

# The Classification of Hyperspectral Images Based on Band-Grouping and Convolutional Neural Network

---

**Guangyuan Fu<sup>1</sup>**

*Xi'an Hi-Tech Institute*

*Xi'an 710025, China*

*E-mail: dr-f@21cn1.c*

**Hongyang Gu**

*Xi'an Hi-Tech Institute*

*Xi'an 710025, China*

*E-mail: guhongyang@yeah.net*

For the classification of hyperspectral images, a classification algorithm based on band grouping and three-dimensional convolutional neural network(3D-CNN-BG) is proposed. The algorithm uses the correlation matrix of hyperspectral images to determine the similarity of bands, and the high similarity bands are grouped together. Then, every bands group is extracted spatial-spectral feature using 3D convolutional neural network. Finally, the high-level feature of every 3D-CNN is stacked together trained by the classifier. With band regroup, the feature in the similar group can be extracted fully by the 3D-CNN, which is good for classification. The experiments showed that compared with other hyperspectral images classification algorithms based on CNN, the proposed algorithm converges more faster, has less training parameters, and acquires higher classification accuracy. After training 100 times, the overall classification accuracy on the indian pines data set can reach 97.42%, increased by approximately 2% ~ 5%, which shows the strong practicability of proposed algorithm in hyperspectral application.

*CENet2017*

*22-23 July 2017*

*Shanghai, China*

---

<sup>1</sup>Speaker

## 1.Introduction

Over the past decade, hyperspectral imagery (HSI) has played a very important role in all areas of remote sensing. With the continuous development of imaging technology, the wealth of spectral information provided by HSI brings great opportunity to the distinction between the objects, but also brings huge challenge to the high-precision classification [1]. At present, the existing methods can be divided into two main categories: spectral classification and joint spectral-spatial classification. Compared with the spectral classification, the combination of spatial information and spectral information is considered to be more advantageous for the classification of HSI [2][3]. Therefore, in recent years, the joint spectral-spatial classification has become the development trend of hyperspectral image classification.

Nowadays, there are two main ways to use spatial features: spatial feature extraction and the post-processing after classification. The former uses the joint spatial characteristics and spectral characteristics to train the classifier for hyperspectral image classification [4][5][6][7]. The latter makes use of the spectral characteristics to obtain the classification results firstly, then the noise in the classification result is eliminated based on the spatial regularity [8][9][10]. The method in the spatial feature extraction is mechanical. The feature extracted is effective for some data sets, but maybe invalid in the other data sets. The adaptive ability of obtained characteristic is poor. The image segmentation is often utilized in the post-processing, but this method always depends on the previous classification results. Therefore, the key in the joint spectral-spatial classification is the extraction feature.

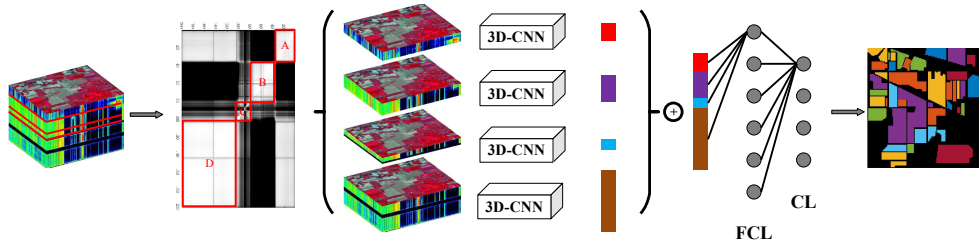
With the development of deep learning, more and more people apply it to spectral-spatial feature extraction of hyperspectral images. For example, convolution neural networks (CNN) [11][12][13][14], Automatic coding machine(AE) [15][16], depth belief network(DBN)[17][18]. Both AE and DBN deal with one-dimensional data. Even if the joint spectral-spatial classification is used, the spatial domain information is transformed into a one-dimensional vector which is combined with the spectral vector to obtain the spectral-spatial features. Chen et al. firstly applied 3D-CNN to classify hyperspectral image [14]. To some extent, the 3D-CNN is more suitable for the joint spectral-spatial classification than other deep learning models, because the joint spectral-spatial data in the hyperspectral images can be seen as the three tensor which is exactly suitable for the input of 3D-CNN. Although 3D-CNN achieved good results, the design of the network is too complex with so many parameters. So how to simplify the complexity of 3D-CNN is the key problem. In the hyperspectral images, the degree of similarity between the bands is relatively high, which brings some difficulty to the classification [19]. Ran proposed a band-grouping 1D-CNN(1D-CNN-BG) for hyperspectral image spectral classification [20]. Before training by the 1D-CNN, the bands in the hyperspectral images is grouped by the similarity.

Enlightened by the above literatures, in this paper, we extend the band-grouping into the 3D-CNN(3D-CNN-BG). According to the similarity between the bands, the bands with high degree of similarity are grouped together. Then the feature extraction is performed on the every bands group using 3D-CNN, and the high-level abstract feature vectors are combined together, which is trained by a fully connected layer and a classification layer. Compared with the 3D-CNN [14], the training parameters required in this paper are greatly reduced. The network structure designed in this paper can make full use of 3D-CNN to excavate similar characteristics in the

similar bands. The experiments with the typical data set proved better results in this paper than other types of CNN algorithm, both in terms of algorithm convergence speed and classification accuracy.

## 2. Proposed Method

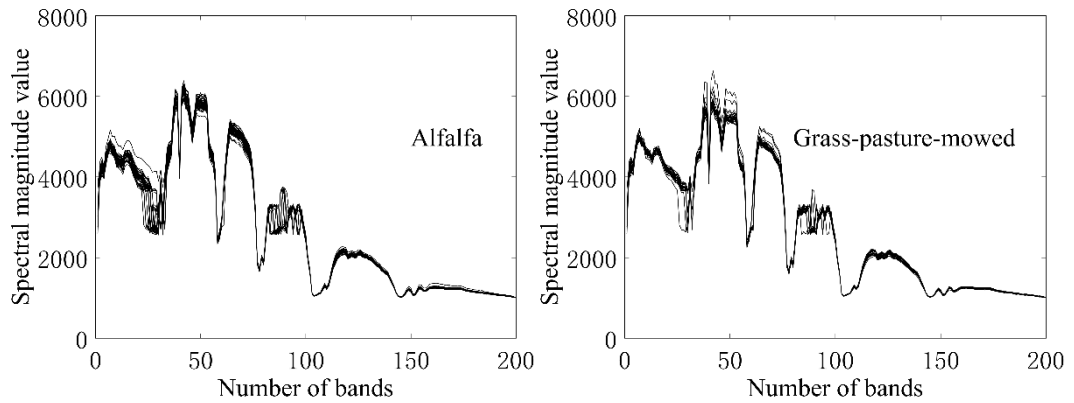
Compared with the 2D-CNN, the 3D-CNN is more suitable for 3D cube data mining. Through the 3D convolution operation, it can fully integrate the 3D data and extract more effective feature information. The classification framework proposed in this paper is shown in figure 1. We can see from figure 1 that the proposed classification framework mainly includes three parts: band regroup, 3D-CNN feature extraction and network training.



**Figure 1:** The classification framework in this paper

### 2.1 Regroup the Bands based on the Similarity between Bands

In the hyperspectral images, some spectral characteristic curves can be distinguished easily, but some are so similar that cannot be distinguished. The spectral characteristic curves of Alfalfa and Grass-pasture-mowed two kinds of objects in the Indian Pines data set [21] as shown in figure 2. It can be observed from the figure that both the spectral characteristic curves are so similar that it is difficult to the classification. Therefore, it is necessary to regroup the similar bands, and then to extract the effective features of the similar bands.



**Figure 2:** Spectrum of Alfalfa and Grass-pasture-mowed

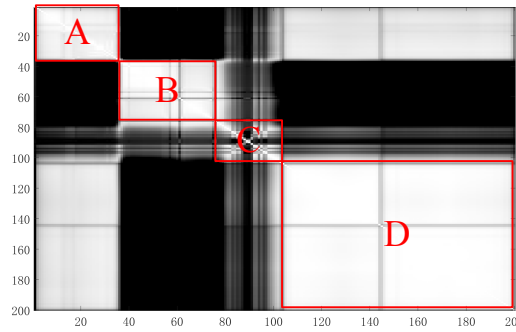
The band-grouping algorithm is based on the spectral correlation matrix. The hyperspectral dataset can be regarded as a 3D tensor  $\mathbf{X} \in R^{m \times n \times b}$ , and  $x_{ijk}$  represents the corresponding value in the spatial position  $(i, j)$  of the  $k$ th band in the  $\mathbf{X}$ . The correlation coefficient between the  $k_1$ th and  $k_2$ th band,

$$c(k_1, k_2) = \frac{\sigma(k_1, k_2)}{\sigma(k_1, k_1)\sigma(k_2, k_2)} \quad (1.1)$$

Here the  $\sigma(k_1, k_2)$  represents the covariance of the  $k_1$ th and  $k_2$ th band,

$$\sigma(k_1, k_2) = \sum_{i=1, j=1}^{i=m, j=n} (x_{ijk_1} - x_{..k_1}^-)(x_{ijk_2} - x_{..k_2}^-) \quad (1.2)$$

$x_{..k_1}^-$  is the average of all pixels in the  $k_1$ th band. The correlation matrix of the Indian Pines dataset is showed as the figure 3, and where the light color indicates a higher degree of correlation. Neighbouring bands with a correlation coefficient difference smaller than a threshold  $\tau$  are grouped together [20], and in this paper  $\tau=0.7$ . According to the degree of correlation of the 200 bands, they can be divided into four band groups: 1~35, 36~77, 78~102, 103~200.



**Figure 3:** Correlation Matrix of Indian Pines

In general, before training network it is necessary for data preprocessing. There are many common preprocessing methods including normalizing, standardizing and principal component analysis (PCA) whitening [22]. Compared with the other two ways, whitening reduces the hyperspectral data correlation between each band to a certain extent, and can make the training speed of the network faster and more stable. So the band group is whitened respectively. A band group can be denoted  $\Psi \in R^{n \times d}$  by ( $d$  represents the number of band in group). The eigenvalues  $\lambda_1, \lambda_2, \dots, \lambda_d (\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_d)$  and the corresponding eigenvectors  $\omega_1, \omega_2, \dots, \omega_d$  can be obtained by the PCA. Let,

$$\Lambda = \text{diag}[\lambda_1, \lambda_2, \dots, \lambda_d]$$

$$Y = \text{diag}[\omega_1, \omega_2, \dots, \omega_d]$$

then,

$$\Psi_{\text{whitening}} = Y \Lambda^{-1/2} (\Psi - \bar{\Psi}) \quad (1.3)$$

Where,  $\bar{\Psi} = 1/n \sum_{i=1}^n \Psi_i$ ,  $\Psi_i$  represents the  $i$ th column vector of  $\Psi$ .

## 2.2 The configuration of 3D-CNN

After the regrouping the bands, the original hyperspectral dataset can be divided into  $g$  band groups,  $\Psi_k \in R^{m \times n \times b_k}$ ,  $k=1, 2, \dots, g$ . Taking into account the spectral-spatial information of the hyperspectral images, so not only the spectral characteristic curve of the target pixel is used to be trained to classify, but also the spatial information. Generally taking a

size  $w$  of the neighborhood space, the target object can be seen as a 3D tensor,  $x_{(i,j)}^k \in R^{w \times w \times b_k}$ . In this paper, we also utilize the 3D-CNN to train these 3D tensor to obtain the classification results.

The traditional CNN is mainly composed of a series of convolution layers, pooling layers, full connection layers and classification layer [23]. The only difference between 2D-CNN and 3D-CNN is that the convolution kernel is a 3D tensor. In this paper, the network consists of 2 convolution layers, 2 pooling layers, 1 full connection layers and 1 classification layer.

For convolution layer, the number of feature maps, the size of the convolution kernel and the selection of the activation function are very important. The number of feature maps is choose to ensure increased layer by layer, and finally the length of flattened vector is no more than 2 times the number of neurons in full connection layer. Because the size of neighborhood space  $w$  is not too large, it means that the spatial domain size ( $ker_1 \times ker_2$ ) in the convolution kernel ( $ker_1 \times ker_2 \times ker_3$ ) can not be selected is too large. So in this paper, we choose  $ker_1 = ker_2 = 2$  or 3, and the choice of  $ker_3$  is decided according to the band number of band group. To the activate function, the sparsity of the "relu" function is more consistent with the neuroscience research. With "relu" function, the speed of the training network is faster than other activation functions, such as "sigmoid", "tanh" function. Therefore, "relu" function is choosed in this paper.

For the pooling layer, the way of pooling has great influence on network training. The common pooling methods include average pooling and max pooling. The two methods have little difference in the final effect, but the training speed of the network with max pooling is faster, so in this paper we choose max pooling as the pooling method. The sampling interval in three dimensions is 2.

### 2.3 Train the 3D-CNN

The high-level abstract features extracted by 3D-CNN flattened to get vectors. Then these vectors are stacked together to be trained by the full connection layer and the classification layer to obtain the classification results.

For the fully connection layer, in accordance with CHEN's 3D-CNN[14] settings, the "sigmoid" activation function is chose with 100 hidden layer neurons.

For the classification layer, hyperspectral classification is a multi-classification problem, so choosing the softmax classification layer is reasonabl. Then the output can be seen as the probability in different categories of a pixel.

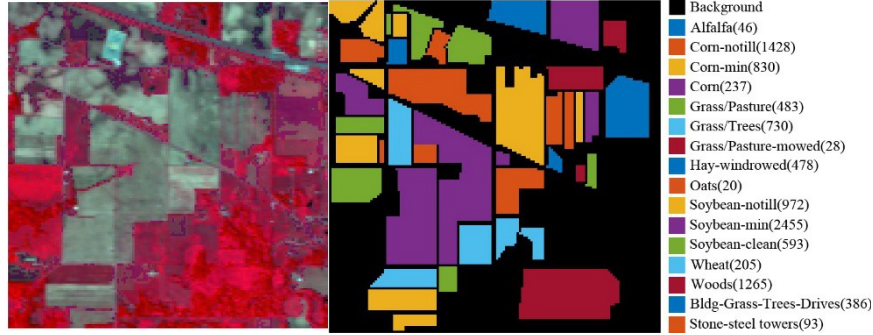
For network training, we choose Nadam algorithm [24] improved on the Adam algorithm [25] to train the network. Compared with the SGD algorithm, Nadam algorithm converges faster, and obtains higher accuracy. In general, if you do not use the regularization, the network will tend to fall into the overfitting. So we introduce dropout mechanism [14] into the network.

## 3.Experiments

### 3.1 Description of the data set

The data sets used in this experiment are Indian Pines hyperspectral data set and Pavia University hyperspectral data set.

Indian Pines hyperspectral data set is taken by AVIRIS in Indiana northwest of India pine forest test. The spatial size of the data is  $145 \times 145$ , with a spatial resolution of about 20m. There are remaining 200 bands available in this data set, with the removal of the water absorption bands (104~108, 150~163, and 220). The data set contains 16 kinds of known objects of a total of 10366 samples. Figure 4 shows the false color map and the ground truth of the data set. Because of the high similarity of the spectral characteristic in this data set, it brings great challenges to the accurate classification of objects. So it is widely used for testing the performance of hyperspectral image classification algorithm.



**Figure 4:** Correlation Matrix of Indian Pines

In order to test the performance of the proposed algorithm, other algorithms including spectral classification and joint spectral-spatial classification are tested on the same data set. In this paper, we mainly modify the CNN, so we will compare our algorithm with other CNN algorithm, including the spectral classification [20] and the joint spectral-spatial classification 3D-CNN [14]. 3D-CNN and our algorithm are set as the same network configuration. All the algorithm will be iterated 100 times to test the speed of convergence.

### 3.2 Results

For the Indian Pines data set, the band groups have been given in the section second: 1~35, 36~77, 78~102, 103~200. 10% samples is chosen as the training sample randomly from every category, and the remaining 90% is the test sample. The architecture of the network is shown in table 1, the network has a total of 6 layers, including 2 convolution layers, 2 pooling layers, 1 full connection layer and 1 classification layer.

No.	Type	Convolutional kernel	Active funcion	Pooling	Dropout
1	Conv+Pooling	6@2×2×3	Relu	Maxpooling	\
2	Conv+Pooling	16@2×2×3	Relu	Maxpooling	50%
3	Full	100	sigmoid	\	50%
4	Classification	16	softmax	\	\

**Table 1:** Architecture of 3D-CNN

The classification accuracy of every algorithm is shown in table 2. The classification accuracy of our proposed algorithm is the highest over majority categories, increased by about 2% than 3D-CNN. Figure 5 shows the classification maps obtained by these three algorithm. The overall classification accuracy of 1D-CNN-BG is relatively low, because in the experiment, every algorithm is iterated only 100 times. But the classification accuracy of the proposed 3D-

CNN-BG can achieve a higher one in the iteration for only the 100 times, which shows that compared with other algorithms, our proposed algorithm owns the faster convergence rate.

Method	1D-CNN-BG	3D-CNN	3D-CNN-BG
OA(%)	66.44(0.42)	95.03(1.01)	<b>97.42(0.35)</b>
AA(%)	56.30(1.73)	93.19(1.24)	<b>94.86(0.32)</b>
$kappa \times 100$	61.21(0.44)	95.25(0.24)	<b>97.06(0.21)</b>
Alfalfa	20.65(3.26)	90.22(6.24)	<b>91.30(3.83)</b>
Corn-notill	56.07(3.21)	93.86(1.31)	<b>96.67(0.67)</b>
Corn-mintill	40.72(2.53)	95.24(1.48)	<b>95.69(1.05)</b>
Corn	26.27(4.17)	93.46(3.61)	<b>95.15(1.73)</b>
Grass-pasture	70.86(6.38)	<b>96.43(1.28)</b>	95.34(1.84)
Grass-trees	92.95(2.07)	<b>99.42(0.11)</b>	99.08(0.31)
Grass-pasture-mowed	22.32(7.73)	90.18(15.02)	<b>95.54(5.85)</b>
Hay-windrowed	98.01(0.31)	<b>100.00(0.00)</b>	99.95(0.09)
Oats	17.50(5.59)	62.50(10.31)	<b>63.75(3.83)</b>
Soybean-notill	48.97(1.21)	93.29(0.54)	<b>95.96(1.07)</b>
Soybean-mintill	74.29(2.82)	96.22(0.91)	<b>97.92(0.50)</b>
Soybean-clean	42.88(3.50)	92.66(1.41)	<b>96.63(0.84)</b>
Wheat	87.68(2.35)	<b>99.63(0.40)</b>	98.66(1.80)
Woods	91.11(1.21)	98.99(0.85)	<b>99.31(0.48)</b>
Buildings-Grass-Trees-Drives	37.89(4.66)	92.23(3.13)	<b>99.16(0.81)</b>
Stone-Steel-Towers	72.58(10.49)	96.77(1.52)	<b>97.58(0.89)</b>

**Table 2:** The classification accuracy of every algorithm after 100 time iteration



(a)1D-CNN-BG

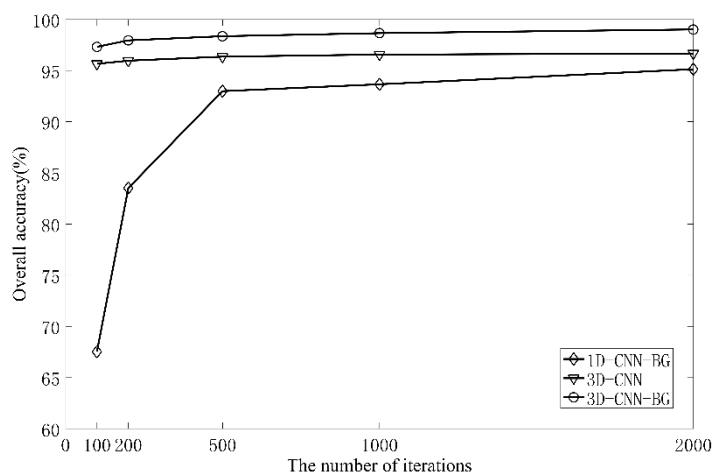
(b)3D-CNN

(c)3D-CNN-BG

**Figure 5:** The best classification results of every algorithm after 100 time iteration

In order to demonstrate the fast convergence of the proposed algorithm from the experimental point of view, the overall classification accuracy of every algorithm under different iteration times is calculated respectively. Figure 6 shows the relationship between the overall classification accuracy of every algorithm and the different iteration times. As can be seen from the figure, with the increase in the number of iterations, the overall classification accuracy of the proposed algorithm is always the highest. When the time of iteration is 2000, the overall classification accuracy can reach 98.79%, while the 3D-CNN and 1D-CNN-BG only can reach 97.93% and 95.35% respectively. Table 3 shows the classification accuracy of every

algorithm after 2000 time iteration, and figure 7 shows the classification maps after 2000 time iteration.



**Figure 6 :** The relationship between OA and the number of iteration

Method	1D-CNN-BG	3D-CNN	3D-CNN-BG
OA(%)	95.35	97.79	98.79
AA(%)	93.79	96.53	98.66
$kappa \times 100$	94.64	97.35	98.99

**Table 3 :** The classification accuracy of every algorithm after 2000 time iteration



**Figure 7 :** The classification results of every algorithm after 2000 time iteration

#### 4. Conclusion

According to the existing CNN algorithms for hyperspectral image classification problems, in this paper we propose a classification algorithm for hyperspectral image based on 3D-CNN combined with the band-groups. The bands are regrouped by the similarity between bands, then these groups will be extracted feature by the 3D-CNN. From the experiment on the indian pines data set, the proposed algorithm can converge more faster, and the parameters in our algorithm are less than the parameters need in the Chen's 3D-CNN. After only iteration 100 time, the OA of our algorithm can reach 97.42%, increased by almost 2.5% compared 3D-CNN.

#### References

- [1] Z. M. Li, J. Zhang, H. Huang and T. Jiang, "Semi-Supervised Bundle Manifold Learning for Hyperspectral Image Classification," *optics & Precision Engineering*, vol. 23, n. 5, 2015, pp. 1434-1442.



- [2] G. Camps-Valls, D. Tuia, L. Bruzzone and J. A. Benediktsson, "Advances in Hyperspectral Image Classification: Earth Monitoring with Statistical Learning Methods," *IEEE Signal Processing Magazine*, vol. 31, n. 1, 2013, pp. 45-54.
- [3] M. Fauvel, Y. Tarabalka, J. A. Benediktsson and J. Chanussot, "Advances in Spectral-Spatial Classification of Hyperspectral Images," *Proceedings of the IEEE*, vol. 101, n. 3, 2013, pp. 652-675.
- [4] H. Huang and X. L. Zheng, "Hyperspectral Image Classification with Combination of Weighted Spatial-Spectral and KNN," *optics & Precision Engineering*, vol. 24, n. 4, 2016, pp. 873-881.
- [5] Z. Q. Tang, G. Y. Fu, J. Chen and L. Zhang, "Multiscale Segmentation-Based Sparse Coding for Hyperspectral Image Classification," *optics & Precision Engineering*, vol. 23, n. 9, 2015, pp. 2708-2714.
- [6] P. Gurram and H. Kwon, "Contextual Svm Using Hilbert Space Embedding for Hyperspectral Classification," *IEEE Geoscience & Remote Sensing Letters*, vol. 10, n. 5, 2013, pp. 1031-1035.
- [7] M. Fauvel, J. A. Benediktsson, J. Chanussot and J. R. Sveinsson, "Spectral and Spatial Classification of Hyperspectral Data Using Svms and Morphological Profiles," *IEEE Transactions on Geoscience & Remote Sensing*, vol. 46, n. 11, 2008, pp. 3804-3814.
- [8] Y. Tarabalka, M. Fauvel, J. Chanussot and J. A. Benediktsson, "SVM- and MRF-Based Method for Accurate Classification of Hyperspectral Images," *IEEE Geoscience & Remote Sensing Letters*, vol. 7, n. 4, 2010, pp. 736-740.
- [9] Y. Tarabalka, J. A. Benediktsson and J. Chanussot, "Spectral-Spatial Classification of Hyperspectral Imagery Based on Partitional Clustering Techniques," *IEEE Transactions on Geoscience & Remote Sensing*, vol. 47, n. 8, 2009, pp. 2973-2987.
- [10] J. Bai, S. Xiang and C. Pan, "A Graph-Based Classification Method for Hyperspectral Images," *IEEE Transactions on Geoscience & Remote Sensing*, vol. 51, n. 2, 2013, pp. 803-817.
- [11] W. Hu, Y. Huang, L. Wei, F. Zhang and H. Li, "Deep Convolutional Neural Networks for Hyperspectral Image Classification," *Journal of Sensors*, vol. 2015, n. 2, 2015, pp. 1-12.
- [12] W. Zhao, Z. Guo, J. Yue, X. Zhang and L. Luo, "On Combining Multiscale Deep Learning Features for the Classification of Hyperspectral Remote Sensing Imagery," *International Journal of Remote Sensing*, vol. 36, n. 13, 2015, pp. 3368-3379.
- [13] H. Liang and Q. Li, "Hyperspectral Imagery Classification Using Sparse Representations of Convolutional Neural Network Features," *Remote Sensing*, vol. 8, n. 2, 2016, pp. 99.
- [14] Y. Chen, H. Jiang, C. Li and X. Jia, "Deep Feature Extraction and Classification of Hyperspectral Images Based on Convolutional Neural Networks," *IEEE Transactions on Geoscience & Remote Sensing*, vol. 54, n. 10, 2016, pp. 1-20.
- [15] 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 & Remote Sensing*, vol. 7, n. 6, 2014, pp. 2094-2107.
- [16] C. Xing, L. Ma and X. Yang, "Stacked Denoise Autoencoder Based Feature Extraction and Classification for Hyperspectral Images," *Journal of Sensors*, vol. 2016, 2016, pp. 1-10.
- [17] 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 & Remote Sensing*, vol. 8, n. 6, 2015, pp. 1-12.
- [18] T. Li, J. Zhang and Y. Zhang, "Classification of Hyperspectral Image Based on Deep Belief Networks," *Proc. IEEE International Conference on Image Processing*, 2015, pp. 5132-5136.

- [19] Y. Li, W. Xie and H. Li, "*Hyperspectral Image Reconstruction by Deep Convolutional Neural Network for Classification*," *Pattern Recognition*, vol. 63, 2016, pp. 371-383.
- [20] L. Ran, Y. Zhang, W. Wei and T. Yang, "*Bands Sensitive Convolutional Network for Hyperspectral Image Classification*," *Proc. International Conference on Internet Multimedia Computing and Service*, 2016, pp. 268-272.
- [21] AVIRIS data, <http://engineering.purdue.edu/~biehl/MultiSpec/hyperspectral.html>.
- [22] S. Chen and H. Yin, "*Novel feature compensation method for face recognition based on whitening PCA image reconstruction*," *Application Research of Computers*, vol. 32, n. 9, 2015, pp. 2853-2856.
- [23] A. Krizhevsky, I. Sutskever and G. E. Hinton, "*Imagenet Classification with Deep Convolutional Neural Networks*," *Proc. International Conference on Neural Information Processing Systems*, 2012, pp. 1097-1105.
- [24] T. Dozat, "*Incorporation nesterov momentum into Adam*," [http://cs229.stanford.edu/proj2015/054\\_report.pdf](http://cs229.stanford.edu/proj2015/054_report.pdf).
- [25] D. P. Kingma and J. Ba, "*Adam: A Method for Stochastic Optimization*," *Computer Science*, 2014.