This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

Classification of Hyperspectral Remote Sensing Image Using Hierarchical Local-Receptive-Field-Based Extreme Learning Machine

Qi Lv, Xin Niu, Yong Dou, *Member, IEEE*, Jiaqing Xu, and Yuanwu Lei

***Abstract*—This letter proposes a novel classification approach for a hyperspectral image (HSI) using a hierarchical local-receptive-field (LRF)-based extreme learning machine (ELM). As a fast and accurate pattern classification algorithm, ELM has been applied in numerous fields, including the HSI classification. The LRF concept originates from research in neuroscience. Considering the local correlations of spectral features, it is promising to improve the performance of HSI classification by introducing the LRFs. Recent research on deep learning has shown that hierarchical architectures with more layers can potentially extract abstract representation and invariant features for better classification performance. Therefore, we further extend the LRF-based ELM method to a hierarchical model for HSI classification. Experimental results on two widely used real hyperspectral data sets confirm the effectiveness of the proposed HSI classification approach.**

***Index Terms*—Deep learning, extreme learning machine (ELM), hyperspectral image (HSI) classification, local receptive field (LRF).**

I. Introduction

HYPERSPECTRAL remote sensing has become one of the frontier techniques in the field of remote sensing due to recent advances of hyperspectral imaging technology. For a hyperspectral image (HSI), hundreds of contiguous narrow spectral bands can be recorded as a data cube (with both spatial and spectral information), which covers a large spectral wavelength range, spanning from the visible to the infrared spectrum. One of the most important HSI processing tasks is classification, by which each pixel is classified into a certain land-cover class. HSI classification has played a heavy role in several applications, including material recognition, environment monitoring, precision agriculture, urban planning, and reconnaissance [1]. However, the large amount of bands and relatively small training sample size of HSI data bring challenges to conventional remote sensing classification methods.

To cope with these challenges and perform HSI classification effectively, several machine learning approaches have been studied, such as support vector machine (SVM), neural networks, manifold learning, active learning, etc. In recent years, other techniques like sparse representation classifier, Gaussian mixture model, and morphological profiles have also been investigated [1]. However, finding the optimal parameters for parametric supervised HSI classification methods is usually difficult and time consuming.

Recently, the extreme learning machine (ELM) was proposed for single-hidden layer feedforward neural networks (SLFNs), and it has been considered as a promising learning algorithm for pattern classification [2]. Numerous works have shown the capabilities of ELM in fast and accurate pattern classification. More recent advances of ELM can be found in [3]. There are already several works applying ELM to the classification of a remote sensing image. In [4], ELM was considered for the land-cover classification of hyperspectral data and provided comparable performance with that of the back-propagation (BP) neural network. ELM was also used for HSI classification in combination with differential evolution [5], ensemble learning [6], and kernel methods [7]. However, the aforementioned ELM-based works failed to sufficiently exploit the relevance among the spectral features of HSI data.

A new biologically inspired ELM framework has been recently proposed in [8], which is implemented by introducing the local receptive field (LRF) concept in neuroscience. The major idea of LRF-based ELM (L-ELM) is the locally dense connection between the input layer and hidden layer. By considering the local connections of spectral features, it is hopeful to further improve the performance of HSI classification. Therefore, a novel HSI classification scheme is adopted in this letter by employing the L-ELM model.

In HSI analysis, it is common to perform feature representation or feature extraction before classification [9], [10]. The goal is that the obtained features are expected to have the ability to discriminate pixels from different classes while being invariant to intraclass variability. During recent years, the deep learning technique has been actively studied and shown to give state-of-the-art results in many fields, including image classification, object detection, speech recognition, natural language processing, and so on [11]. The reason for such success lies in that hierarchical deep architectures can potentially extract more abstract representation and invariant features of data for better classification performance [12]. Several recent studies have adopted deep learning models for HSI classification [13]–[15]. For the purpose of extracting effective features for hyperspectral data, it is a promising attempt to employ the hierarchical architecture to benefit the HSI classification. Thus, in this letter, we further extend the L-ELM model to the hierarchical

Manuscript received July 11, 2015; revised November 8, 2015 and December 16, 2015; accepted January 9, 2016. This work was supported by the National Natural Science Foundation of China under Grants 61125201, 61303070, U1435219, 61402507, and 61402499.

The authors are with the School of Computer, National University of Defense Technology, Changsha 410073, China (e-mail: lvqi@nudt.edu.cn).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/LGRS.2016.2517178

1545-598X © 2016 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission.
See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

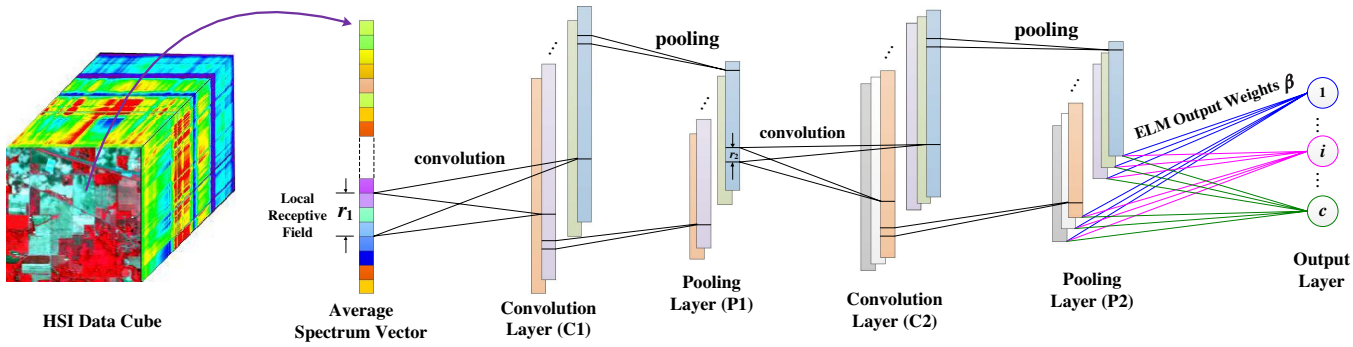

Fig. 1. Schematic diagram of HL-ELM. The framework consists of input layer, hierarchical convolution and pooling layer, and output layer. The connections between the last pooling layer and the output layer are the ELM output weights.

L-ELM (HL-ELM) model to extract spectral–spatial features of HSI data. Experiments on two real HSI data sets demonstrate that our proposed approach achieves both high classification performance and fast training speed.

II. Brief of ELM

ELM [2], [3] is an SLFN that uses randomly assigned input weights. Unlike the BP algorithm, it does not require the adjustment of input weights. In general, ELM is a feedforward neural network with a simple three-layer structure comprising an input layer, a hidden layer, and an output layer.

Suppose that $\mathbf{X}=\{\mathbf{x}_1,\mathbf{x}_2,\ldots,\mathbf{x}_N|\mathbf{x}_i\in\mathbb{R}^D, i=1,2,\ldots,N\}$ is the training set that contains N training samples and $\mathbf{T}=\{\mathbf{t}_1,\mathbf{t}_2,\ldots,\mathbf{t}_N|\mathbf{t}_i\in\mathbb{R}^c, i=1,2,\ldots,N\}$ is the training data target matrix, where $\mathbf{t}_i$ is the vectorized label and c is the number of classes. Then, the ELM model with M hidden neurons and an activation function $g(x)$ can be expressed as

$$\sum_{j=1}^{M}\beta_j g(\mathbf{w}_j\cdot\mathbf{x}_i+b_j)=\mathbf{t}_i,\quad i=1,2,\ldots,N \tag{1}$$

where $\mathbf{w}_j$ and β_j represent the weight vector from the jth hidden neuron to the input neurons and the output neurons, respectively, and b_j is the bias of the jth hidden neuron. The aforementioned N equations can be compactly rewritten as

$$\mathbf{H}\beta=\mathbf{T} \tag{2}$$

where $\beta\in\mathbb{R}^{M\times c}$ is the output weight matrix and $\mathbf{T}\in\mathbb{R}^{N\times c}$ is the target matrix. $\mathbf{H}\in\mathbb{R}^{N\times M}$ is the hidden layer output matrix, and $h_{ij}=g(\mathbf{w}_j\cdot\mathbf{x}_i+b_j)$.

By solving the aforementioned linear equation, the optimal output weights can be obtained as

$$\beta=\mathbf{H}^{\dagger}\mathbf{T} \tag{3}$$

where $\mathbf{H}^{\dagger}$ is the Moore–Penrose generalized inverse of the hidden layer output matrix $\mathbf{H}$.

III. Proposed Method

A. HL-ELM

Recently, a novel bioinspired ELM model named L-ELM has been proposed by Huang *et al.* [8]. In this model, the connections between the input and hidden nodes are sparse and bounded by corresponding LRFs. The LRF has been justified by solid biological evidence, which shows that the visual cortex cell responds only to the subregion of the retina (i.e., input layer). L-ELM learns the local structures and generates more meaningful representations at the hidden layer when dealing with image processing and similar tasks. In this letter, we extend the L-ELM to a multilayer architecture, which is called HL-ELM.

The schematic diagram of our proposed HL-ELM-based HSI classification method is shown in Fig. 1. The layers of HL-ELM can be divided into two parts, namely, the feature extractor and the ELM classifier. The feature extractor is composed of hierarchical convolution and pooling layers. The feature maps in convolution layers are determined by the filters (convolution kernels) sliding on the previous layer pixel by pixel, and the pooling maps in pooling layers are determined by the pooling function on the nonoverlapped pooling fields over the previous convolution layer. Thereafter, the last pooling layer outputs the extracted features to the ELM classifier. As can be seen, the role of the activation function $g(x)$ in the original ELM is replaced by hierarchical convolution and pooling layers in HL-ELM.

To achieve more accurate classification performance, for each pixel of HSI data, the contextual information is involved by using a square neighbor window. Assuming that the size of the neighbor window is $w\times w$, then the spectral reflective value of a pixel in a specific band becomes the mean value of the w^2 pixels in the neighborhood. These mean values of all bands then form an average spectrum vector for a pixel, which is the input of HL-ELM. It is obvious that only spectral information is used if the size of neighbor window w is equal to 1.

Let $\mathbf{x}_1,\ldots,\mathbf{x}_N\in\mathbb{R}^D$ denote the average spectrum vectors of a HSI, where D is the number of spectral bands. Suppose that L is the number of convolution layers and K^l is the number of feature maps in layer l. The convolution layers of HL-ELM are denoted by $C^1,\ldots,C^L$, and the pooling layers are denoted by $P^1,\ldots,P^L$. For the convolution operation in layer l, assuming that the convolution kernel size (i.e., receptive field size) of layer l is r^l and the dimension of data in the previous layer is d^{l-1}, then the feature map after the convolution will have the size of $(d^{l-1}-r^l+1)$. Supposing that P^{l-1} denotes the input data of layer l, then the ith node in the kth convolution feature map, $C^l_{i,k}$, can be calculated as follows:

$$C^l_{i,k}=\sum_{t=1}^{K^{l-1}}\sum_{m=0}^{r^l-1}P^{l-1}_{i+m,t}\cdot W^l_{m,t,k} \tag{4}$$

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

TABLE I
INDIAN PINES CLASSES, TRAIN/TEST SET, AND CLASSIFICATION ACCURACY (IN PERCENT) FOR DIFFERENT METHODS

Class	Samples		Classification Method							
	Train	Test	SVM	ELM	CSVM	CELM	L-ELM	HL-ELM	CNN	SADL
Corn-notill	287	1147	84.73	85.64	94.16	95.45	96.98	**97.63**	95.99	96.89
Corn-mintill	167	667	81.41	50.16	96.30	88.10	98.13	**99.13**	98.15	98.43
Grass/Pasture	100	397	95.97	78.82	96.07	93.05	96.68	**96.80**	94.16	94.71
Grass/Trees	150	597	96.72	96.92	99.05	98.99	99.36	98.66	99.13	**99.43**
Hay-windrowed	98	391	99.57	**100**	99.34	**100**	**100**	99.72	99.86	**100**
Soybean-notill	194	774	76.10	59.38	89.25	88.54	93.28	96.25	92.76	**97.84**
Soybean-mintill	494	1974	87.46	81.13	96.13	96.06	97.72	98.98	98.32	**99.04**
Soybean-clean	123	491	88.04	62.89	95.46	89.00	95.60	**98.49**	97.32	96.54
Woods	259	1035	95.58	96.96	97.78	99.24	98.72	99.03	97.68	**99.52**
Buildings-Grass-Trees-Drives	76	304	70.99	49.97	94.31	91.25	94.53	97.83	**98.27**	97.04
Overall Accuracy (OA)			87.64±0.38	78.76±0.94	95.66±0.33	94.60±0.79	97.27±0.32	**98.36±0.44**	97.19±0.48	98.24±0.15
Average Accuracy (AA)			87.66±0.45	76.19±1.89	95.78±0.53	93.97±1.24	97.10±0.46	**98.25±0.46**	97.16±0.74	97.94±0.24
Kappa Coefficient (κ)			85.64±0.44	75.07±1.20	94.96±0.39	93.72±0.93	96.82±0.37	**98.09±0.51**	96.74±0.55	97.96±0.18
Training Time (seconds)			0.53	2.19	0.42	5.06	6.63	44.12	754.13	69.38

where $l \geq 1, i = 0, 1, \ldots, (d^{l-1} - r^l)$ and $W^l \in \mathbb{R}^{r^l \times K^{l-1} \times K^l}$ is the random weight matrix. It should be noticed that, for the input layer of HL-ELM, the value of l is 0, the input data are denoted by P^0 with the size of $d^0 = D$, and the number of feature maps is denoted by $K^0 = 1$ for consistency.

The square and root pooling structure is applied in the pooling layers. Assuming that the pooling size of layer l is s^l, the ith node in the kth pooling map, $P^l_{i,k}$, can be calculated as follows:

$$P^l_{i,k} = \sqrt{\sum_m \left[C^l_{(i \cdot s^l + m),k}\right]^2} \tag{5}$$

where $0 \leq m < s^l$. The square and rooting pooling introduces rectification nonlinearity and translational invariance into the network, which has been proved to be effective in computer vision area [8].

Simply concatenating all combinatorial node values into a vector, the last pooling layer is fully connected to the output layer. The output weight matrix can be analytically calculated as in (3).

To get better generalization performance, a regularized ELM model [16] is used by adding a constraint to the output weights. The optimization object becomes

$$\min_{\beta \in \mathbb{R}^{M \times c}} \|\mathbf{T} - \mathbf{H}\beta\|_F^2 + \lambda \|\beta\|_F^2 \tag{6}$$

where $\|\cdot\|_F$ is the Frobenius norm and λ is the parameter that controls the tradeoff between the training error and the norm of the output weights. Moreover, the output weights can be calculated as

$$\beta = \left(\frac{\mathbf{I}}{\lambda} + \mathbf{H}^T\mathbf{H}\right)^{-1} \mathbf{H}^T\mathbf{T} \tag{7}$$

where $\mathbf{I}$ is the identity matrix.

B. Training and Test Procedure of HL-ELM

The training process of the proposed HL-ELM-based HSI classification framework is described as follows. The feature maps in the convolution layers are calculated through the convolution operation in a weight-sharing manner, i.e., the nodes in the same feature map share the same convolution kernel. The nodes in the pooling layer are then obtained by the square/rooting pooling operation. Subsequently, the nodes in the pooling maps are combined into a feature vector and connected to c output nodes, where c is the number of land-cover classes. The training is completed when the output weights of ELM have been analytically calculated. It is notable that both convolutional neural networks (CNNs) [17] and HL-ELM use the convolution and pooling operations to extract features. However, they adopt different training strategies. The CNN uses the BP method to update the input weights and output weights iteratively. By contrast, the input weights of HL-ELM are randomly generated and kept unchanged. For HL-ELM, the output weights are the only network weights to be trained, and these weights are analytically calculated.

When the architecture and weights are specified, the HL-ELM model can be used as a classifier for HSI data. As expressed in (2), the classification results can be obtained with a forward-propagation step. The index of the node with the largest value in the output layer is regarded as the predicted label of the current pixel.

IV. EXPERIMENTS

A. HSI Data and Experimental Settings

Experiments were conducted on the Airborne Visible/ Infrared Imaging Spectrometer (AVIRIS) Indian Pines and Reflective Optics System Imaging Spectrometer (ROSIS) Pavia University data sets. The Indian Pines image consists of 145×145 pixels with a spatial resolution of 20 m/pixel. It has 220 spectral bands across the wavelength range of 0.4–2.5 μm. We removed 20 noisy bands (104–108, 150–163, and 220) affected by atmospheric water absorption and used the remaining 200 bands for experiment. Six land-cover classes are discarded for their small size, which results in a ten-class classification problem. There are 9725 samples except the background in total. The specific classes and the number of training and test data in each class are listed in Table I. The Pavia University image consists of 610×340 pixels with a spatial resolution of 1.3 m/pixel. It has 103 bands, and nine land-cover classes are considered. A total of 42 776 samples excluding the background

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

TABLE II
PAVIA UNIVERSITY CLASSES, TRAIN/TEST SET, AND CLASSIFICATION ACCURACY (IN PERCENT) FOR DIFFERENT METHODS

Class	Samples		Classification Method							
	Train	Test	SVM	ELM	CSVM	CELM	L-ELM	HL-ELM	CNN	SADL
Asphalt	548	6083	86.47	77.27	93.04	90.96	96.18	96.14	95.78	**97.27**
Meadows	540	18109	88.99	77.53	97.04	96.98	98.18	**99.63**	98.51	98.62
Gravel	392	1707	75.58	80.14	88.20	84.17	93.53	**97.38**	87.74	97.33
Trees	524	2540	97.43	95.69	97.96	97.76	98.17	99.23	96.93	**99.45**
Metal Sheets	265	1080	99.52	99.69	99.75	**99.99**	99.94	99.90	99.76	99.97
Bare Soil	532	4497	87.44	80.47	98.35	95.75	**99.92**	99.85	98.36	99.87
Bitumen	375	955	91.01	82.97	94.16	93.82	**99.27**	98.70	97.32	99.25
Bricks	514	3168	87.60	70.27	93.60	87.48	96.75	96.79	**96.82**	96.74
Shadows	231	716	99.25	93.49	99.43	96.24	94.78	91.37	92.42	**99.94**
Overall Accuracy (OA)			88.80±0.38	79.58±0.66	96.01±0.25	94.60±0.34	97.76±0.12	**98.59±0.17**	97.24±0.69	98.47±0.25
Average Accuracy (AA)			90.36±0.15	84.17±0.40	95.73±0.20	93.68±0.30	97.41±0.21	97.67±0.28	95.96±0.81	**98.72±0.14**
Kappa Coefficient (κ)			84.97±0.48	73.26±0.76	94.57±0.33	92.65±0.46	96.94±0.16	**98.07±0.24**	96.23±0.94	97.91±0.33
Training Time (seconds)			0.61	1.42	1.06	42.66	11.49	83.31	1311.12	195.94

were used in this data set. The number of training and test samples for each class is provided in Table II.

The classification performance of the proposed HL-ELM method was evaluated by comparing it with those of several classical or state-of-the-art classification approaches, including SVM [18], ELM [4], contextual SVM (CSVM), contextual ELM (CELM), L-ELM, CNN [15], and spatial-aware dictionary learning (SADL) [19]. The SVM and ELM methods use only spectral information, while the other methods use both spectral and contextual information. The neighbor window size w of the contextual methods was chosen from {3, 5, 7, 9}. For the SVM and CSVM classification, we applied the one-versus-one strategy using the LIBSVM[1] Toolbox. A radial basis function (RBF) kernel was adopted; the penalty term and the RBF kernel width were selected using grid search within the given sets $\{2^{-5}, \ldots, 2^{15}\}$ and $\{2^{-15}, \ldots, 2^{3}\}$, respectively. These parameters were determined by a fivefold cross-validation method, which was also used for the other methods. Meanwhile, the tradeoff parameter λ for ELM and CELM was chosen from a set $\{10^{-4}, \ldots, 10^{4}\}$ (this range was also used in L-ELM and HL-ELM), and the hidden neuron number M was chosen from $\{500, \ldots, 3000\}$. For both L-ELM and HL-ELM, the pooling size was set to 2 for both data sets, and the feature map number was allowed to obtain value from $\{20, \ldots, 150\}$. Through cross-validation, the convolution kernel size (LRF size) for L-ELM was empirically set to 13 for the Indian Pines image and 20 for the Pavia University image. For HL-ELM, the selection of LRF sizes will be discussed in Section IV-C. For the CNN method, the MatConvNet[2] Toolbox was used for efficient implementation, and the same convolution and pooling structures were used as HL-ELM. For SADL, we used the settings as reported therein.

B. Classification Results

The different approaches for the Indian Pines data set are compared in Table I, where the classification accuracy for each class (CA), overall accuracy (OA), average accuracy (AA), and κ coefficient are reported. These results were averaged over ten runs, and their standard deviations are also reported. Several conclusions can be made from the table. First, SVM and ELM perform poorly compared with the other contextual methods, and this result stresses the importance of contextual information for HSI classification. Second, CNN achieves comparable results with L-ELM, which are higher than that of the CSVM and CELM methods, and SADL gains about 1% improvement than CNN. Third, the proposed HL-ELM method obtains the best classification results among these methods, with the OA of 98.36%, AA of 98.25%, and κ coefficient of 0.9809. The classification maps of different methods are visually shown in Fig. 2 (see Table I for the legend color).

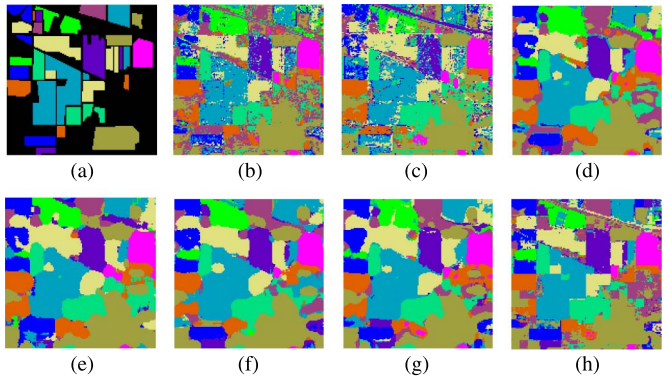

Fig. 2. Classification maps for the Indian Pines image. (a) Ground truth. (b) SVM. (c) ELM. (d) CSVM. (e) CELM. (f) HL-ELM. (g) CNN. (h) SADL.

Experiments were also conducted using the Pavia University image, and Table II shows the numerical results of the CA, OA, AA, and κ coefficient of the different methods. The best results of these methods are denoted in bold. The results show that HL-ELM also outperforms the other methods on this data set.

The last lines of Tables I and II present the training time of each approach in comparison, which is the average value over ten runs. All programs were implemented in MATLAB R2014a and run on a computer with two eight-core Intel Xeon

[1]http://www.csie.ntu.edu.tw/~cjlin/libsvm
[2]http://www.vlfeat.org/matconvnet

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

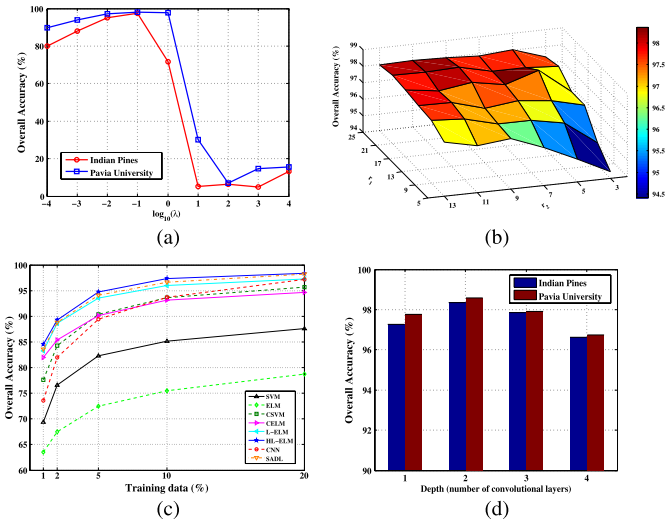

Fig. 3. Impacts on OA of HL-ELM by different factors. (a) Effect of tradeoff parameter λ on two data sets. (b) Effect of LRF size on Indian Pines data set. Note that r_1 and r_2 represent the LRF size of convolution layers C1 and C2, respectively. (c) Effect of training data size on Indian Pines data set. (d) Effect of depth (number of convolution layers) on two data sets.

E5-2650 processors at 2.0 GHz and 128 GB of memory. To accelerate the computing of CNN, an NVIDIA Tesla K20C graphics processing unit (GPU) was utilized as the coprocessor, using the CUDA 6.5 library. The results demonstrate the high efficiency of our proposed classification strategy while yielding high classification accuracy.

C. Discussion

The tradeoff parameter λ in (7) and LRF sizes are important parameters for HL-ELM. Fig. 3(a) shows the impact on the OA of λ. It can be observed that OA reaches the highest value on both Indian Pines and Pavia University data sets when λ is 0.1. Based on this, we set $\lambda = 0.1$ for HL-ELM in our experiments. Next, we evaluate how the OA is affected by the LRF sizes on the Indian Pines data set. The LRF of the first convolution layer (r_1 in Fig. 1) was chosen from $\{5, \ldots, 25\}$, and the LRF of the second convolution layer (r_2 in Fig. 1) was chosen from $\{3, \ldots, 13\}$. Fig. 3(b) shows the effect on the OA of the LRF sizes. It can be seen that the impact of LRF sizes is less sensitive than λ and OA reaches the highest value when $r_1 = 17$ and $r_2 = 5$.

To further compare the different classification methods, the effect of different training sample sizes has been examined using the Indian Pines image. Fig. 3(c) shows the OA of each method for 1%, 2%, 5%, 10%, and 20% training data averaged over ten runs. As the results show, the classification accuracies increase along with the number of the training samples. The SADL, L-ELM, and HL-ELM methods provide a large improvement for a small number of training samples with L-ELM trailing slightly behind. It can also be observed that CNN falls further behind as the number of training samples decreases.

Next, we evaluate the effect on the OA of depth. Here, the depth refers to the number of convolution layers. Fig. 3(d) illustrates that, when the depth increases from 2 to 3 or 4, the OA obtains close or even lower values. However, the running time increases notably. Thus, we use the HL-ELM with two convolution layers in our experiments.

V. Conclusion

In this letter, we have proposed a novel HL-ELM model for HSI classification. By introducing LRFs, the relevance between the spectral features of HSI data can be used more sufficiently. With a hierarchical architecture, multiple levels of abstraction and invariant features can be learned. The experimental results derived from two real hyperspectral data sets demonstrate the effectiveness of the proposed approach, as evidenced by the high classification performance and fast training speed.

References

[1] J. M. Bioucas-Dias *et al.*, "Hyperspectral remote sensing data analysis and future challenges," *IEEE Geosci. Remote Sens. Mag.*, vol. 1, no. 2, pp. 6–36, Jun. 2013.

[2] G. B. Huang, Q. Y. Zhu, and C. Siew, "Extreme learning machine: Theory and applications," *Neurocomputing*, vol. 70, no. 1–3, pp. 489–501, Dec. 2006.

[3] H. Gao, G. B. Huang, S. Song, and K. You, "Trends in extreme learning machines: A review," *Neural Netw.*, vol. 61, pp. 32–48, Jan. 2015.

[4] M. Pal, "Extreme-learning-machine-based land cover classification," *Int. J. Remote Sens.*, vol. 30, no. 14, pp. 3835–3841, Jul. 2009.

[5] Y. Bazi *et al.*, "Differential evolution extreme learning machine for the classification of hyperspectral images," *IEEE Geosci. Remote Sens. Lett.*, vol. 11, no. 6, pp. 1066–1070, Jun. 2014.

[6] A. Samat, P. Du, S. Liu, J. Li, and L. Cheng, "E^{2}LMs: Ensemble extreme learning machines for hyperspectral image classification," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 7, no. 4, pp. 1060–1069, Apr. 2014.

[7] C. Chen, W. Li, H. Su, and K. Liu, "Spectral-spatial classification of hyperspectral image based on kernel extreme learning machine," *Remote Sens.*, vol. 6, no. 6, pp. 5795–5814, Jun. 2014.

[8] G. B. Huang, Z. Bai, L. L. Kasun, and C. M. Vong, "Local receptive fields based extreme learning machine," *IEEE Comput. Intell. Mag.*, vol. 10, no. 2, pp. 18–29, May 2015.

[9] L. Zhang *et al.*, "Ensemble manifold regularized sparse low-rank approximation for multiview feature embedding," *Pattern Recognit.*, vol. 48, no. 10, pp. 3102–3112, Oct. 2015.

[10] L. Zhang, L. Zhang, D. Tao, and X. Huang, "On combining multiple features for hyperspectral remote sensing image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 50, no. 3, pp. 879–893, Mar. 2012.

[11] J. Schmidhuber, "Deep learning in neural networks: An overview," *Neural Netw.*, vol. 61, pp. 85–117, Jan. 2015.

[12] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, May 2015.

[13] Y. Chen, Z. Lin, X. Zhao, G. Wang, and Y. Gu, "Deep learning-based classification of hyperspectral data," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 7, no. 6, pp. 2094–2107, Jun. 2014.

[14] Y. Chen, X. Zhao, and X. Jia, "Spectral-spatial classification of hyperspectral data based on deep belief network," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 8, no. 6, pp. 2381–2392, Jun. 2015.

[15] J. Yue, W. Zhao, S. Mao, and H. Liu, "Spectral-spatial classification of hyperspectral images using deep convolutional neural networks," *Remote Sens. Lett.*, vol. 6, no. 6, pp. 468–477, Jun. 2015.

[16] G. B. Huang, H. Zhou, X. Ding, and R. Zhang, "Extreme learning machine for regression and multiclass classification," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 42, no. 2, pp. 513–529, Apr. 2012.

[17] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proc. IEEE*, vol. 86, no. 11, pp. 2278–2324, Nov. 1998.

[18] F. Melgani and L. Bruzzone, "Classification of hyperspectral remote sensing images with support vector machines," *IEEE Trans. Geosci. Remote Sens.*, vol. 42, no. 8, pp. 1778–1790, Aug. 2004.

[19] A. Soltani-Farani, H. R. Rabiee, and S. A. Hosseini, "Spatial-aware dictionary learning for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 1, pp. 527–541, Jan. 2015.