{2}$$

where M is the number of classes and

$$\mathbf{w}_m^{L+1} = \left[w_{1m}^{L+1}, w_{2m}^{L+1}, ..., w_{J^L m}^{L+1}\right]^T$$

is the parameter vector for the m -th unit of the softmax layer. Equation (2) can be also deemed as the probability $P\left(y = m|\mathbf{x}, \theta\right)$ of the input data x m -th class.

The usual maximum likelihood (ML) method is used to fine-tune the parameters such that they minimize the negative log-likelihood

$$\Upsilon\left(\theta\right) = -\log P\left(\hat{Y}|\hat{X}, \theta\right) = -\sum_{k=1}^{K} \log\left(P\left(\hat{y}_k|\hat{\mathbf{x}}_k, \theta\right)\right) = -\sum_{k=1}^{K} \log\left(O_{\hat{y}_k}\left(\hat{\mathbf{x}}_k, W^{L+1}\right)\right) \tag{3}$$

where $O_{\hat{y}_k}\left(\hat{\mathbf{x}}_k, W^{L+1}\right)$ is the output of the $k-\text{th}$ training sample $\hat{\mathbf{x}}_k$ corresponding to the $\hat{y}_k-\text{th}$ class, that is

$$O_{\hat{y}_k}\left(\hat{\mathbf{x}}_k, W^{L+1}\right) = \frac{\exp\left\{-\sum_{m=1}^{M}\delta\left(\hat{y}_k = m\right)\left(\mathbf{w}_m^{L+1}\right)^T\mathbf{h}^L\left(\hat{\mathbf{x}}_k, W^L, B^L\right)\right\}}{\sum_{n=1}^{M}\exp\left\{-\left(\mathbf{w}_n^{L+1}\right)^T\mathbf{h}^L\left(\hat{\mathbf{x}}_k, W^L, B^L\right)\right\}} \tag{4}$$

$\hat{X} = \{\hat{\mathbf{x}}_1, \hat{\mathbf{x}}_2, ..., \hat{\mathbf{x}}_K\}$ is a set of training samples and $\hat{Y} = \{\hat{y}_1, \hat{y}_2, ..., \hat{y}_K\}$ be the corresponding labels, where $\hat{\mathbf{x}}_k = [\hat{x}_{k1}, \hat{x}_{k2}, ..., \hat{x}_{kD}]^T$ is a spectral signature with D bands,$\hat{y}_k$ takes the label value from $\{1, 2, ..., M\}$ and K is the number of training samples. The stochastic gradient descent (SGD) is usually used to optimize the objective function of (3) using the BP algorithm to compute the needed gradients. More details can be found in work [49].

2.2. CRFs for Spectral-Spatial Classification

In the context of hyperspectral image classification, observed data from an input image $\mathbf{x}$ is a set of spectral vectors $\mathbf{x} = \{\mathbf{x}_i, i \in S\}$, where $\mathbf{x}_i = [x_{i1}, x_{i2}, ..., x_{iD}]^T$ denotes a spectral vector associated with an image site i,S is the set of image sites, and D is the number of bands. The classification task is to assign (class) labels to image sites using the observed spectral vectors. The obtained label image is denoted by $\mathbf{y} = \{y_i, i \in S\}$,where y_i takes value in the set $\{1, 2, ..., M\}$ and M is the number of classes.

Within the Bayesian framework, hyperspectral image classification generally considers the posterior $P(\mathbf{y}|\mathbf{x})$.Within the classical MRF framework, the posterior is formulated as $P(\mathbf{y}|\mathbf{x}) \propto P(\mathbf{y})P(\mathbf{x}|\mathbf{y})$, where the prior distribution $P(\mathbf{y})$ is usually formulated as a Gibbs distribution, and y is said to be an MRF. $P(\mathbf{x}|\mathbf{y})$ is the likelihood and usually assumed as a factored form $P(\mathbf{x}|\mathbf{y}) = \prod_i P(\mathbf{x}_i|y_i)$ for computational feasibility. This assumption is equivalent to the conditional independence of the observed spectral values, and thus makes the MRF use only single-site spectral information to estimate the label of that site and neglect the contextual (spatial) information in the observed data.

In contrast, the CRF framework models directly the posterior of labels given the observed data as a Gibbs distribution:

$$P(\mathbf{y}|\mathbf{x}) = \frac{1}{Z(\mathbf{x}, \theta)}\exp\left\{-\sum_{c\in C}\psi_c(\mathbf{y}_c, \mathbf{x}, \theta)\right\} \tag{5}$$

where $Z(\mathbf{x}, \theta) = \sum_{\mathbf{y}}\exp\left\{-\sum_{c\in C}\psi_c(\mathbf{y}_c, \mathbf{x}, \theta)\right\}$ is the partition function and ψ_c is potential defined over clique c with parameters θ. The commonly used CRF models are formulated up to pairwise clique potentials only (assuming the potentials defined over other higher order cliques to be zeros), that is

$$P(\mathbf{y}|\mathbf{x}, \theta) = \frac{1}{Z(\mathbf{x}, \theta)}\exp\left\{-\sum_{i\in S}\psi_i(y_i, \mathbf{x}, \theta_u) - \sum_{i\in S}\sum_{j\in\eta_i}\psi_{ij}(y_i, y_j, \mathbf{x}, \theta_v)\right\} \tag{6}$$

where η_i is the set of neighbors of site i,$\psi_i(\bullet)$ and $\psi_{ij}(\bullet)$ are the unary and pairwise clique potentials with parameters θ_u and θ_v, respectively. Then the model parameters are $\theta = \{\theta_u, \theta_v\}$.

With the formulation of (5) or (6), the CRF model avoids the problem of explicit modeling of likelihood in MRF framework and thus has advantages over MRF, particularly on the flexible modeling of contextual information. We can observe from (5) that the potential $\psi_c(\mathbf{y}_c, \mathbf{x}, \theta)$ in CRF model is defined over the labels of a clique c and the whole input observation $\mathbf{x}$ and thus in theory, the CRF model has the ability to capture the contextual information in both the labels and observed data.

2.3. DBN-CRF for Deep Spectral-Spatial Classification

It is noted from the previous discussions that the merits of CRFs derive mainly from the flexibility of potentials, and thus defining the potentials is an important issue to formulate a CRF model. We can define the right potentials according to the tasks. For example, the unary and pairwise clique potentials in CRFs can be viewed as arbitrary local discriminative classifiers. This allows one to use domain-specific discriminative classifiers rather

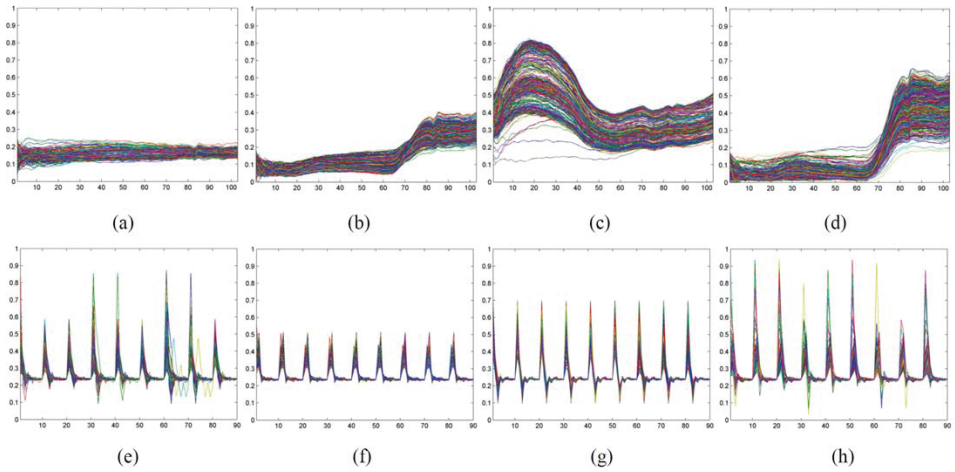

Figure 2. Spectral ((a)-(d)) and spectral-spatial ((e)-(h)) signatures of selected four classes (asphalt, meadows, sheets and trees) from the University of Pavia data set with 103 bands.

than restricting the potentials to a specific form. Another kind of popular potentials are defined over the various features. Then the state-of-the-art spectral or image features in the literature can be easily incorporated into the CRF to improve the classification performance.

In this work, viewing the recent immense success of the deep representation in computer vision, we introduce the features extracted by the deep learning method into the CRF for hyperspectral image classification. To implement this task, we propose a new DBN-CRF model, which takes advantage of the strength of DBNs in deep learning representation and CRFs in contextual (spatial) modeling in both the observations and labels.

1) Unary Potentials of DBN-CRF. The unary potential is used to model and represent observations for single image site. For the task of hyperspectral image classification, the observed spectral values show complex statistics, and thus the simple features cannot fully represent the signals. Fig. 2 illustrates this point. In Fig. 2(a)-(d), each plot shows the spectral signatures of a cover type from the University of Pavia data set. The details of this data set will be depicted in Section 4. Fig. 2(e)-(h) further illustrate a kind of usual spectral-spatial signatures for the selected classes [39]. The spectral-spatial signature of one site were obtained by performing the PCA transformation over the hyperspectral image at first, then extracting the first 10 components as the new representation of the hyperspectral image, and finally concatenating the new representation in a 3×3 window centered at the site. It can be easily calculated that the spectral-spatial signature is a 90-dimentional vector. We can see from these plots that the curve of each land-cover class has its own visual shape. Although the curves of different classes show some difference with each other, how to design (usually hand-crafted) the features to represent these signals, especially the essential specifics and difference, is still a very difficult problem. Fortunately, the deep learning method demonstrated in recent years that it can automatically learn the representation and mine the features similar with the primary human vision. Therefore, the proposed CRF in this work uses the deep features learned by the DBN to capture the complex statistics of the hyperspectral data.

The unary potential is formulated as follows:

$$\psi_i\left(y_i, \mathbf{x}, U, W_u^L, B_u^L\right) = \sum_{m=1}^{M} \delta\left(y_i = m\right) \mathbf{u}_m^T \mathbf{h}^L\left(\mathbf{x}, W_u^L, B_u^L; i\right) \tag{7}$$

Here $\delta(\bullet)$ is the indicator function, which equals 1 if the input is true and 0 otherwise, M is the number of classes,$U = \{\mathbf{u}_m; m = 1, ..., M\}$ is the set of unary parameters, $\mathbf{h}^L\left(\mathbf{x}, W_u^L, B_u^L; i\right)$ is the vector of deep features extracted by the DBN from previous section for site i, i.e., the output of the last second layer ($L-$th layer) of the structure shown in Fig. 3, $W_u^L = \left\{\mathbf{w}_u^1, ..., \mathbf{w}_u^L\right\}$ and $B_u^L = \left\{\mathbf{b}_u^1, ..., \mathbf{b}_u^L\right\}$ are the weight and bias parameters from

first to $L-\text{th}$ layers of DBN for the unary potentials, $\mathbf{w}_u^l$ and $\mathbf{b}_u^l$ are the weight and bias parameters of $l-\text{th}$ layer, $\mathbf{u}_m = \left[u_{1m}, u_{2m}, ..., u_{J_u^L m}\right]^T$ is the parameter vector of CRF for the $m-\text{th}$ class and J_u^L is the number of the units of the $L-\text{th}$ layer of DBN for the unary potentials.

2) Pairwise Potentials of DBN-CRF. To define the pairwise potential, we mainly focus on its ability to encode contextual information. There are complex interactions in neighboring spectral vectors: a spectral vector belonging to a type of terrain is highly dependent upon its neighbors, since in a type of terrain, spatial variations of pixel spectral vector may follow some underlying patterns rather than being random; moreover for different classes the underlying patterns may be different. Although the classical Ising/Potts models usually used as the pairwise clique potentials in MRF can model the contextual information in labels, they do not permit the use of observed data, and thus cannot capture the contextual information in the observed data.

A generalized Ising model is used to model the pairwise potential in this work, i.e.,

$$\psi_{ij}\left(y_i, y_j, \mathbf{x}, V, W_p^L, B_p^L\right) = \sum_{m=1}^{M}\sum_{n=1}^{M} \delta\left(y_i = m\right)\delta\left(y_j = n\right)\mathbf{v}_{mn}^T \mathbf{h}^L\left(\mathbf{x}, W_p^L, B_p^L; i, j\right) \tag{8}$$

Here $V = \{\mathbf{v}_{mn}; m, n = 1, ..., M\}$ is the set of pairwise parameters, $\mathbf{h}^L\left(\mathbf{x}, W_p^L, B_p^L; i, j\right)$ is a feature vector extracted from observation $\mathbf{x}$ based on the DBN for a pair of sites (i, j),$W_p^L = \left\{\mathbf{w}_p^1, ..., \mathbf{w}_p^L\right\}$ and $B_p^L = \left\{\mathbf{b}_p^1, ..., \mathbf{b}_p^L\right\}$ are the weight and bias parameters from first to $L-\text{th}$ layers of the DBN for the pairwise potentials, and $\mathbf{w}_p^l$ and $\mathbf{b}_p^l$ are the weight and bias parameters of the $l-\text{th}$ layer. In this work, pairwise feature vector $\mathbf{h}^L\left(\mathbf{x}, W_p^L, B_p^L; i, j\right)$ is obtained by concatenating all elements of the inputs for sites i and j at first, and then using the DBN to extract the deep feature vector from the concatenated input. For example, if the input for each site i is simply the $D-$ dimentional spectral vector $\mathbf{x}_i = [x_{i1}, x_{i2}, ..., x_{iD}]^T$, the input to the DBN for the site pair (i, j) is a $2D-$ dimentional vector $\left[\mathbf{x}_i^T, \mathbf{x}_j^T\right]^T = [x_{i1}, x_{i2}, ..., x_{iD}, x_{j1}, x_{j2}, ..., x_{jD}]^T$. The vector $\mathbf{v}_{mn}$ in (8) is the parameters of CRF for the pair class (m, n) and has the dimension as J_p^L , i.e., the number of units in $L-\text{th}$ layer of DBN for the pairwise potentials.

Our formulation of pairwise potentials is different from the classical Ising/Potts model, which uses only the contextual information of labels to enforce neighborhood smoothness. The formulation (8) in this work explicitly depends on the whole observed data $\mathbf{x}$ and the neighboring labels and thus can capture the two kinds of contextual information in both the labels and observed data. Furthermore, the contextual information in labels is based on the idea of pairwise discrimination of the observed data, making it be data-adaptive instead of being fixed as a priori in MRFs.

In addition, the formulation of pairwise potentials is also very different from that used in [47]. The formulation in [47] concatenates two deep features from sites i and j to get the feature vector of the site pair (i, j) , while our method concatenates the input of sites i and j at first, and then uses only one DBN to extract the deep feature for the pair input. Our processing method can fully model and use the contextual information of the pair observations of two classes. In addition, the most obvious difference between the formulations is that our formulation has the extra CRF parameter vector $\mathbf{v}_{mn}$ for the pair potentials and $\mathbf{u}_m$ in the unary potentials, which could make the CRF have more discriminative ability. Moreover the extra CRF parameters $\mathbf{u}_m$ and $\mathbf{v}_{mn}$ bring our method the merit of convenient and efficient end-to-end training procedure: the training of our CRF model can be implemented as two efficient DBN trainings and thus the CRF and DBN in the proposed DBN-CRF can be jointly learned. This point will be demonstrated in Section 4.

3) Graphical Representation of the Proposed DBN-CRF. Fig. 3 shows the graphical representation of the proposed DBN-CRF for spectral-spatial hyperspectral image classification. In order to make the figure clearer, only one-dimensional CRF model is presented, and a two-dimensional CRF is actually used for the hyperspectral image classification. In the figure, only the definitions of the unary site i and pair sites $(i, i+1)$ are completely illustrated and other sites have the similar structures.

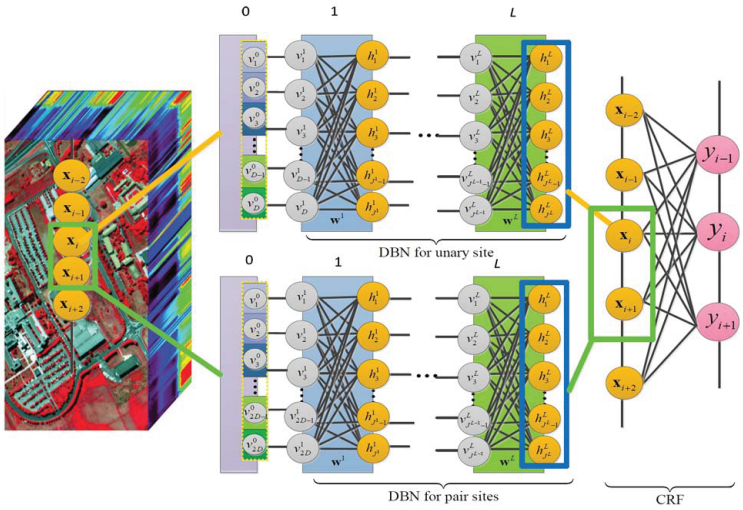

Figure 3. Graphical representaton of the proposed DBN-CRF for spectral-spatial hyperspectral image classification.

3. Efficient training of the DBN-CRF model

The parameters needed to be estimated in the proposed DBN-CRF model include $W = \left\{W_u^L, W_p^L\right\} = \left\{\mathbf{w}_u^1, ..., \mathbf{w}_u^L, \mathbf{w}_p^1, ..., \mathbf{w}_p^L\right\}$ and $B = \left\{B_u^L, B_p^L\right\} = \left\{\mathbf{b}_u^1, ..., \mathbf{b}_u^L, \mathbf{b}_p^1, ..., \mathbf{b}_p^L\right\}$ in DBNs, and $U = \{\mathbf{u}_m; m = 1, ..., M\}$ and $V = \{\mathbf{v}_{mn}; m, n = 1, ..., M\}$ in unary and pairwise potentials of CRF. Our objective is to develop an end-to-end training method to estimate the parameters in DBN and CRF simultaneously. Using the CRF definition in (5), classical ML parameter estimation method chooses parameter values $\theta = \{W, B, U, V\}$ such that they minimize the negative log-likelihood

$$\Upsilon(\theta) = -\sum_{q=1}^{Q} \log P\left(\hat{\mathbf{y}}^q | \hat{\mathbf{x}}^q, \theta\right) = -\sum_{q=1}^{Q} \sum_{c \in C} \log\left\{-\psi_c\left(\hat{\mathbf{y}}_c^q, \hat{\mathbf{x}}^q, \theta\right)\right\} + \sum_{q=1}^{Q} \log Z\left(\hat{\mathbf{x}}^q, \theta\right) \tag{9}$$

where $\hat{\mathbf{x}} = \{\hat{\mathbf{x}}^q, q = 1, 2, ..., Q\}$ are Q i.i.d. training images and $\hat{\mathbf{y}} = \{\hat{\mathbf{y}}^q, q = 1, 2, ..., Q\}$ are the label images.

Exact ML estimation is intractable in general due to combinatorial size of the label space in computing the partition function $Z\left(\hat{\mathbf{x}}^q, \theta\right)$. In addition, the ML method needs the whole training images. However as aforementioned, the most usual task in hyperspectral image classification is to select some samples from a given image for classifier training, and then use the learned classifier to classify the whole given image. Therefore the ML can not be directly used for the task at hand. A feasible method is to estimate the parameters locally. On the one hand, this method can approximate the partition function $Z\left(\hat{\mathbf{x}}^q, \theta\right)$ efficiently, on the other hand, it uses only local training samples and thus is suitable for the usual hyperspectral image classification task. Pseudolikelihood estimation is a classical local estimation method. But as demonstrated in [50], its accuracy can be poor on some data. In this paper, we focus on an alternative named piecewise training framework [50]. Within the framework, an efficient training method will be developed for the proposed DBN-CRF model.

3.1. Piecewise Training for DBN-CRF

The intuition of piecewise training is that if each factor $\psi_a\left(\mathbf{y}_a, \mathbf{x}, \theta\right)$ of an objective function can independently predicts $\mathbf{y}_a$ and $\mathbf{x}$ accurately, then the prediction of the global factor graph will also be accurate [32, 47, 50]. Let $a \in A$ be a graph factor composed of a set of sites, and A is the set of all graph factors. In this work, the objective function is divided as multiple factors according to the types of the cliques. Let C_s be the set of the type

of cliques with s sites. Then, $a \in A$ is a clique c in the set $A = \{C_s, s = 1, 2, ...\} \equiv C$. The objective function in (9) is approximated in piecewise training framework with this special division for CRF as

$$\Upsilon_{PW}(\theta) = -\sum_{c \in \hat{C}} \log \frac{\exp\{-\psi_c(\hat{\mathbf{y}}_c, \hat{\mathbf{x}}, \theta)\}}{\sum_{\mathbf{y}_c} \exp\{-\psi_c(\mathbf{y}_c, \hat{\mathbf{x}}, \theta)\}} \tag{10}$$

where $\hat{C} = \left\{\hat{C}_s, s = 1, 2, ...\right\}$ is the set of all selected cliques, $\hat{C}_s$ denotes the set of cliques selected for training and $\hat{\mathbf{y}}_c$ is the labels of training samples over clique c. The meaning of "piece" corresponds to a term in (10), and that term would be the exact likelihood of the piece if the rest of the graph were omitted.

In this work, the CRF in (6) with only unary and pair potentials is used. Let $\left\{\hat{X}, \hat{Y}\right\} = \{\hat{\mathbf{x}}_k, \hat{y}_k; k = 1, 2, ..., K\}$ be a set of training samples, where $\hat{\mathbf{x}}_k$ is a selected training spectral vector, $\hat{y}_k$ is the corresponding label and K is the number of training samples. The objective function of the piecewise training framework of the proposed DBN-CRF can be written as the following factorized form:

$$\Upsilon_{PW}(\theta) = -\sum_{i \in \hat{C}_1} \log \frac{\exp\{-\psi_i(\hat{y}_i, \hat{\mathbf{x}}, \theta_u)\}}{\sum_{\mathbf{y}_i} \exp\{-\psi_i(y_i, \hat{\mathbf{x}}, \theta_u)\}} - \sum_{(i,j) \in \hat{C}_2} \log \frac{\exp\{-\psi_{ij}(\hat{y}_i, \hat{y}_j, \hat{\mathbf{x}}, \theta_p)\}}{\sum_{y_i, y_j} \exp\{-\psi_{ij}(y_i, y_j, \hat{\mathbf{x}}, \theta_p)\}} \equiv \Upsilon_{\theta_u} + \Upsilon_{\theta_p} \tag{11}$$

where $\theta_u = \left\{W_u^L, B_u^L, U\right\}$ and $\theta_p = \left\{W_p^L, B_p^L, V\right\}$ are the sets of parameters of the unary and pair potentials respectively, and Υ_{θ_u} and Υ_{θ_p} denote the first and second term at the right of equal sign in (11). Equation (11) shows that, under the piecewise training framework with the special division, DBN-CRF training can be implemented by independently training the local classifiers over each kind of cliques. Furthermore, we will demonstrate that, with the potentials defined as (7) and (8) the local classifiers are exactly two DBN models with extra softmax layers.

3.2. Training for Unary Clique Potentials

With the definition of the unary potential as (7), the objective function Υ_{θ_u} in (11) can be written as

$$\begin{aligned}
\Upsilon_{\theta_u} &= -\sum_{i \in \hat{C}_1} \log \frac{\exp\left\{-\sum_{m=1}^{M} \delta(\hat{y}_i = m) \mathbf{u}_m^T \mathbf{h}^L\left(\hat{\mathbf{x}}, W_u^L, B_u^L; i\right)\right\}}{\sum_{y_i} \exp\left\{-\sum_{m=1}^{M} \delta(y_i = m) \mathbf{u}_m^T \mathbf{h}^L(\hat{\mathbf{x}}, W_u^L, B_u^L; i)\right\}} \\
&= -\sum_{i \in \hat{C}_1} \log \frac{\exp\left\{-\sum_{m=1}^{M} \delta(\hat{y}_i = m) \mathbf{u}_m^T \mathbf{h}^L\left(\hat{\mathbf{x}}, W_u^L, B_u^L; i\right)\right\}}{\sum_{m=1}^{M} \exp\{-\mathbf{u}_m^T \mathbf{h}^L(\hat{\mathbf{x}}, W_u^L, B_u^L; i)\}} \\
&= -\sum_{i \in \hat{C}_1} \log\left(O_{\hat{y}_i}\left(\hat{\mathbf{x}}, W_u^{L+1}; i\right)\right)
\end{aligned} \tag{12}$$

with

$$O_{\hat{y}_i}\left(\hat{\mathbf{x}}, W_u^{L+1}; i\right) = \frac{\exp\left\{-\sum_{m=1}^{M} \delta(\hat{y}_i = m) \mathbf{u}_m^T \mathbf{h}^L\left(\hat{\mathbf{x}}, W_u^L, B_u^L; i\right)\right\}}{\sum_{m=1}^{M} \exp\{-\mathbf{u}_m^T \mathbf{h}^L(\hat{\mathbf{x}}, W_u^L, B_u^L; i)\}} \tag{13}$$

Compared with (3) and (4), (12) is just the objective function of fine-tuning DBN with the weight parameters $\mathbf{w}_u^{L+1}$ of the last layer as $\mathbf{u}_m$. Therefore, the available usual algorithms can be directly used to learn the "piece" corresponding to the unary potential of the proposed DBN-CRF. Especially, the learning can be implemented by the SGD, with the BP algorithm to efficiently compute the gradients.

3.3. Training for Pairwise Clique Potentials

Based on the formulation of pairwise clique potential in (8), the second term Υ_{θ_p} in (11) can be written as

$$\begin{aligned}\Upsilon_{\theta_p} &= -\sum_{(i,j)\in\hat{C}_2}\log\frac{\exp\{-\psi_{ij}(\hat{y}_i,\hat{y}_j,\hat{\mathbf{x}},\theta_p)\}}{\sum_{y_i,y_j}\exp\{-\psi_{ij}(y_i,y_j,\hat{\mathbf{x}},\theta_p)\}}\\ &= -\sum_{(i,j)\in\hat{C}_2}\log\frac{\exp\Big\{-\sum\limits_{m=1}^{M}\sum\limits_{n=1}^{M}\delta(y_i=m)\delta(y_j=n)\mathbf{v}_{mn}^T\mathbf{h}^L\big(\hat{\mathbf{x}},W_p^L,B_p^L;i,j\big)\Big\}}{\sum\limits_{m=1}^{M}\sum\limits_{n=1}^{M}\exp\big\{-\mathbf{v}_{mn}^T\mathbf{h}^L\big(\hat{\mathbf{x}},W_p^L,B_p^L;i,j\big)\big\}}\\ &= -\sum_{(i,j)\in\hat{C}_2}\log\Big(O_{(\hat{y}_i,\hat{y}_j)}\big(\mathbf{x},W_p^{L+1};i,j\big)\Big)\end{aligned} \tag{14}$$

with

$$O_{(\hat{y}_i,\hat{y}_j)}\big(\mathbf{x},W_p^{L+1};i,j\big)=\frac{\exp\left\{-\sum\limits_{m=1}^{M}\sum\limits_{n=1}^{M}\delta\left(\hat{y}_i=m\right)\delta\left(\hat{y}_j=n\right)\mathbf{v}_{mn}^T\mathbf{h}^L\left(\hat{\mathbf{x}},W_p^L,B_p^L;i,j\right)\right\}}{\sum\limits_{m=1}^{M}\sum\limits_{n=1}^{M}\exp\left\{-\mathbf{v}_{mn}^T\mathbf{h}^L\left(\hat{\mathbf{x}},W_p^L,B_p^L;i,j\right)\right\}} \tag{15}$$

It can be easily noted through the comparison between (4) and (15) that $O_{(\hat{y}_i,\hat{y}_j)}\left(\hat{\mathbf{x}},W_p^{L+1};i,j\right)$ is just the output of a DBN with the last layer as a softmax classifier for M^2 classes. Therefore, similar to the learning of the unary potential, (14) is also the objective function of fine-tuning DBN with the weight parameters $\mathbf{w}_p^{L+1}$ of the last layer as $\mathbf{v}_{mn}$. As before, the SGD with BP training algorithm can be also used to minimize (14).

We further analyze the usual real-world hyperspectral image classification. It is time-consuming to label the pixels near the spatial borders between different classes. Therefore, we usually do not have enough pair samples with different labels to learn the parameters $\mathbf{v}_{mn}$ $(m\neq n)$. To deal with the problem and meanwhile to make the training procedure more efficient, as the setting in [31], the parameter vector is set to $\mathbf{0}$ if $m\neq n$, and we only consider parameters $\{\mathbf{v}_{mm}, m=1,2,...,M\}$. Then (15) can be equivalently written as

$$O_{(\hat{y}_i,\hat{y}_j)\equiv m}\big(\mathbf{x},W_p^{L+1};i,j\big)=\begin{cases}\dfrac{\exp\Big\{-\sum\limits_{m=1}^{M}\sum\limits_{n=1}^{M}\delta(\hat{y}_i=m)\delta(\hat{y}_j=m)\mathbf{v}_{mn}^T\mathbf{h}^L\big(\hat{\mathbf{x}},W_p^L,B_p^L;i,j\big)\Big\}}{\tau+\sum\limits_{n=1}^{M}\exp\big\{-\mathbf{v}_{nn}^T\mathbf{h}^L\big(\hat{\mathbf{x}},W_p^L,B_p^L;i,j\big)\big\}}, & \text{if } m\leq M\\ \dfrac{\tau}{\tau+\sum\limits_{n=1}^{M}\exp\big\{-\mathbf{v}_{nn}^T\mathbf{h}^L\big(\hat{\mathbf{x}},W_p^L,B_p^L;i,j\big)\big\}}, & \text{if } m=M+1\end{cases} \tag{16}$$

where $\tau=M(M-1)$ is a constant, and $(\hat{y}_i,\hat{y}_j)\equiv m$ denotes the fact $\hat{y}_i=\hat{y}_j=m$ if $m\leq M$, while $(\hat{y}_i,\hat{y}_j)\equiv m$ means the fact $\hat{y}_i\neq\hat{y}_j$ if $m=M+1$. Therefore, (16) is the output of a DBN with the last layer as a softmax classifier for $M+1$ classes. Consequently (14) is exactly the objective of fine-tuning DBN with only $M+1$ classes.

To sum up, this section developed an end-to-end learning solution to jointly train DBN and CRF in the proposed DBN-CRF model. Finally the learning of the proposed DBN-CRF model can be implemented efficiently as the learning of two DBNs with the last layer as a softmax classifier. The graphical representation of the proposed efficient learning method is shown in Fig. 4. We can see from the figure that the training data set in the usual training method are some spatial continuous samples, while the training data set in the usual training method are some spatial continuous samples, while the proposed training method allows selecting the samples randomly to construct the training samples for different pieces. This makes the proposed method have more flexibilities to fit to the real-world hyperspectral image classification. In addition, the figure also shows that the DBN-CRF learning divided as the learning of different DBNs corresponding to different potentials can be implemented through a parallel way.

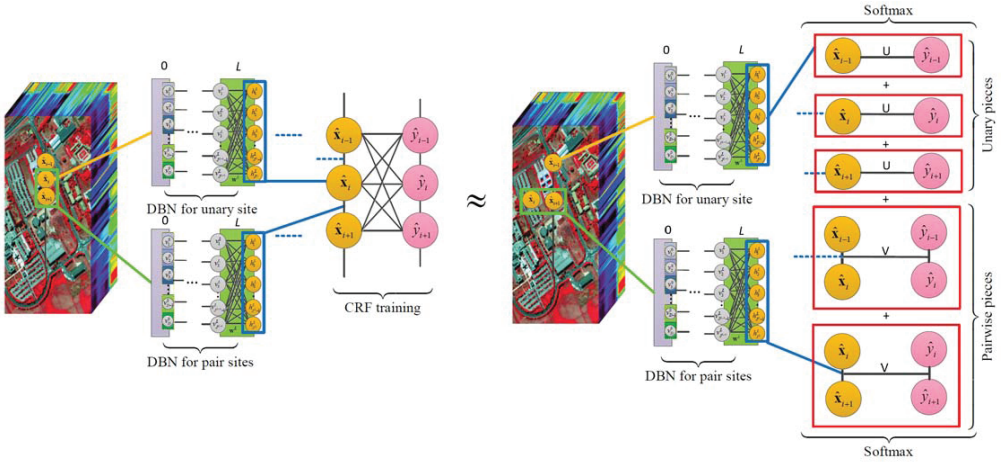

Figure 4. Graphical model of the piecewise training for DBN-CRF. The left of equal sign is original DBN-CRF training, right is the version trained by the proposed piecewise training method. This figure presents only up to pairwise factors.

3.4. Model Combination in Inference

The previous subsections demonstrate that the proposed training method for the DBN-CRF divides the model into two DBNs with the last layer as a softmax classifier, and they are trained independently. However, this paralleled training method may lead to problems with over-counting during inference [51]. It is difficult to assess analytically the degree of over-counting introduced by dependences between the different terms in DBN-CRF model. Instead, as our previous work [31] and [32] , this work introduces the scalar powers for each term. Thus, given the input of the test image $\mathbf{x}$ and the learned parameters $\tilde{W} = \left\{\tilde{W}_u^L, \tilde{W}_p^L\right\} = \left\{\tilde{\mathbf{w}}_u^1, ..., \tilde{\mathbf{w}}_u^L, \tilde{\mathbf{w}}_p^1, ..., \tilde{\mathbf{w}}_p^L\right\}$,$\tilde{B} = \left\{\tilde{B}_u^L, \tilde{B}_p^L\right\} = \left\{\tilde{\mathbf{b}}_u^1, ..., \tilde{\mathbf{b}}_u^L, \tilde{\mathbf{b}}_p^1, ..., \tilde{\mathbf{b}}_p^L\right\}$,$\tilde{U} = \{\tilde{\mathbf{u}}_m; m = 1, ..., M\}$ and $\tilde{V} = \{\tilde{\mathbf{v}}_{mn}; m, n = 1, ..., M\}$, the independently trained models can be combined in the CRF to infer the label image as

$$\begin{aligned}\tilde{\mathbf{y}} &= \arg\max_{\mathbf{y}} P\left(\mathbf{y}|\mathbf{x}, \tilde{\theta}\right) \\ &= \arg\max_{\mathbf{y}} \exp\left\{-\lambda_1\left[\sum_{i\in S}\psi_i\left(y_i, \mathbf{x}, \tilde{U}, \tilde{W}_u^L, \tilde{B}_u^L\right)\right] - \lambda_2\left[\sum_{i\in S}\sum_{j\in\eta_i}\psi_{ij}\left(y_i, y_j, \mathbf{x}, \tilde{V}, \tilde{W}_p^L, \tilde{B}_p^L\right)\right]\right\}\end{aligned} \tag{17}$$

where λ_1 and λ_2 are the fixed powers for the unary and pairwise clique potentials, respectively. Because the fixed powers λ_1 and λ_2 function in the same manner by assigning weights to their corresponding potentials, λ_2 is fixed to be one and only λ_1 is required to be adjusted. The optimal selection of the power is an area of active research. Same as the work in [51] does, this paper optimizes the power discriminatively using the cross validation. Then, based on the learned parameters and selected power, the inference of the form (17) can be efficiently implemented by loopy belief propagation (LBP) [52].

4. Experimental results

4.1. Experimental Data sets

In our experiments, two hyperspectral data sets were applied to evaluate the proposed method. A hyperspectral image can be considered as a cube of observed pixels, made up of several 2-D arrays. Figs. 5 and 6 show the

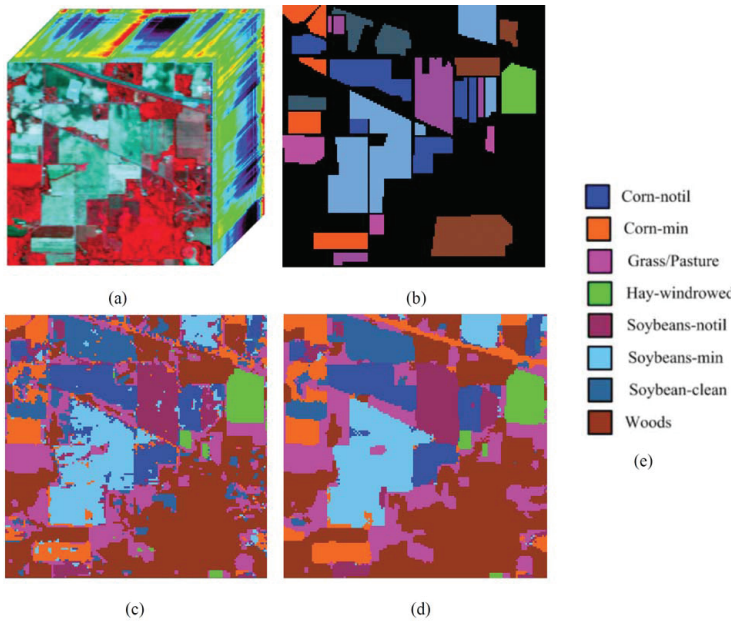

Figure 5. Indian Pines data set and its example classification results. (a) Original image produced by the mixture of three bands. (b) Ground truth with eight classes. (c) and (d) The classification results of DBN-CRF-U and DBN-CRF. (e) Map color.

two hyperspectral images, representing different environments in remote sensing: the Indian Pines data set show a mixed vegetation site, while the University of Pavia data set represents a typical urban site.

1) Indian Pines: AVIRIS (Airborne Visible/Infrared Imaging Spectrometer) image of the Indian Pines test site in NW Indian taken on June 12, 1992. This image contains 145×145 pixels and 220 bands. As an example, the mixture image of three bands from the data set is shown in Fig. 5(a). The major advantage in using this data set is the availability of a reference map [see Figs. 5(b) and (e)] prepared from the field surveys conducted at the time of image acquisition. We also noted that there are several bands, such as the bands with the index between 150 and 170, show weak correlations with others because affected by atmospheric problems. Thus we discarded 20 of the original 220 spectral channels.

2) University of Pavia : This data set was taken by a sensor known as the reflective optics system imaging spectrometer (ROSIS-3) over the city of Pavia, Italy. The image contains 610×340 pixels and 115 bands collected over 0.43-0.86 μm range of the electromagnetic spectrum. In the available data online, some bands were removed due to noise and the remaining 103 channels were used for the classification in this work. Nine land-cover classes were selected, which are shown in Fig. 6.

4.2. Experimental Setup

To evaluate the proposed method, the available labeled samples were randomly divided into training set and test set. The Indiana Pines data set has sixteen different land-cover classes available in the original ground truth. To make the experimental analysis more significant from the statistical viewpoint, as the setting in [40], eight classes were discarded since only few training samples were available. The remaining eight classes were distributed by 8598 elements. For each class, 200 samples were randomly selected as the training samples and the remaining samples were used as test samples. For the University of Pavia data set, all the nine land-cover classes were used to validate the proposed method. We also selected randomly 200 samples as the training samples for each class. Tables I and II show the details of the training and test samples over Indian Pines and University of Pavia data set, respectively.

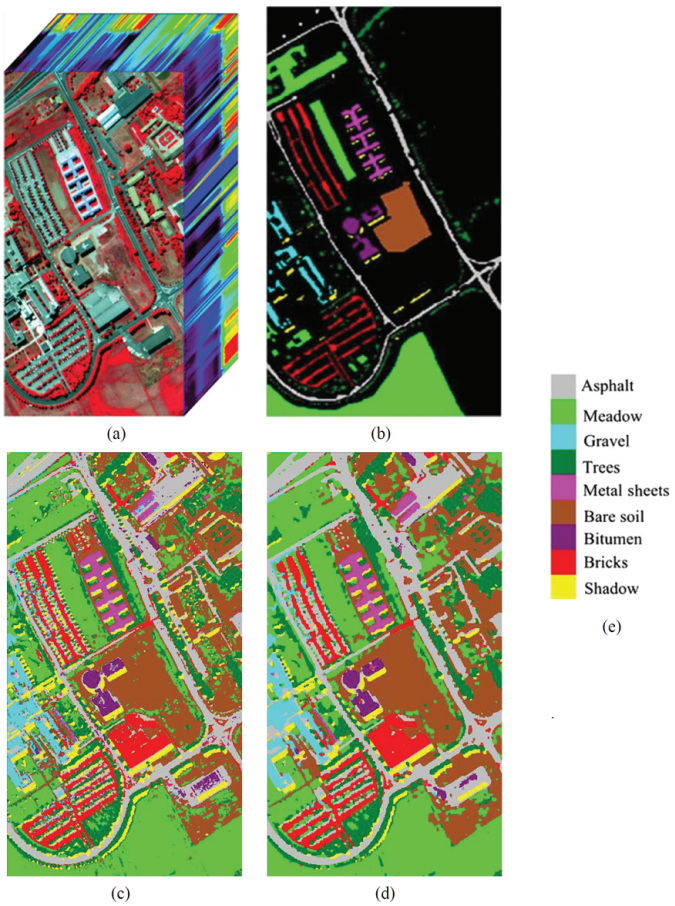

Figure 6. University of Pavia data set and its example classification results. (a) Original image produced by the mixture of three bands. (b) Ground truth with nine classes. (c) and (d) The classification results of DBN-CRF-U and DBN-CRF. (e) Map color.

TABLE I

NUMBER OF THE TRAINING AND TEST SAMPLES USED IN THE INDIAN PINES DATA SET

ID	CLASS NAME	TRAINING	TEST
1	Corn-notill	200	1234
2	Corn-mintill	200	634
3	Grass-pasture	200	297
4	Hay-windrowed	200	289
5	Soybean-notill	200	768
6	Soybean-mintill	200	2268
7	Soybean-clean	200	414
8	Woods	200	1094
Total		1600	6998

TABLE II

NUMBER OF THE TRAINING AND TEST SAMPLES USED IN THE UNIVERSITY OF PAVIA DATA SET.

ID	CLASS NAME	TRAINING	TEST
1	Asphalt	200	6431
2	Meadows	200	18449
3	Gravel	200	1899
4	Trees	200	2864
5	Sheets	200	1145
6	Bare soil	200	4829
7	Bitumen	200	1130
8	Bricks	200	3482
9	Shadows	200	747
Total		**1800**	**40976**

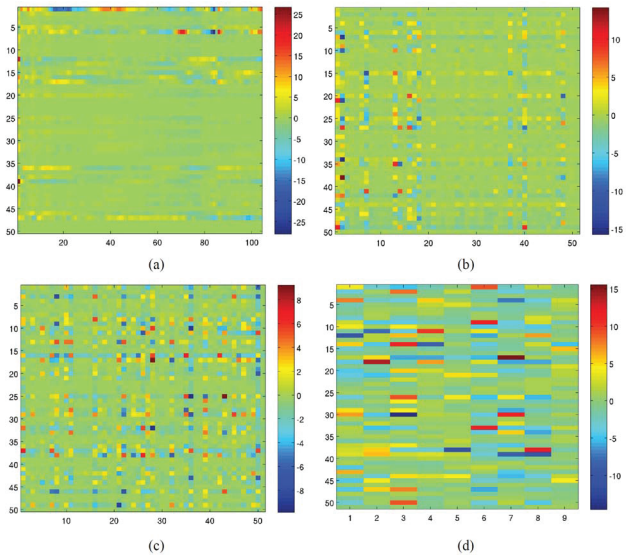

Figure 7. Example results of the learned weight parameters over the University of Pavia data set: (a) - (d) are the learned weight parameters from layer 1 to 4.

Equation (8) also shows that the training of pairwise DBN needs pairwise pixels to extract pairwise features. In theory, any two pixels from same land cover class (corresponding to the setting $\mathbf{v}_{mn} = 0$ if $m \neq n$) in the given image can be selected and combined as the pairwise pixels. Therefore, given the 200 selected training samples for each class, we can construct 40,000 pairwise training samples for each class, and total 320,000 and 360,000 pairwise training samples for Indian Pines and University of Pavia data set, respectively.In order to accelerate the training procedure, only 200 pairwise samples for each pair-class were randomly selected for training. The experimental results demonstrated that the selected training pairwise samples are enough to produce good results, and of course, increasing the number of the training sample could further improve the results. Before the training procedure, all spectral vectors were normalized to have the values between 0 and 1. For each data set, ten different training and test sets were created in our experiments

TABLE III
CONFUSION MATRIX OF THE PROPOSED DBN-CRF MODEL OVER INDIAN PINES DATA SET.

	C1	C2	C3	C4	C5	C6	C7	C8
C1	**88.98**	2.84	0.16	0	3.40	2.84	1.62	0.16
C2	1.58	**94.32**	0.16	0.32	1.26	0.79	1.58	0
C3	0	0.34	**96.63**	1.01	0	0	1.35	0.67
C4	0	0	1.38	**98.62**	0	0	0	0
C5	1.43	1.95	0	0	**93.36**	1.30	1.95	0
C6	2.78	2.69	1.15	0.22	3.17	**87.65**	2.03	0.31
C7	1.21	1.69	0.48	0	0.97	0.24	**95.41**	0
C8	0	0	0.55	0	0	0	0.64	**98.81**

4.3. Performance Evaluation

1) General Classification Performance: The performance of the proposed method is affected significantly by the structure of the DBN, which determines the quality of learned features from various aspects such as invariance and abstraction. Generally, if given sufficient training samples, more layers could make the DBN have more abilities to represent the input data. But for the limited training samples in the usual hyperspectral image classification and meanwhile in consideration of the computational complexity, the structures of the unary and pairwise DBNs are 200-50-50-50 and 400-50-50-50 for the Indian Pines data set, while 103-50-50-50 and 206-50-50-50 for University of Pavia data set. Details about the effects of the structures of DBNs on the performances can be found in [39]. The proposed DBN-CRFs were trained over the given training samples through the efficient training method proposed in Section 3. Then the learned models were used to classify the whole hyperspectral images, i.e., solving the optimization of (17) through LBP algorithm. The power parameter λ_1 in (17) was learned through cross validation as 0.9.

As an example, Fig. 7 shows the learned weight parameters $\theta_u = \left\{W_u^L, B_u^L, U\right\}$ over the University of Pavia data set. Although they are not so obvious as that in the filters trained over 2-D visual signals, some structures in the filters still can be observed. The learned first layer weights are localized continuous structure filters, and the weights in the second and third layers are shown as the local singular filters, which could correspond to the higher level abstract representation of the input signals. These results are consistent with much prior work, especially in 2-D signal representation. Figs. 5 and 6 show the classification results for visual evaluation. Figs. 5(c)-(d) and 6(c)-(d) give the classification results of DBN-CRFs with only unary potentials (setting parameters$\{\mathbf{v}_{mm}, m = 1, 2, ..., M\}$ as 0 in inference, named as DBN-CRF-U) and both unary and pairwise potentials. It can be noted that the DBN-CRF-U, which does not take into account any neighborhood interactions in both observed data and labels, always results in noisy classifications. However, the full DBN-CRF model obtained much better classification results, where the full DBN-CRF model protected the shapes and details of some objects, while simultaneously removed the isolated classification noise. This apparently demonstrates how combining the unary potentials, which focus on discrimination from deep representations, and pairwise potentials, which capture contextual information in both labels and deep representations, improves the classification results.

In order to carry out quantitative evaluation, we computed confusion matrices and average values from overall accuracies (OAs), average accuracies (AAs), and Kappa statistics (Kappa) of ten run of trainings and tests. Tables III and IV present the confusion matrix over Indian Pines and University of Pavia data sets, respectively. Inspection of the confusion matrices and classification accuracies for each class confirms that the most critical classes to separate in Indian Pines data set are corn-notill, corn-mintill, soybean-notill and soybean-mintill, while in University of Pavia data set, they are asphalt, bitumen and bricks. This situation was expected, as the spectral behaviors of such classes in each data set are quite similar. Table V presents the classification performances of

TABLE IV

CONFUSION MATRIX OF THE PROPOSED DBN-CRF MODEL OVER UNIVERSITY OF PAVIA DATA SET.

	C1	C2	C3	C4	C5	C6	C7	C8	C9
C1	**89.43**	0.25	1.71	0	0.19	0.36	4.90	3.09	0.02
C2	0	**95.52**	0.02	0.92	0	3.47	0	0.06	0
C3	0.53	0.42	**89.94**	0	0	0	0	8.95	0
C4	0	2.03	0	**97.17**	0	0.70	0	0	0
C5	0	0.09	0	0	**99.56**	0.17	0	0	0
C6	0.10	4.56	0	0.08	0.06	**94.80**	0	0.29	0
C7	3.72	0	0.27	0	0.09	0	**95.75**	0	0
C8	1.72	0.17	7.61	0	0.06	0.57	0.32	**89.43**	0
C9	0	0	0	0	0	0	0	0	**100**

TABLE V

CLASSIFICATION ACCURACIES OF DBN-CRF-U AND DBN-CRF OVER INDIAN PINES (TOP) AND UNIVERSITY OF PAVIA DATA SET (BOTTOM). SEVERAL ACCURACY MEASURES ARE INCLUDED: CLASS PERCENTAGE ACCURACY ([%]), OVERALL ACCURACY (OA[%]), AVERAGE ACCURACY(AA[%]), AND KAPPA STATISTIC (KAPPA)

METHOD	CLASS PERCENTAGE ACCURACY [%]									OA[%]	AA[%]	KAPPA
	C1	C2	C3	C4	C5	C6	C7	C8	C9			
DBN-CRF-U	84.12	91.80	96.97	97.23	88.54	81.44	93.48	98.63		88.34	91.53	0.8581
DBN-CRF	88.98	94.32	96.63	98.62	93.36	87.65	95.41	98.81		**92.15**	**94.22**	**0.9044**
DBN-CRF-U	87.40	92.14	85.78	95.08	99.56	92.01	92.65	87.28	100	91.24	92.43	0.8836
DBN-CRF	89.43	95.52	89.94	97.17	99.56	94.80	95.75	89.43	100	**94.02**	**94.62**	**0.9201**

DBN-CRF-U and full DBN-CRF. The OA and AA of the full DBN-CRF over Indian Pines data set are 92.15% and 94.22%, which are much better than 88.34% and 91.53% of the DBN-CRF-U. Over the University of Pavia data set, the full DBN-CRF obtained 94.02% OA and 94.62% AA, which are also higher than 91.24% and 92.43 obtained by the DBN-CRF-U. In addition, Table V shows that the full DBN-CRF also obtained better Kappa measures than that of the DBN-CRF-U.

2) Effects of Training Set Size on Classification Accuracies: The experiments are conducted to further verify that the proposed method is suitable for classifying data sets with limited training samples. Over the Indian Pines data set, eight different situations were analyzed, i.e., 50, 100, 150, 200, 250, 300, 350 and 400 samples of each class were randomly selected to train the models and the remaining sample sets were used to evaluate the classification accuracies, while over the University of Pavia , we analyzed seven different situations where training data sets have 100, 200, 300, 400, 500, 600 and 700 samples for each class. Fig. 8 shows the results of different models. It is obvious that our proposed full DBN-CRF consistently provides higher accuracy than the DBN-CRF-U. Although the classification accuracy of DBN-CRF suffers from the decrease of the training samples and the increase of the test samples, the proposed DBN-CRF generally demonstrates relatively stable classification performance.

3) Computational Performance: At the end of this subsection, we evaluate the computational performances of the proposed method. Within the piecewise training framework, the DBN-CRF model parameters $\theta_u = \left\{W_u^L, B_u^L, U\right\}$ and $\theta_p = \left\{W_p^L, B_p^L, V\right\}$ of the unary and pairwise potentials were separately learned by the efficient training

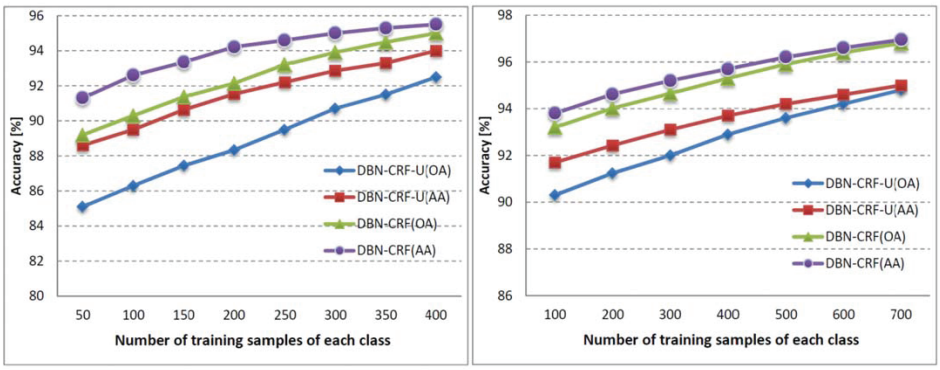

Figure 8. Classification accuracies versus numbers of training samples (each class) for the Indian Pines data set (a) and University of Pavia data set (b).

method proposed in Section 3. Corresponding to the unary DBNs with structure as 200-50-50-50 and 103-50-50-50 for Indian Pines and University of Pavia respectively, there are total 15,000 ($200 \times 50 + 50 \times 50 + 50 \times 50$) and 10,150 ($103 \times 50 + 50 \times 50 + 50 \times 50$) weight parameters in the W_u^L and 150 and 150 bias parameters in the B_u^L, while for the pairwise DBNs with structure as 400-50-50-50 and 206-50-50-50, there are total 25,000 ($400 \times 50 + 50 \times 50 + 50 \times 50$) and 15,300 ($206 \times 50 + 50 \times 50 + 50 \times 50$) elements in the W_p^L and 150 and 150 elements in B_p^L. To fine-tune the DBNs, there are 400(50×8) and 450(50×9) CRF parameters in U and V for the Indian Pines and University of Pavia data sets, respectively. The whole training procedure of the DBN-CRFs were implemented with 1000 epochs of pre-training and 10000 epochs of fine-tuning for both the unary and pairwise DBNs.

In order to demonstrate the usability of the proposed method, the very common machine with a 2.6-GHz Intel (R) Core (IM) i5 and 8GB memory was used to run our method. Over the Indian Pines data set, the pre-training and fine-tuning of the unary DBN implemented by our unoptimized Matlab code took about 64 and 675 seconds (s) respectively, while 171 and 1250 s over the pairwise DBN. Then the training of the unary and pairwise DBNs took about 739 and 1421 s, respectively. The time cost for the whole training procedure (sequentially performed the training of unary and pairwise potentials) is then 2160 s. Over the University of Pavia data set, the pre-training and fine-tuning of the unary DBN took about 59 and 563 s respectively, while 155 and 1126 s for the pairwise DBN. Then the training of the unary and pairwise DBNs took about 622 and 1281 s, respectively. The time cost for the whole training procedure is then 1903 s. Using the estimated parameters, the learned DBNs extracted the deep features for the unary and pairwise potentials, and then the LBP algorithm solved the maximization of (17) to infer the labels. The whole procedure, including the feature extraction and the optimization, cost about 32 and 287 s over the Indian Pines and University of Pavia data sets, respectively.

We must admit that the training process is relatively time-consuming to achieve the good performances. However, the proposed DBN-CRF shares the same advantages of deep learning algorithms, such as the relatively fast inference and the good representation of hyperspectral image and thus the good classification performance. Moreover, our method could be improved greatly on efficiency. Firstly, the proposed training method naturally allows the parallel way to train the unary and pairwise DBNs. Secondly, the codes of the pre-training and fine-tuning of DBNs can be also modified to run on the GPUs, which can significantly accelerate the training procedure.

4.4. Comparison to Other Methods

To thoroughly evaluate the performance of the proposed method, we ran several sets of experiments to compare it with the most recent results in hyperspectral image classification. Firstly, we compared our method with the

successful SVM-based method to demonstrate the performance difference between our deep method and the state-of-the-art 'shallow' methods. Secondly, we compared our method with the newly developed deep CNN model without using contextual (spatial) information. The comparison was used to demonstrate the ability of our method to use the contextual (spatial) information and its importance in hyperspectral image classification. Finally, the proposed method was compared with the recent DBN with the spectral-spatial information in the observation. The comparison is designed to show the advantages of our method in modeling and using the contextual information in both the observations and labels, and thus the merits to improve the classification performance. We compute the McNemar's test, which is based upon the standardized normal test statistic[53] , to assess the statistical significance of differences between the accuracies achieved by two different methods. The statistic can be computed as

$$F_{ij} = \frac{f_{ij} - f_{ji}}{\sqrt{f_{ij} + f_{ji}}} \tag{18}$$

where F_{ij} measures the pairwise statistical significance of the difference between the accuracies of the $i-\text{th}$ and $j-\text{th}$ methods and the f_{ij} is the number of samples classified correctly by $i-\text{th}$ method but wrongly by $j-\text{th}$ method. At the 95% level of confidence, the difference of accuracies between the different methods is statistically significant if $|F_{ij}| > 1.96$. Table VI and VII show the classification results of the different methods over the Indian Pines and University of Pavia data set, respectively.

1) Comparison to SVM. SVM-based method can be deemed as the benchmark 'shallow' hyperspectral image classification method. SVM-based method and our CRF were trained and tested on same training and test data sets with the sizes presented in Table I and II. Over the Indian Pines data set, the SVM-Poly obtained the classification result with OA, AA and Kappa as 87.65%, 91.01% and 0.8499, while the proposed method obtained the better result with OA, AA and Kappa as 92.15%, 94.22% and 0.9044. Over the University of Pavia data set, the proposed method also produced much better results than that did by the SVM-Poly. In addition, the computed $|F_{ij}|$between SVM-Poly and our method over the Indian Pines and DC Mall are 15.73 and 19.01, which mean the better results of our method over the SVM-Poly are statistically significant. Since the SVM-Poly is a typical 'shallow' classifier, thus the comparison between the results demonstrated that the DBN representations from the deep learning and the contextual information captured by the CRF model can benefit the hyperspectral image classification.

2) Comparison to shallow CRF. The CRF for hyperspectral image classification has been proposed in our previous work [31]. The multimodal logistic regression (MLR) was used to define the unary potentials of the CRF model, while the pairwise potentials were defined as the Ising model. Combined with the proposed efficient training method, the CRF model is fit for the real-world hyperspectral image classification. Moreover, the CRF has the ability to capture and use the important contextual information in both the observations and labels. Therefore, the CRF obtained the promising results over the real-world hyperspectral images. However, the MLR used to define the unary potentials is a discriminative classifier and shallow, thus the CRF is actually a shallow discriminative model. Moreover, the discriminative characteristic makes the shallow CRF focus mainly on the model's ability to discriminate different land cover classes, and thus lack the typical abilities of the generative models to use the good description model of the observation to improve classification performance. In addition, although the shallow CRF and the proposed DBN-CRF have the similar formulations of the potentials, the features used in the proposed DBN-CRF derive from the deep representation models. Therefore, the shallow CRF and DBN-CRF have the similar ability to capture the contextual information, but the DBN-CRF has the extra merits from the deep representation. The merits make the proposed DBN-CRF obtain much better classification results (see the experimental results in Table VI and VII). Moreover, the computed $|F_{ij}|$ between the shallow CRF and our method over the Indian Pines and University of Pavia data sets are 13.28 and 16.36, which mean that the difference between our method and the shallow CRF are statistically significant. This experimentally validates that the deep representation combined with the contextual information can significantly benefit the classification performance.

3) Comparison to spectral CNN. CNNs are biologically inspired and multilayer classes of deep learning models. They have demonstrated excellent performance on various visual tasks, including the classification of common two-dimensional images. Work [40] further introduced the CNN into the hyperspectral images classification and produced very promising results. Therefore, we further compare our method to the CNN. The architecture of the proposed CNN contains five layers, including the input layer, the convolutional layer, the max pooling layer, the

TABLE VI

CLASSIFICATION ACCURACIES OF DIFFERENT METHODS OVER INDIAN PINES. SEVERAL ACCURACY MEASURES ARE INCLUDED: CLASS PERCENTAGE ACCURACY ([%]), OVERALL ACCURACY (OA[%]), AVERAGE ACCURACY(AA[%]), AND KAPPA STATISTIC (KAPPA). $|F_{ij}|$ IS THE COMPUTED VALUES OF THE MCNEMAR'S TEST BETWEEN THE GIVEN METHOD AND THE PROPOSED DBN-CRF MODEL.

METHOD	CLASS PERCENTAGE ACCURACY [%]								OA[%]	AA[%]	KAPPA	$\|F_{ij}\|$
	C1	C2	C3	C4	C5	C6	C7	C8				
SVM-POLY	83.55	90.85	95.96	98.62	87.37	80.42	92.75	98.54	87.65	91.01	0.8499	15.73
CRF	87.60	87.70	94.28	98.96	88.41	81.97	93.96	98.63	88.73	91.44	0.8629	13.28
CNN	-	-	-	-	-	-	-	-	90.16	-	-	-
DBN-LR	84.20	91.48	96.30	97.92	88.67	81.31	93.24	98.45	88.25	91.45	0.8572	14.26
DBN-LR-S-S	88.01	93.06	95.96	98.62	92.45	85.80	94.44	98.72	91.07	93.38	0.8912	6.39
DBN-CRF	88.98	94.32	96.63	98.62	93.36	87.65	95.41	98.81	**92.15**	**94.22**	**0.9044**	-
DBN-CRF-S-S	89.79	94.79	96.97	98.96	93.75	88.10	96.38	99.18	**92.67**	**94.74**	**0.9107**	2.85

TABLE VII

CLASSIFICATION ACCURACIES OF DIFFERENT METHODS OVER UNIVERSITY OF PAVIA DATA SET. SEVERAL ACCURACY MEASURES ARE INCLUDED: CLASS PERCENTAGE ACCURACY ([%]), OVERALL ACCURACY (OA[%]), AVERAGE ACCURACY(AA[%]), AND KAPPA STATISTIC (KAPPA). $|F_{ij}|$ IS THE COMPUTED VALUES OF THE MCNEMAR'S TEST BETWEEN THE GIVEN METHOD AND PROPOSED DBN-CRF MODEL.

METHOD	CLASS PERCENTAGE ACCURACY [%]									OA	AA	KAPPA	$\|F_{ij}\|$
	C1	C2	C3	C4	C5	C6	C7	C8	C9				
SVM-POLY	85.68	91.73	85.62	95.39	99.39	92.01	94.34	85.93	100	90.73	92.23	0.8771	19.01
CRF	89.97	91.12	83.94	96.40	99.21	93.17	92.30	87.94	100	91.37	92.67	0.8856	16.36
CNN	87.34	94.63	86.47	96.29	99.65	93.23	93.19	86.42	100	92.56	93.02	0.9006	-
DBN-LR	87.37	92.10	85.57	95.11	99.74	91.94	92.21	87.02	100	91.18	92.34	0.8828	17.99
DBN-LR-S-S	87.47	94.80	87.31	96.54	99.65	93.46	94.16	87.11	100	92.83	93.39	0.9043	11.67
DBN-CRF	89.43	95.52	89.94	97.17	99.56	94.80	95.75	89.43	100	**94.02**	**94.62**	**0.9201**	-
DBN-CRF-S-S	89.72	95.79	91.00	97.73	99.48	95.22	96.64	89.72	100	**94.37**	**95.03**	**0.9248**	3.16

full connection layer, and the output layer. Although the proposed CNN can capture some contextual information through the convolutional and pooling layer, the proposed CNNs have been performed over only the spectral domain, and thus neglected the important spatial information of the hyperspectral images. We name this method as spectral CNN in this paper.

For the fair comparison, our method was performed under the experimental setup same as that in work [40]. Moreover, we used directly the results from work [40]. However, only partial results corresponding to the evaluations in this work have been presented in work [40]. For the Indian Pines data set, only the OAs have been provided, while for the University of Pavia data set, although the work [40] provided only the OAs, we calculated the AAs and Kappa values using the available results in work [40]. The results show that the proposed DBN-CRF model produced better results than that of CNNs. This means that besides the deep representation of the spectral observations, the spatial information in the hyperspectral images also play a very important role to improve the hyperspectral image classification.

4) Comparison to spectral-spatial DBN: Finally, the proposed DBN-CRF model is compared with another deep learning method, i.e., the DBN model with the last layer as a LR classifier (DBN-LR). Work [39] implemented two DBN-LR classifiers. The first one uses only the spectral signatures as the input, then trains the DBN to extract the deep features, and finally inputs the deep features into the LR to get the final labels. The same training and test data sets with the sizes presented in Table I and II were used to train the DBN-LR, and the experimental setting in training and test DBN-LR is same as in [39]. It can be noted from the results in Table VI and VII that DBN-LR produced better results than the 'shallow' SVM did. This demonstrates the merits of deep representation that the deep learning methods share. Further checking the results of DBN-LR and another analogue deep model, i.e., spectral CNN [40], shows that the spectral CNN obtained a little better results. This could derive from the fact that the spectral CNN can model and use the contextual information in spectral bands through the convolutional and pooling layers, while the DBN-LR cannot sufficiently use the contextual information since the variables in one RBM layer (stacked to formulate the DBN-LR) are assumed to be independent. It can be easily noted from Table VI and VII that our method produced better results than both the DBN-LR and spectral CNN.

To further improve the performance of DBN-LR, work [39] proposed a novel deep architecture, which combines the spectral-spatial feature and classification together. The new method (named DBN-LR-S-S) performs the principal component analysis (PCA) transformation over the hyperspectral image, then the first several principal components of the neighboring sites are concatenated to form the contextual feature of one site, and finally the extracted contextual features are input into the DBN and LR to get the final labels. The results in Table VI and VII show that the DBN-LR-S-S not only outperformed the DBN-LR, but also produced better results than the CNN did. This means the spatial information in the observations did benefit the hyperspectral image classification. Our method further captured and used the spatial information in both the observations and labels, and thus obtained better results than that of the DBN-LR-S-S. Moreover, the computed $|F_{ij}|$ between the DBN-LR-S-S and our method over the Indian Pines and University of Pavia data sets are 6.39 and 11.37. This means that the superiority of our method over the DBN-LR-S-S is statistically significant. Especially, if the same contextual features used in DBN-LR-S-S were used in our method (named DBN-CRF-S-S), the classification performance of our method can be further improved, and the improvement is statistically significant (see the computed $|F_{ij}|$ in Table VI and VII). The comparisons demonstrate that the deep representations and contextual information in both the observations and labels can play an important role in the hyperspectral image classification. Our method can effectively fuse the important information through the designed DBN-CRF structure and the proposed end-to-end training method.

5. Conclusion and Discussion

In this paper, we have proposed a novel DBN-CRF model for hyperspectral image classification. The proposed model takes advantage of the strength of DBNs in deep learning representation and CRFs in contextual (spatial) modeling in both the observations and labels. Therefore, the deep representations and contextual information in the hyperspectral image have been combined to improve the hyperspectral image classification. In addition, the proposed end-to-end training method of the DBN-CRF jointly trains the DBN and CRF in a very efficient way through the usual back-propagation method. The experimental results over the real-world hyperspectral data validated the efficiency and effectiveness of our method in the classification task.

Our current method uses the DBNs to extract the deep features to define the potentials of CRFs. Other deep learning methods, such as the deep CNNs, also have great potentiality for hyperspectral image classification. In

theory, these deep learning methods can be directly introduced into our method. Especially, if the last layer of the deep models is the classifier similar with the soft-max, the training of the new models can be also implemented as the trainings of some simple models. Another future work is to use the more diverse contextual information in the CRF model through define different potentials. The current pairwise potentials model only the contextual information between the sites with same class label. The contextual information lies in the sites of different cover class can be modeled through setting the non-zero pairwise CRF parameters. Another way to increase the diversity of contextual information is introducing the high-order potentials, which can model the high-order statistics and capture the long-range contextual information.

Acknowledgements

This research was conducted with support of the Natural Science Foundation of China under Grant 61671456 and 61271439, A Foundation for the Author of National Excellent Doctoral Dissertation of P. R. China (FANEDD) under Grant 201243, Program for New Century Excellent Talents in University under Grant NECT-13-0164.

REFERENCES

[1] P. W. Yuen and M. Richardson,"An introduction to hyperspectral imaging and its application for security, surveillance and target acquisition," Imaging Science Journal, vol. 58, no. 5, pp. 241-253, 2010..

[2] T. Chen, P. Yuen, M. Richardson, G. Liu, and Z. She, "Detection of psychological stress using a hyperspectral imaging technique," IEEE Transactions on Affective Computing, vol. 5, no. 4, pp. 391-405, 2014.

[3] X. Tong, H. Xie, and Q. Weng, "Urban land cover classification with airborne hyperspectral data: what features to use?" IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, vol. 7, no. 10, pp. 3998-4009, 2014.

[4] C. M. Gevaert, J. Suomalainen, J. Tang, and L. Kooistra, "Generation of spectral-temporal response surfaces by combining multispectral satellite and hyperspectral UAV imagery for precision agriculture applications," IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, vol. 8, no. 6, pp. 3140-3146, 2015.

[5] J. M. Bioucas-Dias, A. Plaza, G. Camps-Valls, P. Scheunders, N. M. Nasrabadi, and J. Chanussot, "Hyperspectral remote sensing data analysis and future challenges," IEEE Geoscience and Remote Sensing Magazine, vol. 1, no. 2, pp. 6 - 36, 2013.

[6] P. Zhong, P. Zhang, and R. Wang, "Dynamic learning of sparse multinomial logistic regression for feature selection and classification of hyperspectral data," IEEE Geoscience and Remote Sensing Letter, vol. 5, no. 2, pp. 280-284, 2008.

[7] P. Zhong and R. Wang, "Jointly learning the hybrid CRF and MLR model for simultaneous denoising and classification of hyperspectral imagery," IEEE Transactions on Neural Networks and Learning Systems, vol. 25, no. 7, pp. 1319-1334, 2014.

[8] M. Khodadadzadeh, J. Li, A. Plaza, and J. M. Bioucas-Dias, "A subspace-based multinomial logistic regression for hyperspectral image classification," IEEE Geoscience and Remote Sensing Letter, vol. 11, no. 12, pp. 2105-2109, 2014.

[9] F. Ratle, G. Camps-Valls, and J. Weston, "Semisupervised neural networks for efficient hyperspectral image classification," IEEE Transactions on Geoscience and Remote Sensing, vol. 48, no. 5, pp. 2271-2282, 2010.

[10] Y. Zhong, W. Liu, J. Zhao, and L. Zhang, "Change detection based on pulse-coupled neural networks and the NMI feature for high spatial resolution remote sensing imagery, " IEEE Geoscience and Remote Sensing Letters, vol. 12, no. 3, pp. 537-541, 2015.

[11] F. Melgani and L. Bruzzone, "Classification of hyperspectral remote sensing images with support vector machines," IEEE Transactions on Geoscience and Remote Sensing, vol. 42, no. 8, pp. 1778-1790, 2004.

[12] J. Peng, Y. Zhou, and C. L. P. Chen, "Region-kernel-based support vector machines for hyperspectral image classification," IEEE Transactions on Geoscience and Remote Sensing, vol. 53, no. 9, pp. 4810-4824, 2015.

[13] G. Camps-Valls, T. V. B. Marsheva, and D. Zhou, "Semi-supervised graph-based hyperspectral image classification," IEEE Transactions on Geoscience and Remote Sensing, 45, pp. 3044-3054. 2007.

[14] Y. Gao, R. Ji, P. Cui, Q. Dai, and G. Hua, "Hyperspectral image classification through bilayer graph-based learning," IEEE Transactions on Image Processing, vol. 23, no. 7, pp. 2769-2778, 2014.

[15] S. Kawaguchi and R. Nishii, "Hyperspectral image classification by bootstrap AdaBoost with random decision Stumps," IEEE Transactions on Geoscience and Remote Sensing, vol. 45, no. 11, pp. 3845-3851, 2007.

[16] S. Sun, P. Zhong, H, Xiao, and R. Wang, "Active learning with Gaussian process classifier for hyperspectral image classification," IEEE Transactions on Geoscience and Remote Sensing, vol. 53, no. 4, pp. 1746-1760, 2014.

[17] J. Ham, Y. Chen, M. M. Crawford, and J. Ghosh, "Investigation of the random forest framework for classification of hyperspectral data," IEEE Transactions on Geoscience and Remote Sensing, vol. 43, no. 3, pp. 492-501, 2005.

[18] M. Fauvel, Y. Tarabalka, J. A. Benediktsson, J. Chanussot, and J. C. Tilton, "Advances in spectral-spatial classification of hyperspectral images," Proceedings of the IEEE, vol. 101, no. 3, pp. 652-675, 2013.

[19] X. Jia, B. Kuo, and M. M. Crawford, "Feature mining for hyperspectral image classification," Proceedings of the IEEE, vol. 101, no. 3, pp. 676-697, 2013.

[20] Y. Zhou and Y. Wei, "Learning hierarchical spectral-spatial features for hyperspectral image classification," IEEE Transactions on Cybernetics, vol. PP, no. 99, pp. 1-12, 2015.

[21] P. Ghamisi, J. A. Benediktsson, G. Cavallaro, and A. Plaza, "Automatic framework for spectral-spatial classification based on supervised feature extraction and morphological attribute profiles," IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, vol. 7, no. 6, pp. 2147-2160, 2014.

[22] N. Falco, J. A. Benediktsson, and L. Bruzzone, "Spectral and spatial classification of hyperspectral images based on ICA and reduced morphological attribute profiles," IEEE Transactions on Geoscience and Remote Sensing, vol. 53, no. 11, pp. 6223-6240, 2015.

[23] Z. Zhong, B. Fan, J. Duan, L. Wang, K. Ding, S. Xiang, and C. Pan, "Discriminant tensor spectral-spatial feature extraction for hyperspectral image classification," IEEE Geoscience and Remote Sensing Letter, vol. 12, no. 5, pp. 1028-1032, 2015.

[24] Y. Qian, M. Ye, and J. Zhou, "Hyperspectral image classification based on structured sparse logistic regression and three-dimensional wavelet texture features," IEEE Transactions on Geoscience and Remote Sensing, vol. 51, no. 4, pp. 2276-2291, 2013.

[25] W. Li, C. Chen, H. Su, and Q. Du, "Local binary patterns and extreme learning machine for hyperspectral imagery classification," IEEE Transactions on Geoscience and Remote Sensing, vol. 53, no. 7, pp. 3681-3693, 2015.

[26] F. Tsai and J. Lai, "Feature extraction of hyperspectral image cubes using three-dimensional gray-level cooccurrence," IEEE Transactions on Geoscience and Remote Sensing, vol. 51, no. 6, pp. 3504-3513, 2013.

[27] Y. Tarabalka, J. Chanussot, and J. A. Benediktsson, "Segmentation and classification of hyperspectral images using minimum spanning forest grown from automatically selected markers," IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics, vol. 40, no. 5, pp. 1267-1279, 2010.

[28] J. Bai, S. Xiang, and C. Pan, "A graph-based classification method for hyperspectral images," IEEE Transactions on Geoscience and Remote Sensing, vol. 51, no. 2, pp. 803-817, 2013.

[29] S. Sun, P. Zhong, H. Xiao, and R. Wang, "An MRF model-based active learning framework for the spectral-spatial classification of hyperspectral imagery," IEEE Journal of Selected Topics in Signal Processing, vol. 9, no. 6, pp. 1074-1088, 2015.

[30] Y. Yuan, J. Lin, and Q.Wang, "Hyperspectral image classification via multitask joint sparse representation and stepwise MRF optimization," IEEE Transactions on Cybernetics, vol. PP, no. 99, pp. 1-12, 2015.

[31] P. Zhong and R. Wang, "Learning conditional random fields for classification of hyperspectral images," IEEE Transactions on Image Processing, vol. 19, no.7, pp. 1890-1907, 2010.

[32] P. Zhong and R. Wang, "Learning sparse CRFs for feature selection and classification of hyperspectral imagery," IEEE Transactions on Geoscience and Remote Sensing, vol. 46, no. 12, pp. 4186-4197, 2008.

[33] P. Zhong and R. Wang, "Modeling and classifying hyperspectral imagery by CRFs with sparse higher order potentials," IEEE Transactions on Geoscience and Remote Sensing, vol. 49, no. 2, pp. 688-705, 2011.

[34] M. Alioscha-Perez and H. Sahli, "Efficient learning of spatial patterns with multi-scale conditional random fields for region-based classification," Remote Sensing, vol. 6, no. 8, pp. 6727-6764, 2014.

[35] M. Y. Zhong, J. Zhao, and L. Zhang, "A hybrid object-oriented conditional random field classification framework for high spatial resolution remote sensing imagery, " IEEE Transactions on Geoscience and Remote Sensing, vol. 52, no. 11, pp. 7023-7037, 2014.

[36] F. Li, L. Xu, P. Siva, A. Wong, and D. A. Clausi, "Hyperspectral image classification with limited labeled training samples using enhanced ensemble learning and conditional random fields," IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, vol. 8, no. 6, pp. 2427-2438, 2015.

[37] Y. Chen, Z. Lin, X. Zhao, G. Wang, and Y. Gu, "Deep learning-based classification of hyperspectral data," IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, vol. 7, no. 6, pp. 2094-2107, 2014.

[38] C. Tao, H. Pan, Y. Li, and Z. Zou, "Unsupervised spectral-spatial feature learning with stacked sparse autoencoder for hyperspectral imagery classification," IEEE Geoscience and Remote Sensing Letter, vol. 12, no. 12, pp. 2438-2442, 2015.

[39] Y. Chen, X. Zhao, and X. Jia, "Spectral-spatial classification of hyperspectral data based on deep belief network," IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, vol. 8, no. 6, pp. 2381-2392, 2015.

[40] W. Hu, Y. Huang, L. Wei, F. Zhang, and H. Li, "Deep convolutional neural networks for hyperspectral image classification," Journal of Sensors, vol. 2015, Article ID: 258619, 12 pages, 2015.

[41] A. Romero, C. Gatta, and G. Camps-Valls, "Unsupervised deep feature extraction for remote sensing image classification," IEEE Transactions on Geoscience and Remote Sensing, vol. PP, no. 99, pp. 1-14, 2015.

[42] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," Nature, vol. 521 , pp. 436-444, 2015.

[43] A. Krizhevsky, I. Sutskever and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," In Proc. Advances in Neural Information Processing Systems (NIPS), 25, pp. 1090-1098, 2012.

[44] R. Girshick, J. Donahue, T. Darrell and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," In Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2014.

[45] S. Zheng, S. Jayasumana, B. Romera-Paredes, V. Vineet, Z. Su, D. Du, C. Huang, and P. Torr, "Conditional random fields as recurrent neural networks," In Proc. International Conference on Computer Vision (ICCV), 2015.

[46] L. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "Semantic image segmentation with deep convolutional nets and fully connected CRFs," In Proc. International Conference on Learning Representations (ICLR), 2015.

[47] G. Lin, C. Shen, I. Reid, and A. Hengel, "Efficient piecewise training of deep structured models for semantic segmentation," In arXiv:1504.01013, 2015.

[48] G. E. Hinton, S. Osindero, and Y. Teh, "A fast learning algorithm for deep belief nets," Neural Computation, vol. 18, pp. 1527-1554, 2006.

[49] C. M. Bishop, "Neural networks for pattern recognition", Oxford University Press, 1996.

[50] C. Sutton and A. McCallum, "Piecewise training of undirected models," in Proc. Conference on Uncertainty in Artificial Intelligence (UAI), 2005.

[51] J. Shotton, J. Winn, C. Rother, and A. Criminisi, "TextonBoost: Joint appearance, shape and context modeling for multi-class object recognition and segmentation," in Proc. European Conference on Computer Vision (ECCV), 2006.

[52] B. Frey and D. J. C. Mackay, "A revolution: Belief propagation in graphs with cycles," in Proc. Advances in Neural Information Processing Systems (NIPS), 1997.

[53] Y. Bazi and F. Melgani, "Toward an optimal SVM classification system for hyperspectral remote sensing images," IEEE Transactions on Geoscience and Remote Sensing, vol. 44, no. 11, pp. 3374-3385, 2006.