



A Deep Network Based on Multiscale Spectral-Spatial Fusion for Hyperspectral Classification

Zhaokui Li^(✉), Lin Huang, Deyuan Zhang, Cuiwei Liu^(✉),
Yan Wang, and Xiangbin Shi

School of Computer Science, Shenyang Aerospace University, Shenyang,
Liaoning, People's Republic of China
{lzk, liucuiwei}@sau.edu.cn

Abstract. In this paper, we propose a deep network based on multiscale spectral-spatial fusion (MSS-Net) for Hyperspectral Image (HSI) classification. For the purpose of extracting better joint spectral-spatial features, the proposed network adopts multiscale spectral-spatial fusion method because different scale regions contain different spatial structure, texture features and more abundant neighborhood correlation which are helpful for classification. For every scale of input, we take the 3-D cubes from the raw data to the spatial and spectral learning module respectively. These two learning modules can extract the features with more abundant and original spectral-spatial correlation from the 3-D raw input data and then these features are combined as fusion spectral-spatial features. And we can get multiscale fusion spectral-spatial features which are fed to the two consequent residual learning block. Every residual block contains two 3-D convolutional layers and it can make full use of fusion features to learn more discriminative and high-level features. Furthermore, it also can help the network maintain higher accuracy when the network is deeper. After residual learning, multiscale fusion spectral-spatial features are concatenated and sent to fully convolutional layer for classification. The validation of our method is proved on three HSI data sets and the experimental results show that our method outperforms the other state-of-the-art methods.

Keywords: Hyperspectral Image classification · Multiscale Spectral-spatial fusion · Deep network

1 Introduction

Hyperspectral Image (HSI) has hundreds of continuous spectral bands and high spatial correlation, so it contains abundant spectral and spatial information which is useful for classification of different materials. Due to the high dimension of spectral, it causes the difficulties of classification and calculation. So the dimensionality reduction (DR) is required for HSI classification. Feature selection and feature extraction are the traditional methods to implement the DR [1]. Feature selection aims to find more discriminative bands to represent the entire image [2]. Compared with feature selection, feature extraction finds more useful features through mathematical transformation to

help learning of model. Currently, many deep learning models are proposed and they can learn more distinguished features with the goal of high classification accuracy [3–5]. The typical deep learning model stacked autoencoders (SAEs) can combine spatial and spectral features for HSI classification [6]. Deep convolutional neural network (CNN) [7, 8] is adopted to get the spatial feature and it has no requirement for the input dimensions. [9] proposes an end-to-end framework to learn the spectral and spatial features, but in this framework the input of spectral data is 1-D dimension. It misses the neighborhood information of spatial dimension and the classification accuracy will decrease when the network is deeper. [10] proposes a supervised spectral–spatial residual network and the residual block mitigates the decreasing-accuracy phenomenon, but this network firstly learns the spectral features which are as the input to extract the spatial information, so the spatial features are got from data which has been transformed and it misses the original spatial correlation.

To solve these problems and extract more discriminative spectral-spatial features, we propose a deep network based on multiscale spectral-spatial fusion (MSS-Net) for Hyperspectral Image classification. Because HSI has abundant spectral and spatial information which is most important for classification, so we consider different scale 3-D cubes from the raw data as the inputs to spatial learning and spectral learning module simultaneously. Then the features with the same scale are combined as fusion spectral-spatial features and these multiscale spectral-spatial features are fed to the two consequent residual learning block. The residual learning [11] can maintain a higher accuracy when the network is deeper and make the network more robustness. After average pooling, multiscale outputs as vectors are concatenated and adopted to the softmax layer to classify image.

The three major contributions of this paper includes: (1) multiscale spectral-spatial fusion is proposed to extract the fusion spectral-spatial features which contain different neighborhood correlation and low-level feature, such as spatial structure, texture feature. This advantage is more suitable for finding the discriminative features, (2) the input of spectral learning module is 3-D cube and it includes more abundant spectral-spatial correlation. The spatial learning is started from the raw image rather than the transformed data and it can extract more accurate and original spatial structure information, and (3) the spectral-spatial feature is combined at beginning, the fusion spectral-spatial feature as an input is fed to the convolutional layers and residual learning layers. The residual layers can make full use of fusion features to learn more high-level features for classification and it can help the network maintain a higher accuracy with deeper layers.

2 Proposed Framework

HSI data can be denoted as $\mathbf{R} \in \mathbb{R}^{M \times N \times L}$, $\mathbf{R}_i \in \mathbb{R}^{M \times N}$ is i th band image, M, N, L denote that the Hyperspectral Image has $M \times N$ pixels, L bands respectively. Multiscale 3-D data cubes as the inputs to do the same convolution operation which contains the spatial learning and spectral learning on these inputs respectively. After that, we fuse the same scale of spectral-spatial features as an input to the next layer. Then we use the two consequent residual blocks to get more discriminative features

which are learned from the fusion spectral-spatial features. The equation of residual learning is described as:

$$y = F(x, \{W\}) + x \quad (1)$$

Where x, y are the input and output cube respectively. The convolutional filters W need to be learned and the function F is the residual mapping of W . The function is defined as:

$$F = R(x) * W + b \quad (2)$$

Where b is the bias on the next layer of input x and W is convolutional kernels. R is the rectified linear unit activation function which sets elements with negative numbers to zero. In the residual learning, we use the batch normalization (BN) to regularize the learning process for every convolutional operation and BN is formulated as:

$$Y^n = R(y^{n-1} * W^n + b^n) \quad (3)$$

And Y^n represents the output of n th layer after BN operation, W^n, b^n mean the convolutional kernels and bias respectively on the n th layer. And the y^{n-1} is defined as:

$$y^{n-1} = \frac{Y^{n-1} - E(Y^{n-1})}{Var(Y^{n-1})} \quad (4)$$

Which Y^{n-1} is the output of $(n - 1)$ th layer after BN operation. After the residual block layer, the average pooling operation is done for the output of the residual block and we can get multiscale 1-D vectors which are concatenated and the equation of fusion is:

$$Y = g\{W_{last} * [y_{input1} \oplus y_{input2} \oplus \dots \oplus y_{inputi}] + b_{last}\} \quad (5)$$

$$g(x) = \max(0, x) \quad (6)$$

Where Y means the final output, \oplus is concatenating the outputs from the multiscale input, W_{last}, b_{last} denote the convolutional kernels and bias in the last layer respectively and y_{inputi} is the output of pooling from i th size of input. After concatenating the features, the softmax operation is done for the HSI classification.

For the architecture of MSS-Net, We take the Indian Pines Data Set as an example to describe our method in Fig. 1. Choosing the sample with size of $7 \times 7 \times 200$ as the input data. Through the convolutional operation of spatial learning and spectral learning module with the size of 128, 3×3 and 128, 1×1 respectively to get the features with size of $7 \times 7 \times 128$. The outputs of spectral-spatial features are fused to do the next convolutional and BN operation, finally it can generate the size of $7 \times 7 \times 128$ feature map and it is used as an input to the residual learning layer which contains two residual blocks. The residual block uses the size of 24, $3 \times 3 \times 128$ filters to extract features from the fusion spectral-spatial feature tensor, and it generates

the feature cube of $5 \times 5 \times 24$ and BN operation is used after every convolutional layer. Then the average pooling operation is conducted to transform the $5 \times 5 \times 24$ feature tensor into $1 \times 1 \times 24$ vector. Changing the scale of input data as $9 \times 9 \times 200$, $11 \times 11 \times 200$ and these inputs are done the same operation as the mentioned above. These multiscale vectors after pooling are concatenated as the final vector with the size of $1 \times 1 \times 72$ and sent to softmax layer for classification.

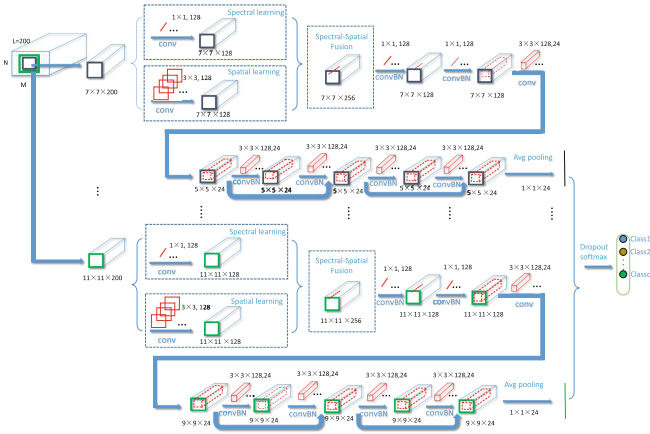


Fig. 1. Architecture of the proposed MSS-Net for hyperspectral classification. The size of multiscale inputs are $7 \times 7 \times 200$, $9 \times 9 \times 200$, $11 \times 11 \times 200$ and these three inputs are done the same operation. Finally the outputs of pooling from these three inputs are concatenated and used to classify the HSI image.

3 Experiments

3.1 Data Sets and Experimental Setting

We use three data sets which contain the Indian Pines Data Set, Pavia University Data Set and KSC Data Set to validate the effectiveness of our method. The Overall Accuracy (OA) and the Average Accuracy (AA) are used to evaluate the classification performance of all methods.

The Indian Pines Data Set (IN) has 16 vegetation classes and 224 bands with spatial size of 145×145 pixels and the spatial resolution is 20 m per pixel. Because of water absorption and noise, some bands are discarded and 200 bands are remained. The Pavia University Data Set (UP) has 610×340 pixels with a resolution of 1.3 m per pixel and 9 urban land-cover classes. The number of remaining bands is 103 after discarding the useless bands. The KSC Data Set contains 512×614 pixels with spatial resolution of 18 m per pixel and the ground-truth classes are 13. After removing the noise bands, 176 bands are retained and used for the experiment.

To reduce the random effects due to choosing different training samples every time, we do the experiment twenty times then choose the average effect as the final result. For the IN, we randomly choose 20%, 10%, 70% samples per class to form the training,

validation, testing set and the training iteration is set to 100. And in UP and KSC, the ratio is 5%, 10%, 85% and 10%, 10%, 80% respectively. In our implementation, the 50% dropout operation is adopted, the optimizer adopts the standard stochastic gradient descent method. The batch size is set to 64, the optimum learning rates in IN, UP, KSC data set are fixed as 0.0003, 0.0001 and 0.0001 respectively, and the momentum is set to 0.9.

3.2 Experimental Results

The proposed method is compared with state-of-the-art methods including the SVM [12], ResNet [11], SAE [4], 3-D CNN [4] and deep joint spectral-spatial CNN (Two-CNN) [9]. The SVM and ResNet is considered as a baseline in HSI classification. The framework of ResNet adopts the same residual blocks as our method and it does not contain the spectral-spatial learning module.

We change the size of input on the three data sets to find the more suitable multiscale for HSI classification. The results are showed in Table 1 and we can see the accuracy is higher with larger spatial size. The accuracy changes smaller when the size of the spatial domain larger than 11×11 on the three data sets and the accuracy is relatively higher when the size is 7×7 . For the purpose of selecting the relatively small size of input with higher classification accuracy, we choose the suitable size of the spatial domain 7×7 , 9×9 , 11×11 are as multiscale inputs to do the next experiment.

Table 1. Classification results (OA%) of our methods on the three data sets with different input size. (The percentage of training samples of IN, UP and KSC are 20%, 5%, 10% respectively.)

Spatial size	IN	UP	KSC
3×3	85.32	94.24	89.11
5×5	97.83	96.49	92.56
7×7	98.26	98.28	97.14
9×9	99.35	99.14	97.47
11×11	99.41	99.31	99.29
13×13	99.27	99.35	99.32
15×15	99.31	99.40	99.36

In order to validate the multiscale input is more beneficial for HSI classification than single input, we take some experiments in these three data sets. From Table 2, it illustrates the performance of multiscale inputs have obvious advantage than the single input when the percentage of training samples per class is 5% on the three data sets and it means multiscale inputs are more effective even if the training sample is limited. Because multiscale inputs can generate multiscale fusion spectral-spatial features and these features contain more abundant spectral-spatial correlation and spatial structure information than the single input. The classification accuracy of most situations are improving accuracy with increasing the training samples and the multiscale inputs outperform the single input. The results prove multiscale inputs can get more discriminative features and are more useful for HSI classification.

Table 2. Classification results (OA%) of our methods on the three data sets with different input size when the percentage of training samples is changing.

Data Set	Size of input	5%	7%	9%	10%
IN	7 × 7	86.47	92.02	93.76	95.43
	9 × 9	89.38	93.79	95.38	96.82
	11 × 11	92.66	95.05	96.17	97.26
	7 × 7, 9 × 9, 11 × 11	96.13	96.96	97.52	98.34
UP	7 × 7	98.28	99.11	99.25	99.57
	9 × 9	99.14	99.27	99.38	99.61
	11 × 11	99.31	99.36	99.43	99.67
	7 × 7, 9 × 9, 11 × 11	99.62	99.68	99.74	99.81
KSC	7 × 7	94.97	96.02	96.49	97.14
	9 × 9	95.72	96.29	97.33	97.47
	11 × 11	96.55	97.62	98.37	99.29
	7 × 7, 9 × 9, 11 × 11	97.83	98.31	99.16	99.54

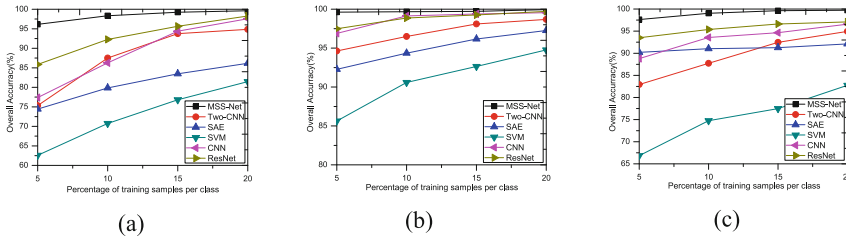


Fig. 2. The Overall Accuracy of changing the percentage of training samples by all methods on the three data sets. (a) Overall Accuracy on IN. (b) Overall Accuracy on UP. (c) Overall Accuracy on KSC.

Changing the number of training samples is considered to illustrate the impact of different number of samples to the classification accuracy. From Fig. 2(a), it shows the classification accuracy of each method in IN is improving when the percentage of training samples is increasing and it proves abundant labeled sample is helpful for HSI classification. MSS-Net has a great advantage than other methods in all cases. Compared with ResNet, MSS-Net is better than it and the result illustrates multiscale fusion spectral-spatial features contain more abundant neighborhood correlation and it can make the network learn more representative features. The Fig. 2(b) shows all methods’ performance in UP. MSS-Net is better than other methods when the percentage of training samples is 5%, and it proves our method constantly performs better than others with limited labeled samples. Because of the resolution of 1.3 m per pixel in UP, so the accuracy of MSS-Net, ResNet and CNN are closer when the percentage of training set is increasing. Figure 2(c) displays the results of methods on the KSC. Our method performs better than Two-CNN, it shows the spectral learning module in our method can extract more useful spectral-spatial correlation from the 3-D input cube than the 1-D input cube in Two-CNN.

Table 3. Classification accuracy (%) of different methods on the three data sets. (The percentage of training samples of IN, UP and KSC are 20%, 5%, 10% respectively.)

Data Set	IN		UP		KSC	
Metric	OA	AA	OA	AA	OA	AA
Two-CNN	94.86 ± 0.45	94.94 ± 0.29	94.63 ± 0.27	93.31 ± 0.22	87.74 ± 0.34	85.77 ± 0.25
SAE	86.16 ± 0.51	85.72 ± 0.38	92.27 ± 0.35	92.58 ± 0.29	91.04 ± 0.36	89.75 ± 0.28
SVM	81.41 ± 0.45	80.14 ± 0.56	85.63 ± 0.31	84.17 ± 0.34	74.76 ± 0.25	72.93 ± 0.41
CNN	97.58 ± 0.27	97.03 ± 0.32	96.89 ± 0.17	96.75 ± 0.25	93.58 ± 0.24	93.65 ± 0.37
ResNet	98.26 ± 0.37	97.31 ± 0.34	97.48 ± 0.20	97.03 ± 0.23	95.37 ± 0.26	94.86 ± 0.27
MSS-Net	99.62 ± 0.27	99.47 ± 0.36	99.62 ± 0.14	99.34 ± 0.19	99.54 ± 0.21	98.92 ± 0.28

Comparison results in Table 3 illustrate that our method has better performance than other state-of-the-art methods. From the Table 3, we can see the methods of deep learning have a great advantage than SVM and the accuracy is higher than the baseline. In these data sets, MSS-Net has an obvious advantage than Two-CNN, because multiscale spectral-spatial features contain more abundant neighborhood correlation than the single fusion spectral-spatial features in Two-CNN. And MSS-Net outperforms the ResNet, because our method learns more representative features through the spectral-spatial learning module rather than directly using the raw data to the residual learning block. The residual learning block can solve the problem of declining accuracy due to the deeper layers, so it makes our network has an excellent performance in deeper network.

4 Conclusions

A deep network based on multiscale spectral-spatial fusion (MSS-Net) for Hyperspectral Image classification is proposed in this paper. Multiscale spectral-spatial learning contains more abundant neighborhood correlation and spatial structure information which are beneficial for extracting more discriminative features and the effective of multiscale spectral-spatial fusion is proved on the experiments. The fusion spectral-spatial features are gotten from the 3-D raw input data and it means the following learning is based on the original correlation of spatial and spectral domain. The residual learning block makes this network more robustness and maintain a higher accuracy for the deeper layers. The experimental results show our method has better performance with limited labeled samples on the three data sets over other state-of-the-art methods.

Acknowledgment. This work was supported in part by the Natural Science Foundation of China (NSFC) under Grant No. 61602320, and the Natural Science Foundation of Liaoning under Grant Nos. 201601180 and 201601172.

References

1. Jia, X., Kuo, B.-C., Crawford, M.M.: Feature mining for hyperspectral image classification. *Proc. IEEE* **101**(3), 676–697 (2013)
2. Fang, X., et al.: Locality and similarity preserving embedding for feature selection. *Neurocomputing* **128**, 304–315 (2014)
3. Li, W., Wu, G., Zhang, F., Du, Q.: Hyperspectral image classification using deep pixel-pair features. *IEEE Trans. Geosci. Remote Sens.* **55**(2), 844–853 (2017)
4. Chen, Y., Jiang, H., Li, C., Jia, X., Ghamisi, P.: Deep feature extraction and classification of hyperspectral images based on convolutional neural networks. *IEEE Trans. Geosci. Remote Sens.* **54**(10), 6232–6251 (2016)
5. Zhao, W., Du, S.: Spectral–spatial feature extraction for hyperspectral image classification: a dimension reduction and deep learning approach. *IEEE Trans. Geosci. Remote Sens.* **54**(8), 4544–4554 (2016)
6. Tao, C., Pan, H., Li, Y., Zou, Z.: Unsupervised spectral–spatial feature learning with stacked sparse autoencoder for hyperspectral imagery classification. *IEEE Geosci. Remote Sens. Lett.* **12**(12), 2438–2442 (2015)
7. Szegedy, C., et al.: Going deeper with convolutions. In: *CVPR* (2015)
8. Li, Y., Zhang, H., Shen, Q.: Spectral–spatial classification of hyperspectral imagery with 3D convolutional neural network. *Remote Sens.* **9**(1), 67 (2017)
9. Yang, J., Zhao, Y.Q., Chan, C.W.: Learning and transferring deep joint spectral-spatial features for hyperspectral classification. *IEEE Trans. Geosci. Remote Sens.* **55**(8), 1–14 (2017)
10. Zhong, Z, Li, J., Luo, Z., et al.: Spectral-spatial residual network for hyperspectral image classification: a 3-d deep learning framework. *IEEE Trans. Geosci. Remote Sens.*, 1–12 (2017)
11. He, K., Zhang, X., Ren, S., et al.: Deep residual learning for image recognition, pp. 770–778 (2015)
12. Waske, B., van der Linden, S., Benediktsson, J.A., Rabe, A., Hostert, P.: Sensitivity of support vector machines to random feature selection in classification of hyperspectral data. *IEEE Trans. Geosci. Remote Sens.* **48**(7), 2880–2889 (2010)