

Deep Hashing Neural Networks for Hyperspectral Image Feature Extraction

Leyuan Fang, *Senior Member, IEEE*, Zhiliang Liu, Weiwei Song, *Student Member, IEEE*

Abstract—Recently, deep learning has been recognized as a powerful tool to extract hierarchical features of hyperspectral images (HSIs). The existing deep learning-based methods exploit label information of land classes as the supervised information to train deep networks. However, considering that HSIs exhibit very complex spectral-spatial characteristic, e.g., the large intraclass variations and small interclass variations, these semantic information (i.e., label information) based deep networks may not effectively cope with the above problem. In this letter, we propose a novel deep model, named deep hashing neural network (DHNN), to learn similarity-preserving deep features (SPDF) for HSI classification. Firstly, a well pretrained network is introduced to simultaneously extract features of a pair of input samples. Secondly, a novel hashing layer is inserted after the last fully connected layer to transfer the real-value features into binary features, which can significantly speed up the computation for feature distance. Then, a loss function is elaborately designed to minimize the feature distance of similar pairs and maximize the feature distance of dissimilar pairs in Hamming space. Finally, the SPDF extracted by propagating the samples through the trained DHNN are fed into a support vector machine (SVM) classifier for HSI classification. Experimental results on two real HSIs demonstrate that the proposed feature extraction method in conjunction with a linear SVM classifier outperforms other feature extraction methods and competitive classifiers.

Index Terms—Hyperspectral images, deep learning, feature extraction, classification, hashing.

I. INTRODUCTION

HYPERSPECTRAL images (HSIs) usually contain hundreds of spectral bands spanning from the visible-to-infrared spectrum. With the rich spectral information, HSI classification, which categorizes each hyperspectral pixel into a certain class based on its spectral characteristics, has become an active and important task and drawn broad attentions in the remote sensing field [1]. For HSI classification, one of the challenges is the huge spectral dimension, which may cause the curse of dimensionality. In the early stage of the study on HSI classification, most methods mainly focus on exploring the role of the spectral signatures of HSIs for the purpose of classification [2], [3]. However, those methods are not suitable for the analysis of hyperspectral data with

This work was supported in part by the National Natural Science Foundation under Grant No. 61771192, the National Natural Science Foundation for Young Scientist of China under Grant No. 61501180, China Postdoctoral Science Foundation funded Project No. 2017T100597, and National Natural Science Foundation of Hunan Province Grant No. 2018JJ3077.

L. Fang, Z. Liu, and W. Song are with the College of Electrical and Information Engineering, Hunan University, Changsha, 410082, China, and also with the Key Laboratory of Visual Perception and Artificial Intelligence of Hunan Province, Changsha, 410082, China (e-mail: fangleyuan@gmail.com; liuzhilian@hnu.edu.cn; weiweisong415@gmail.com).

an inherently nonlinear relation between the captured spectral information and the corresponding materials. In this situation, many kernel-based approaches [4], [5] have been proposed to cope with the Hughes phenomenon [6]. The main idea is to transform the inseparable samples in low-dimensional space into separable samples in high-dimensional space by means of nonlinear transformation.

In the past several years, many studies have reported that spatial features are very useful in improving the representation of hyperspectral data and increasing the classification accuracies [7]–[10]. More and more spectral-spatial methods have been proposed for HSI classification. For instance, extended morphological profiles (EMPs) were introduced to explore the spatial information of hyperspectral data via multiple morphological operations [11]. In [12], the spatial contexture within a neighboring region was incorporated into a sparse representation model, where a hyperspectral pixel can be represented by a linear combination of a few common pixels from the same class. In addition, some postprocessing techniques were also introduced to classify HSI, e.g., edge-preserving filtering (EPF) [13] and Markov random field (MRF) [14] were used to refine the classification results obtained by SVM. However, these traditional classification methods heavily depend on the experts knowledge and parameter setting, the extracted features lack of good ability of representation and robustness for HSIs. Thus, designing a satisfactory feature-extraction framework is considered as a crucial procedure for HSI classification task.

Recently, deep learning has gained wide attention and achieved great breakthrough in many computer vision tasks [15]. As a powerful feature-extraction tool, deep networks can automatically extract high-level features with a large number of hierarchical layers. Motivated by these successful applications, deep learning has also been introduced to classify HSIs in the field of remote sensing. For instance, a CNN-based spatial-spectral feature extraction framework was employed for HSI classification [16], [17]. In [18], the spectral and spatial features were extracted by a spectral long short term memory (LSTM) and spatial LSTM, respectively. More advanced, in [19], deep residual networks [20] were also introduced to learn deeper features of HSIs with identity mapping mechanism. Most existing deep learning-based methods exploit semantic information (i.e., label information) of hyperspectral pixels as supervised information to train deep networks. However, HSIs usually exhibit very complex spectral-spatial characteristic, e.g., the large intraclass variations and small interclass variations, these semantic information-based features may not effectively discriminate those classes. In order to cope with the

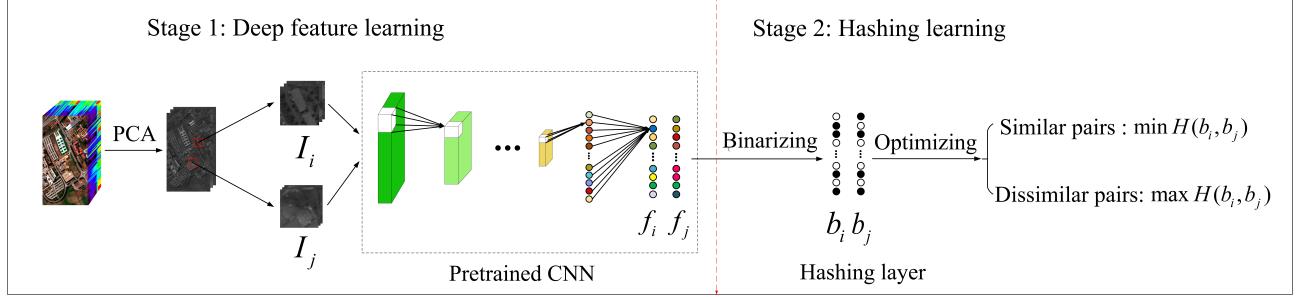


Fig. 1. The proposed DHNN based HSI feature extraction framework, which can be divided into deep feature learning and hashing learning. The first stage aims at learning deep features via a well pretrained CNN. In the second stage, the hashing layer is inserted after the last fully connected layer to transfer the high-dimension and real-value features into the low-dimension and binary features.

above problems, a few works focus on learning the correlation between hyperspectral pixels. For instance, Li *et al.* used a CNN to extract pixel-pair features, where the network input consists of two arbitrary hyperspectral pixels in training phase [21]. The main drawback of this method is that the spatial information is not considered since the convolution operation is mainly executed in the spectral domain. In [22], a siamese CNN (S-CNN) was employed to extract the spectral-spatial features, such that the feature distance between samples of the same class is small and the feature distance between samples of the different classes is large. Nevertheless, the adopted distance function is Euclidean function which is not efficient for computing high-dimension feature.

In this letter, we propose a novel deep hashing neural network (DHNN) which incorporates the hashing learning into a CNN to extract the SPDF for HSI classification. Firstly, we use a well pretrained network to simultaneously extract deep features of a pair of samples. Secondly, a hashing layer is inserted after the last fully connected layer to transform the high-dimension real-value features into the low-dimension compact binary features, which can significantly speed up the computation for feature distance. Then, a loss function is elaborately designed to minimize the feature distance of similar pairs and maximize the feature distance of dissimilar pairs. Finally, the SPDF obtained by propagating all samples through the trained DHNN are fed into an SVM classifier for HSI classification. Different from the previous methods that only consider the semantic information of individual pixel, the proposed method aims at exploiting the correlation between pixels in HSIs. Furthermore, hashing technique is introduced to learn the compact binary codes. To the best of our knowledge, this is the first time to integrate the hashing learning into deep networks for HSI classification.

The remainder of this letter is organized as follows. In Section II, the methodology is introduced in detail. In Section III, the experimental results and analysis are presented. Section IV makes some concluding remarks.

II. METHODOLOGY

Due to the complex imaging condition, HSI classification faces with many challenges. For instance, different objects may share similar spectral properties or the spectral curves of same objects are various. Most existing deep learning-based

methods use label information as supervised information to train deep models, which may not effectively solve the above problem. In this letter, we propose a novel feature extraction method to learn the SPDF for HSI classification. Fig. 1 shows the whole framework, which can be mainly divided into deep feature learning and hashing learning. In the following parts, we will describe the procedures in detail.

A. Deep Feature Learning

In recent years, deep networks have demonstrated great potential to extract the robust and discriminative features for classification tasks [15], [23]. CNN is the most popular deep model due to its local connection and weight sharing. Training a deep network from scratch requires a larger number of samples to learn the model parameters. However, obtaining labeled data is expensive and time consuming in the remote sensing field. To solve the above problem, we introduce transfer learning to decrease the burden on training samples. Specifically, we transfer the network parameters of CNN-S [24] which was pretrained on the ImageNet ILSVRC [25] and fine-turned on the PASCAL VOC [26] to our DHNN, including the first five convolutional layers and two fully connected layers. The detailed configuration of feature-learning network is shown in Table I. We use the following aspects to describe convolutional layers: “filter” specifies the number of convolution filters and kernel size; “stride” and “pad” indicate the convolutional strides and spatial padding, respectively; “LRN” refers to Local Response Normalization [15]; “pool” specifies the max-pooling size. For fully connected layers, “4096” indicates the feature dimension and the dropout technique [15] is applied to full6 and full7. The activation function for all weight layers is the REctification Linear Unit (RELU) [15]. Please note that we only adopt a relatively simple network (i.e., CNN-S [24]) as our feature extraction model. We can also replace the CNN-S with some recent powerful networks, e.g., VGG [27] and ResNet [20]. However, the focus of this letter is to verify the effectiveness of combination between deep learning and hashing learning, rather than the investigation of different deep networks.

Consider a hyperspectral data set with N labeled samples denoted as $\mathbf{X} = \{\mathbf{x}_i\}_{i=1}^N$ in an \mathbb{R}^d , where d refers to the number of bands. In order to comprehensively utilize the

TABLE I
THE CONFIGURATION OF DEEP NETWORK USED IN FEATURE LEARNING PROCEDURE

Layer	Configuration
conv1	filter $96 \times 7 \times 7$, stride 2×2 , pad 0, LRN, pool 3×3
conv2	filter $256 \times 5 \times 5$, stride 1×1 , pad 1, pool 2×2
conv3	filter $512 \times 3 \times 3$, stride 1×1 , pad 1
conv4	filter $512 \times 3 \times 3$, stride 1×1 , pad 1
conv5	filter $512 \times 3 \times 3$, stride 1×1 , pad 1, pool 3×3
full6	4096, dropout
full7	4096, dropout

spectral and spatial information to improve the performance of classification, we extend each spectral pixel to an image patch with the size of $q \times q \times d$ by choosing a $q \times q$ neighborhood of the central pixel. Let s_{ij} be the label of a pair of image patch $(\mathbf{x}_i, \mathbf{x}_j)_{i,j=1}^N$, we have the following definition $s_{ij} = 1$ if \mathbf{x}_i and \mathbf{x}_j come from same class and 0 otherwise. Through the above mentioned deep network, the deep features $(\mathbf{f}_i, \mathbf{f}_j)$ can be easily obtained by

$$\mathbf{f}_t = \Phi(\mathbf{W}, \mathbf{b} | \mathbf{x}_t), t = i, j \quad (1)$$

where Φ is the network function characterized by the network weight \mathbf{W} and bias \mathbf{b} . This propagation actually performs a series of nonlinear and linear transformations, including convoluting, pooling, and nonlinear mapping.

B. Hashing Learning

In order to obtain SPDF, the deep features extracted in above step are constrained to preserve the similarity of original space. More specifically, the feature distance of similar pairs should be as small as possible, the dissimilar pairs are expected to be away from each other in feature space. An intuition idea to achieve the above object is to adopt Euclidean distance as the metric to measure the similarity degree between deep features [22], which can be represented as

$$D = \|\mathbf{f}_i - \mathbf{f}_j\|_2^2. \quad (2)$$

However, the Euclidean distance is not a feasible choice when the feature dimension is very high. In order to achieve effective computation for feature distance, we incorporate hashing learning into a deep network. Specifically, a hashing layer is inserted after the last fully connected layer to transfer the high-dimension and real-value features into low-dimension and binary features, which can be formulated as

$$\mathbf{b}_t = sgn(\mathbf{f}_t), t = i, j \quad (3)$$

where $sgn(\cdot)$ performs element-wise operations for a matrix or a vector, i.e., $sgn(x) = 1$ if $x > 0$ and -1 otherwise.

Once obtaining the binary codes $\mathbf{B} = \{\mathbf{b}_t\}_{t=1}^N$ for all the samples, the likelihood of the pairwise labels $S = \{s_{ij}\}$ can be defined as

$$p(s_{ij} | \mathbf{B}) = \begin{cases} \sigma(\omega_{ij}), & s_{ij} = 1 \\ 1 - \sigma(\omega_{ij}) & s_{ij} = 0 \end{cases} \quad (4)$$

where $\sigma(\cdot)$ is the logistic function and $\sigma(x) = \frac{1}{1+e^{-x}}$, $\omega_{ij} = \frac{1}{2}\mathbf{b}_i^T \mathbf{b}_j$. Based on the above definition, the loss function can

be given by taking the negative log-likelihood of the observed pairwise labels in S

$$\begin{aligned} \mathcal{L}_1 &= -\log p(S | \mathbf{B}) = -\sum_{s_{ij} \in S} \log p(s_{ij} | \mathbf{B}) \\ &= -\sum_{s_{ij} \in S} (s_{ij} \omega_{ij} - \log(1 + e^{\omega_{ij}})). \end{aligned} \quad (5)$$

Through minimizing the above loss function, the feature distance in Hamming space (i.e., Hamming distance) between two similar samples can be optimized as small as possible, and the Hamming distance between two dissimilar samples becomes as large as possible. However, directly solving the problem (5) is very hard due to the discrete values in formulation. Motivated by [28], the above loss function can be reformulated in a discrete way

$$\begin{aligned} \mathcal{L}_2 &= -\sum_{s_{ij} \in S} (s_{ij} \psi_{ij} - \log(1 + e^{\psi_{ij}})) \\ &\quad + \beta \sum_{i=1}^N \|\mathbf{b}_i - \mathbf{f}_i\|_2^2 \end{aligned} \quad (6)$$

where $\psi_{ij} = \frac{1}{2}\mathbf{f}_i^T \mathbf{f}_j$, $i, j = 1, 2, \dots, N$, β is a regularization parameter which can constrain \mathbf{f}_i approach to \mathbf{b}_i .

C. Classification

After designing the loss function, the DHNN can be trained in an end-to-end way with the stochastic gradient descent (SGD) algorithm. Once the network is trained enough, we can easily obtain the SPDF by propagating all samples through the trained DHNN. Particularly, for an unknown sample \mathbf{x}_k , the feature \mathbf{f}_k can be computed by the network function Φ

$$\mathbf{f}_k = \Phi(W, b | \mathbf{x}_k), k = 1, 2, \dots, N. \quad (7)$$

Finally, in order to evaluate the effectiveness of the learned features, these features are fed into an SVM classifier [3] for the subsequent classification.

III. EXPERIMENTS

A. Experimental Data and Setting

In this section, we conduct several experiments to verify the effectiveness of the proposed method on two real hyperspectral data, including the University of Pavia and Salinas scenes.

The University of Pavia image, which captures an urban area over the University of Pavia, Italy, was recorded by the ROSIS-03 sensor. This image is of size $610 \times 340 \times 115$ with a spatial resolution of 1.3 m per pixel. Nine classes of interest are considered for this image. Before the experiments, 12 very noisy bands were removed.

The Salinas image was captured by the AVIRIS sensor over the area of Salinas Valley, California. This image comprises of 512×217 pixels with a spatial resolution of 3.7 m. After removing 20 water absorption bands, there are 204 spectral bands are used in our experiments. The available ground reference map covers 16 classes of interest.

TABLE II
CLASS ACCURACIES (CA), OVERALL ACCURACIES (OA), AVERAGE ACCURACIES (AA), AND KAPPA COEFFICIENTS OF THE SVM [3], JSR [12], 3D-CNN [16], CNN-PPF [21], SVM+S-CNN [22], SPDF*-SVM, AND SPDF-SVM ON THE UNIVERSITY OF PAVIA IMAGE. ALL THE RESULTS ARE IN PERCENTAGE. THE STANDARD DEVIATION VALUES ARE ALSO GIVEN IN THE BRACKETS.

class	SVM	JSR	3D-CNN	CNN-PPF	SVM+S-CNN	SPDF*-SVM	SPDF-SVM
1	85.49	83.80	99.03	97.23	95.47	97.59	97.94
2	92.12	96.31	98.11	95.27	98.71	96.94	98.84
3	85.77	98.63	88.56	95.13	97.32	98.77	99.01
4	96.41	93.89	83.51	96.89	97.72	96.39	96.50
5	98.60	98.38	99.49	99.99	100	99.80	100
6	92.52	99.54	95.33	98.55	97.67	99.55	99.69
7	93.79	98.50	96.31	96.56	98.36	99.97	99.86
8	86.56	96.51	97.58	94.43	95.65	98.17	98.32
9	97.97	75.88	96.25	99.39	100	98.08	98.09
OA	90.78 (0.43)	94.43 (0.62)	96.37 (0.63)	97.63 (0.53)	97.93 (0.38)	97.72 (0.36)	98.61 (0.32)
AA	92.14 (0.54)	92.57 (0.85)	94.82 (0.89)	97.04 (0.75)	97.88 (0.49)	98.41 (0.51)	98.59 (0.45)
Kappa	0.8813 (0.0058)	0.9349 (0.0108)	0.9502 (0.0148)	0.969 (0.0098)	0.9743 (0.0045)	0.9700 (0.0048)	0.9791 (0.0041)

For the proposed classification method denoted as SPDF-SVM, there are three hyper-parameters that may effect the final classification performance. According to abundant of experiments, these parameters are set as followings: the length of hashing layer k is set to 64 and 48 for the University of Pavia and Salinas images, respectively; the patch size $q \times q$ is set to 25×25 and 37×37 for the University of Pavia and Salinas images, respectively; the regularization parameter β is set to 6 and 10 for the two images.

The performance of the proposed SPDF-SVM is also compared with other methods, including two classical machine learning-based methods, i.e., SVM [3], JSR [12], and three deep learning-based methods, i.e., 3D-CNN [16], CNN with pixel-pair features (CNN-PPF) [21], siamese CNN (S-CNN) with SVM classifier (SVM+S-CNN) [22]. All the parameters of the above compared methods are set to the default values in [3], [11], [16], [21], [22]. In addition, a counterpart denoted as SPDF*-SVM, which uses the Euclidean distance instead of Hamming distance to constrain the similarity of a pair of images, is also adopted as a compared method.

B. Experimental Results and Analysis

The first experiment was performed on the University of Pavia image, where 200 samples per class were randomly selected as training samples, and the rest of samples were regarded as test samples. The four objective metrics, i.e., the overall accuracy (OA), average accuracy (AA), and Kappa coefficient and class accuracies (CA), were adopted to evaluate the classification results. Table I gives quantitative results of various methods, which were averaged over ten random experiments. From this table, it can be seen that the proposed method shows the advantages in classifying the majority of classes (i.e., 5 out of 9 classes). In addition, the OA of SPDF-SVM is about 0.89% and higher than that of SPDF*-SVM, which demonstrates effectiveness of hashing learning.

TABLE III
CLASS ACCURACIES (CA), OVERALL ACCURACIES (OA), AVERAGE ACCURACIES (AA), AND KAPPA COEFFICIENTS OF THE SVM [3], JSR [12], 3D-CNN [16], CNN-PPF [21], SVM+S-CNN [22], SPDF*-SVM, AND SPDF-SVM ON THE SALINAS IMAGE. ALL THE RESULTS ARE IN PERCENTAGE. THE STANDARD DEVIATION VALUES ARE ALSO GIVEN IN THE BRACKETS.

class	SVM	JSR	3D-CNN	CNN-PPF	SVM+S-CNN	SPDF*-SVM	SPDF-SVM
1	99.61	100	99.94	99.84	100	99.89	100
2	99.69	99.83	85.45	99.77	97.75	99.94	99.97
3	99.56	99.92	100	98.11	98.88	100	100
4	99.41	94.77	99.77	99.57	100	99.94	99.90
5	98.72	83.31	99.96	98.54	99.93	99.72	99.97
6	99.77	92.34	100	99.92	89.48	99.84	99.91
7	99.52	95.92	99.60	99.96	99.30	99.83	99.91
8	76.74	96.97	99.31	89.11	98.69	97.75	98.18
9	99.37	99.99	99.97	99.69	99.34	100	100
10	95.13	92.42	99.41	97.78	100	99.96	100
11	99.32	93.86	100	99.33	99.93	99.94	100
12	99.70	83.94	100	99.89	99.94	100	
13	99.15	89.25	100	99.67	88.62	99.98	100
14	98.38	70.06	100	98.75	88.72	99.49	99.66
15	75.56	96.74	90.36	89.99	90.62	98.17	98.43
16	99.23	99.81	85.93	99.07	99.96	100	99.99
OA	90.92 (0.94)	95.33 (0.72)	97.28 (1.04)	94.87 (0.62)	97.62 (0.98)	99.20 (0.43)	99.34 (0.17)
AA	96.18 (1.03)	94.78 (0.82)	97.48 (1.24)	98.07 (0.27)	96.94 (0.36)	99.65 (0.17)	99.74 (0.08)
Kappa	0.8985 (0.0012)	0.9307 (0.0063)	0.9695 (0.0136)	0.9404 (0.0096)	0.9510 (0.0110)	0.9910 (0.0049)	0.9926 (0.0019)

The second experiment was conducted on the Salinas image. The training set consists of 200 samples per class, the rest of samples were regarded as test set. The classification results obtained by different methods are shown in Table II. From this table, we can see that the better classification results in terms of OA, AA and Kappa indexes obtained by the SPDF-SVM also demonstrate the overall superiority of the proposed method over the compared ones.

For the SPDF-SVM, there are three hyper-parameters that may effect the final classification performance. In the following part, their influences on OA are separately discussed in detail. Fig. 2 shows the effect of these parameters on OA values, where k , q , and β refer to the length of hashing layer, size of image patch, and regularization parameter existed in formulation 6, respectively. From the Fig. 2 (a), it can be observed that the OA values first improve, and then become stable as the k increases. The highest OA values are obtained when the k is set to 64 and 80 for the University of Pavia and Salinas images, respectively. As can be seen from Fig. 2 (b), the optimal q on the University of Pavia (i.e., 25×25) is smaller than one on the Salinas image (i.e., 37×37), which demonstrates the University of Pavia scene is more sensitive than the Salinas scene in terms of the size of input. The main reason is that the Salinas image has larger smooth regions and the University of Pavia image has more detailed regions. From the Fig. 2 (c), we can see that the effect of β on OA values is relatively steadily compared with other two parameters. The optimal β is set to 6 and 10 for the University of Pavia and Salinas images, respectively.

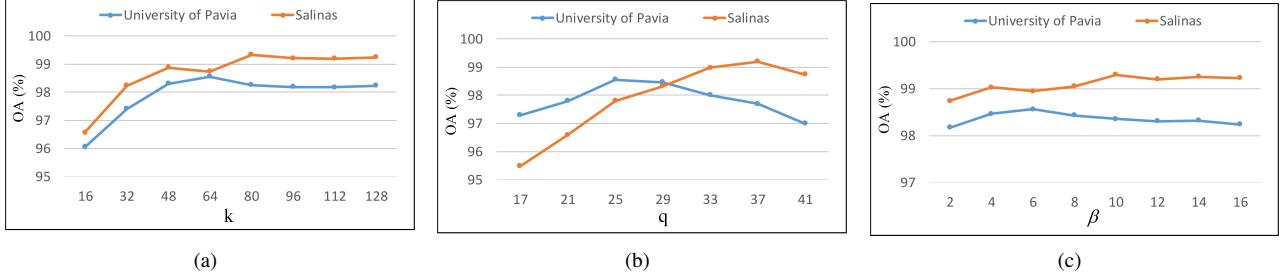


Fig. 2. Effect of three parameters on OA values. (a) the length of hashing layer (denoted as k) versus OA values, (b) the size of image patch (denoted as q) versus OA values, (c) the regularization parameter β versus OA values.

IV. CONCLUSION

Most existing deep learning-based methods only exploit semantic information (i.e., label information) of HSIs to extract deep features. However, these kind of methods may not be effective when HSIs exhibit very complex spectral-spatial characteristic. In this letter, a novel network named DHNN, which incorporates the hashing learning into a CNN, is proposed for HSI feature extraction. More specifically, a well trained CNN is used to extract deep features of HSIs with preserving great similarity information existed in the original space. In order to effectively compute the feature distance, a new hashing layer is inserted after the last fully connected layer to transfer the real-value features into binary features. Afterward, features extracted via DHNN are used to train a linear SVM classifier for HSI classification. The experimental results on two well-known hyperspectral data sets demonstrate that the proposed classification method i.e., SPDF-SVM, can greatly improve the classification performance.

REFERENCES

- [1] P. Ghamisi, *et al.*, “Advances in hyperspectral image and signal processing: A comprehensive overview of the state of the art,” *IEEE Geosci. Remote Sens. Mag.*, vol. 5, no. 4, pp. 37–78, Dec. 2017.
- [2] Y. Zhong and L. Zhang, “An adaptive artificial immune network for supervised classification of multi-/hyperspectral remote sensing imagery,” *IEEE Trans. Geosci. Remote Sens.*, vol. 50, no. 3, pp. 894–909, Mar. 2012.
- [3] 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.
- [4] G. Camps-Valls and L. Bruzzone, “Kernel-based methods for hyperspectral image classification,” *IEEE Trans. Geosci. Remote Sens.*, vol. 43, no. 6, pp. 1351–1362, 2005.
- [5] G. Camps-Valls, L. Gomez-Chova, J. Munoz-Mari, J. Vila-Frances, and J. Calpe-Maravilla, “Composite kernels for hyperspectral image classification,” *IEEE Geosci. Remote Sens. Lett.*, vol. 3, no. 1, pp. 93–97, 2006.
- [6] G. Hughes, *et al.*, “On the mean accuracy of statistical pattern recognizers,” *IEEE Trans. Inf. Theory*, vol. 14, no. 1, pp. 55–63, Jan. 1968.
- [7] M. Fauvel, Y. Tarabalka, J. A. Benediktsson, J. Chanussot, and J. C. Tilton, “Advances in spectral-spatial classification of hyperspectral images,” *Proc. of the IEEE*, vol. 101, no. 3, pp. 652–675, 2013.
- [8] R. Hang and Q. Liu, “Dimensionality reduction of hyperspectral image using spatial regularized local graph discriminant embedding,” *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 11, no. 9, pp. 3262–3271, Sep. 2018.
- [9] B. Tu, X. Zhang, X. Kang, G. Zhang, and S. Li, “Density peak-based noisy label detection for hyperspectral image classification,” *IEEE Trans. Geosci. Remote Sens.*, DOI: 10.1109/TGRS.2018.2867444, to be published, 2019.
- [10] B. Tu, X. Yang, N. Li, X. Ou, and W. He, “Hyperspectral image classification via superpixel correlation coefficient representation,” *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, , no. 99, pp. 1–15, 2018.
- [11] J. A. Benediktsson, J. Palmason, and J. R. Sveinsson, “Classification of hyperspectral data from urban areas based on extended morphological profiles,” *IEEE Trans. Geosci. Remote Sens.*, vol. 43, no. 3, pp. 480–491, Mar. 2005.
- [12] Y. Chen, N. M. Nasrabadi, and T. D. Tran, “Hyperspectral image classification using dictionary-based sparse representation,” *IEEE Trans. Geosci. Remote Sens.*, vol. 49, no. 10, pp. 3973–3985, Oct. 2011.
- [13] X. Kang, S. Li, and J. A. Benediktsson, “Spectral-spatial hyperspectral image classification with edge-preserving filtering,” *IEEE Trans. Geosci. Remote Sens.*, vol. 52, no. 5, pp. 2666–2677, May 2014.
- [14] Y. Tarabalka, M. Fauvel, J. Chanussot, and J. A. Benediktsson, “Svm- and mrf-based method for accurate classification of hyperspectral images,” *IEEE Geosci. Remote Sens. Lett.*, vol. 7, no. 4, pp. 736–740, Oct. 2010.
- [15] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “Imagenet classification with deep convolutional neural networks,” in *Proc. Adv. Neural Inf. Process. Syst.*, 2012, pp. 1097–1105.
- [16] Y. Chen, H. Jiang, C. Li, X. Jia, and P. Ghamisi, “Deep feature extraction and classification of hyperspectral images based on convolutional neural networks,” *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 10, pp. 6232–6251, Oct. 2016.
- [17] J. Zhu, L. Fang, and P. Ghamisi, “Deformable convolutional neural networks for hyperspectral image classification,” *IEEE Geosci. Remote Lett.*, vol. 15, no. 8, pp. 1254–1258, Aug. 2018.
- [18] F. Zhou, R. Hang, Q. Liu, and X. Yuan, “Hyperspectral image classification using spectral-spatial lstms,” *Neurocomputing*, to be published, 2019.
- [19] W. Song, S. Li, L. Fang, and T. Lu, “Hyperspectral image classification with deep feature fusion network,” *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 6, pp. 3173–3184, Jun. 2018.
- [20] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recog.*, 2016, pp. 770–778.
- [21] W. Li, G. Wu, F. Zhang, and Q. Du, “Hyperspectral image classification using deep pixel-pair features,” *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 2, pp. 844–853, Feb. 2017.
- [22] B. Liu, X. Yu, P. Zhang, A. Yu, Q. Fu, and X. Wei, “Supervised deep feature extraction for hyperspectral image classification,” *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 4, pp. 1909–1921, Apr. 2018.
- [23] Q. Liu, R. Hang, H. Song, and Z. Li, “Learning multiscale deep features for high-resolution satellite image scene classification,” *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 1, pp. 117–126, Jan. 2018.
- [24] K. Chatfield, K. Simonyan, A. Vedaldi, and A. Zisserman, “Return of the devil in the details: Delving deep into convolutional nets,” *arXiv preprint arXiv:1405.3531*, 2014.
- [25] O. Russakovsky, *et al.*, “Imagenet large scale visual recognition challenge,” *Int. J. Comput. Vis.*, vol. 115, no. 3, pp. 211–252, 2015.
- [26] M. Everingham, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman, “The pascal visual object classes (VOC) challenge,” *Int. J. Comput. Vis.*, vol. 88, no. 2, pp. 303–338, 2010.
- [27] C. Szegedy, *et al.*, “Going deeper with convolutions,” in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recog.*, 2015, pp. 1–9.
- [28] W. Li, S. Wang, and W. Kang, “Feature learning based deep supervised hashing with pairwise labels,” in *Proc. Int. Joint Conf. Artif. Intell.*, 2016, pp. 1711–1717.