Hyperspectral image classification using spectral-spatial hypergraph convolution neural network

Ma, Zhongtian, Jiang, Zhiguo, Zhang, Haopeng

Zhongtian Ma, Zhiguo Jiang, Haopeng Zhang, "Hyperspectral image classification using spectral-spatial hypergraph convolution neural network," Proc. SPIE 11862, Image and Signal Processing for Remote Sensing XXVII, 118620I (12 September 2021); doi: 10.1117/12.2599787

SPIE.

Event: SPIE Remote Sensing, 2021, Online Only

Hyperspectral image classification using spectral-spatial hypergraph convolution neural network

Zhongtian Ma[a,b], Zhiguo Jiang[a,b], and Haopeng Zhang*[a,b]

[a]Department of Aerospace Information Engineering, School of Astronautics, Beihang University, Beijing 102206, China
[b]Beijing Key Laboratory of Digital Media, Beijing 102206, China

ABSTRACT

Deep learning methods, especially convolutional neural networks(CNN), have been widely used in hyperspectral image(HSI) classification. Recently, graph convolutional networks (GCN) have shown great potential in HSI classification problem. However, the existing GCN-based methods have several problems. First, the existing methods rely too much on the adjacency matrix, which cannot be changed during training. Furthermore, most of them can only use a single kind of feature, and fail to extract the spectral-spatial information from the HSI. Finally, for the existing GCN-based methods, it is difficult to achieve the same accuracy as the mature CNN methods. In this paper, we propose a spectral-spatial hypergraph convolutional neural network (S^2HCN) for HSI classification. Compared with the existing GCN-based methods, S^2HCN has the following advantages. Different from the adjacency matrix that is fixed during training of GCN, S^2HCN can dynamically update the weight of the hyperedge during training, which reduces the reliance on prior information to a certain extent. In addition, S^2HCN generates hyperedges from the spectral and spatial features independently, and adopts the incidence matrix composed of all hyperedges to replace the adjacency matrix in GCN. In this way, the spectral and spatial features can be better integrated. Finally, compared to a simple graph structure, the hypergraph structure can express the high-dimensional relationships in the data, which is beneficial to classification problems. Sufficient experiments on two popular HSI datasets have proved the effectiveness of S^2HCN.

Keywords: Graph convolution networks, hypergraph learning, hyperspectral image (HSI) classification, feature fusion, deep learning.

1. INTRODUCTION

Hyperspectral imaging refers to an imaging technique that continuously samples the entire electromagnetic spectrum, and the obtained hyperspectral images(HSIs) include hundreds of spectral bands. Nowadays, HSIs are widely used in biomedical imaging, agriculture and astronomy.[1]

Due to the abundant information, HSIs have outstanding performance in remote sensing land-cover classification tasks. Hyperspectral classification refers to the pixel-level classification in HSIs. Hyperspectral classification has roughly passed through two stages of traditional machine learning methods and deep learning methods. In the traditional machine learning stage, researchers mainly focus on feature extraction and classifier selection. In the research on feature extraction, feature dimensionality reduction and spatial-spectral feature fusion are the two main research issues. Feature dimensionality reduction is used to deal with the Hughes phenomenon[2] caused by the redundancy of spectral information. Rodarmel *et al.*[3] first used principal component analysis(PCA) to reduce the dimensionality of the spectral band, and then PCA became a common preprocessing method. The method of manifold learning is also used to reduce the dimensionality of hyperspectral image features, such as[4] ,[5] . Spatial-spectral feature fusion simultaneously utilizes the spatial and spectral information of HSIs, which can effectively improve the classification performance. Rajadell *et al.*[6] proposed a spectral-spatial pixel characterization method that utilize Gabor filters to extract texture features. Fauvel *et al.*[7] use morphological methods to fuse spatial and spectral features. Classifier selection is another focus of traditional machine learning

Further author information: (Send correspondence to Haopeng Zhang)
Haopeng Zhang: E-mail: zhanghaopeng@buaa.edu.cn, Telephone: +86 10 6171 6978

Image and Signal Processing for Remote Sensing XXVII, edited by Lorenzo Bruzzone, Francesca Bovolo, Proc. of SPIE Vol. 11862, 118620I · © 2021 SPIE
CCC code: 0277-786X/21/$21 · doi: 10.1117/12.2599787

methods. In the actual design of the algorithm, classifier selection is often combined with feature extraction methods. Many machine learning classification algorithms are applied to HSI classification, such as support vector machine(SVM)[8] ,[9] , decision tree[10] , etc.

After the development of deep learning, the field of HSI classification has also moved to the second stage. In Hu *et al.*'s work[11] , convolutional neural network(CNN) is empolyed to perform HSI classification in the spectral domain. Chen *et al.*[12] thoroughly explored the application of CNN in HSI classification, and proposed 3D-CNN for HSI classification with spatial-spectral feature, which has a profound impact on subsequent research. Unlike traditional machine learning methods that use the two-step process of first extracting features and then selecting classifiers, most deep learning methods utilize end-to-end networks to directly obtain classification results. Most obviously, the CNN method cuts the entire HSI into small patches containing only tens of pixels, resulting in the loss of many non-local information.

Recently, deep learning on graph data has received rapid development and widespread attention. A novel graph convolutional neural network(GCN)[13] has also been tried for HSI classification. In the conventional machine learning stage of HSI classification, there exist methods that treat HSIs as graph data and use graph learning methods for classification. GCN is a graph representation learning algorithm that incorporates neural networks, and is an attempt of deep learning on graph data. Unlike the CNN-based method that cuts HSI into small patches containing tens of pixels, the GCN-based method performs semi-supervised classification of graph nodes on the entire HSI. Such an operation enables the GCN-based method to make full use of non-local information and pay attention to long-term dependencies. Qin *et al.*[14] first applied GCN to HSI classification, and specifically proposed a spatial-spectral GCN. Another representative work is Hong *et al.*[15] . They compared the classification performance of GCN and CNN, and proposed three strategies to combine GCN and CNN. Wan *et al.*[16] use superpixel segmentation as preprocessing, first segment the hyperspectral image into superpixels, and then use GCN for classification, which increases the efficiency of the method.

Nevertheless, the existing GCN method still has two shortcomings. One is that it relies too much on prior information to construct the graph structure, and the constructed graph structure cannot be changed during training. Second, the accuracy of the GCN-based method is slightly insufficient compared with the mature CNN method.

In this paper, we propose a spectral-spatial hypergraph convolutional neural network (S^{2}HCN) for HSI classification. The hypergraph convolutional neural network (HCN)[17,18] is an extension of the GCN on hypergraph structure. The proposed S^{2}HCN first extract the hypergraph structure from the HSI using the spectral and spatial features, and then input the extracted hypergraph structure and the original HSI into the HCN for training. Compared with the existing GCN-based methods, S^{2}HCN has the following advantages. Different from the adjacency matrix that is fixed during training of GCN, S^{2}HCN can dynamically update the weight of the hyperedge during training, which reduces the reliance on prior information to a certain extent. In addition, S^{2}HCN generates hyperedges from the spectral and spatial features independently, and adopts the incidence matrix composed of all hyperedges to replace the adjacency matrix in GCN. In this way, the spectral and spatial features can be better integrated. Finally, compared to a simple graph structure, the hypergraph structure can express the high-dimensional relationships in the data. As a more powerful model, the hypergraph can extract richer information, which is beneficial to classification problems.

2. METHODOLOGY

The overall architecture of S^{2}HCN is shown in Fig. 1. We first extract the spatial-spectral features from the original HSI, then construct the hypergraph structure, and finally send the HSI and the constructed hypergraph structure into the designed HCN for learning. In this section, we first introduce hypergraph convolution, and then present technical details of S^{2}HCN.

2.1 Hypergraph Convolution

The difference between a hypergraph and a graph is that the hyperedges of the hypergraph are degree-free, while the degree of the edges of the graph is fixed 2. That is to say, a hyperedge in a hypergraph can connect more than two nodes, while an edge in a graph can only connect two nodes. A hypergraph is denoted as $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathbf{W})$,

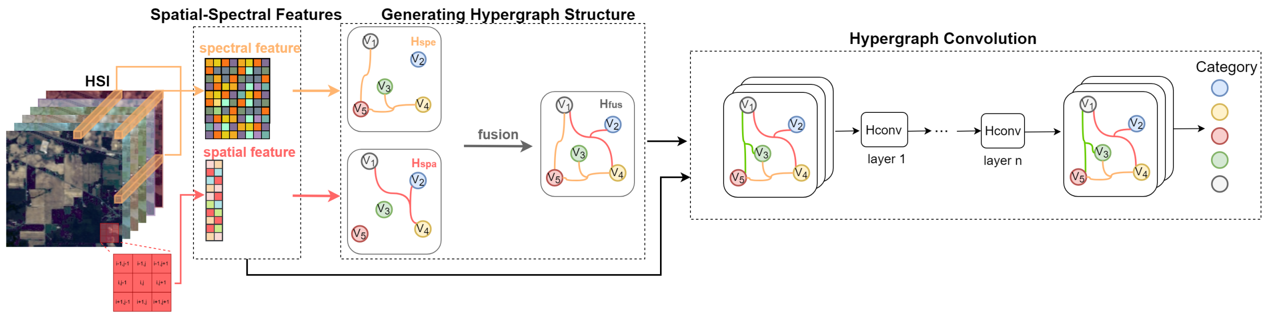


Figure 1. Overview of the proposed S^{2}HCN.

where $\mathcal{V}$ is the set of nodes, $\mathcal{E}$ is the set of hyperedges, and $\mathbf{W}$ is the hyperedge weight matrix. The incidence matrix $\mathbf{H}$ is usually used to represent the hypergraph, and its definition is as follows,

$$h(v, e) = \begin{cases} 1, & \text{if } v \in e \\ 0, & \text{if } v \notin e \end{cases} \tag{1}$$

The Laplacian matrix $\mathbf{L}$ can be calculated from $\mathbf{H}$ as follows,

$$\mathbf{L} = \mathbf{I} - \mathbf{D}_v^{-1/2}\mathbf{H}\mathbf{W}\mathbf{D}_e^{-1}\mathbf{H}^\top\mathbf{D}_v^{-1/2} \tag{2}$$

where $\mathbf{D}_v$ and $\mathbf{D}_e$ are the degree matrices of nodes and edges, respectively.

Hypergraph convolution is generalized from graph convolution. The definition of hypergraph convolution is given in Feng *et al.*[17] For a hypergraph data $\mathbf{X}^{n\times c_1}$ with n nodes and c channels, the hypergraph convolution acting on it can be expressed as,

$$\mathbf{Y} = \mathbf{D}_v^{-1/2}\mathbf{H}\mathbf{W}\mathbf{D}_e^{-1}\mathbf{H}^\top\mathbf{D}_v^{-1/2}\mathbf{X}\mathbf{\Theta} \tag{3}$$

where $\mathbf{Y}^{n\times c_2}$ is the output, $\mathbf{W} = diag(w_1, w_2, ..., w_n)$ and $\mathbf{\Theta}^{c_1\times c_2}$ are the trainable parameters.

Therefore, the hypergraph convolutional layer with the activation function can be expressed as,

$$\mathbf{X}^{(l+1)} = \sigma\left(\mathbf{D}_v^{-1/2}\mathbf{H}\mathbf{W}\mathbf{D}_e^{-1}\mathbf{H}^\top\mathbf{D}_v^{-1/2}\mathbf{X}^{(l)}\mathbf{\Theta}^{(l)}\right) \tag{4}$$

where l is the lth layer, σ is the activation function.

2.2 S^{2}HCN

In S^{2}HCN , we first extract spatial-spectral features, and then combine the two features to construct a hypergraph. For the spatial feature, we use the coordinates of the pixel as the feature, as follows,

$$\mathbf{X}_{spatial}[i] = [h(i), v(i)] \tag{5}$$

where h(i) and v(i) is the horizontal and vertical coordinates of pixel i. The spectral characteristics $\mathbf{X}_{spectral}$ can be obtained directly from the original HSI. After obtaining the two features, we respectively generate the corresponding hypergraph adjacency matrix $\mathbf{H}_{spectral}$ and $\mathbf{H}_{spatial}$ through Eq. 6.

$$h(i, j) = \begin{cases} e^{-\sigma\|\mathbf{x}_i - \mathbf{x}_j\|^2/\boldsymbol{mean}}, & \text{if } \mathbf{x}_i \in \mathcal{N}_k(\mathbf{x}_j) \\ 0, & \text{otherwise} \end{cases} \tag{6}$$

where σ is a hyperparameter, $\boldsymbol{mean}$ is the average value of the Euclidean distance of all nodes $v \in \mathcal{V}$.

Next, we splice the two incidence matrices $\mathbf{H}_{spectral}$ and $\mathbf{H}_{spatial}$ into a fused adjacency matrix $\mathbf{H}_{fusion}$. Then, we put the obtained incidence matrix $\mathbf{H}_{fusion}$ and the original HSI into the designed two-layer HCN for training.

3. EXPERIMENTS

In this section, we conducted a variety of experiments on two representative datasets, including comparison experiments of different methods, limited training samples classification experiments, and hyperparameter analysis experiments.

3.1 Datasets

(1) Indain Pines

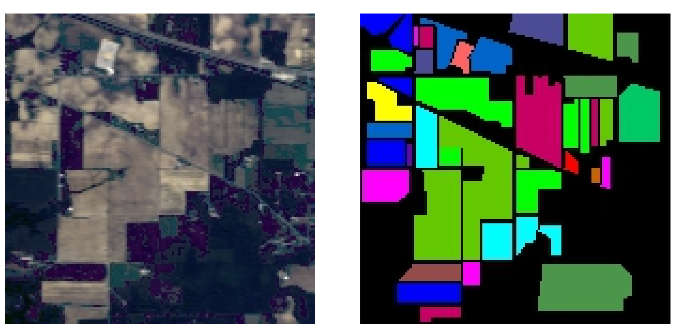
This dataset was photographed in northwestern Indiana. The size of the iamge is 145 × 145. It has 220 spectral bands. A total of 16 classes of land-covers are labeled for classification. The number of labeled samples and training samples for each category is shown in Table 1. The pseudo-color image and ground-truth map is shown in Fig. 2.

(2) Kennedy Space Center(KSC)

KSC dataset was taken in Florida, the size is 614 × 256. KSC contains 13 classes of land-covers and 176 spectral bands. The number of labeled samples and training samples for each category is shown in Table 1. The pseudo-color image and ground-truth map is shown in Fig. 2.

Table 1. The number of training samples and test samples for each class of land-cover in Indian Pines.

Class No.	Class Color	Class Name	Training	Testing
1		Alfalfa	15	31
2		Corn Notill	50	1378
3		Corn Mintill	50	780
4		Corn	50	187
5		Grass Pasture	50	433
6		Grass Trees	50	680
7		Grass Pasture Mowed	15	13
8		Hay Windrowed	50	428
9		Oats	15	5
10		Soybean Notill	50	922
11		Soybean Mintill	50	2405
12		Soybean Clean	50	543
13		Wheat	50	155
14		Woods	50	1215
15		Buildings Grass Trees Drives	50	336
16		Stone Steel Towers	50	43
		Total	695	9554

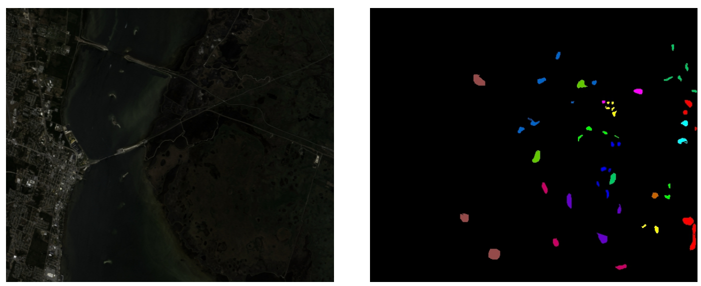


(a) (b)

Figure 2. Visualization of India Pines dataset. (a) Pseudo-color map. (b) Ground truth map.

Table 2. The Number Of Training Samples And Test Samples For Each Class Of Land-cover In Kennedy Space Center

Class No.	Class Color	Class Name	Training	Testing
1		Srub	30	728
2		Willow swamp	30	220
3		CP hammock	30	232
4		Slash pine	30	228
5		Oak/Broadleaf	30	146
6		Hardwood	30	207
7		Swamp	30	96
8		Graminoid	30	393
9		Spartina marsh	30	469
10		Cattail marsh	30	365
11		Salt marsh	30	378
12		Mud flats	30	454
13		Water	30	836
		Total	390	4752



(a) (b)

Figure 3. Visualization of KSC dataset. (a) Pseudo-color map. (b) Ground truth map.

3.2 Experimental Settings

For S^2HCN, we use the Adam optimizer[19] , the initial learning rate is set to 0.01 and will be dynamically updated, the hyperparameter σ is set to 1000, and the number of epochs is set to 200.

Simultaneously, we select four comparison methods, namely SVM with Gaussian kernel function, 2DCNN, 3DCNN[20] , and Funet-M[15] . Four evaluation indexes per-class accuracy, average accuracy, overall accuracy and kappa coefficient were selected to compare the results.

3.3 Classification Result

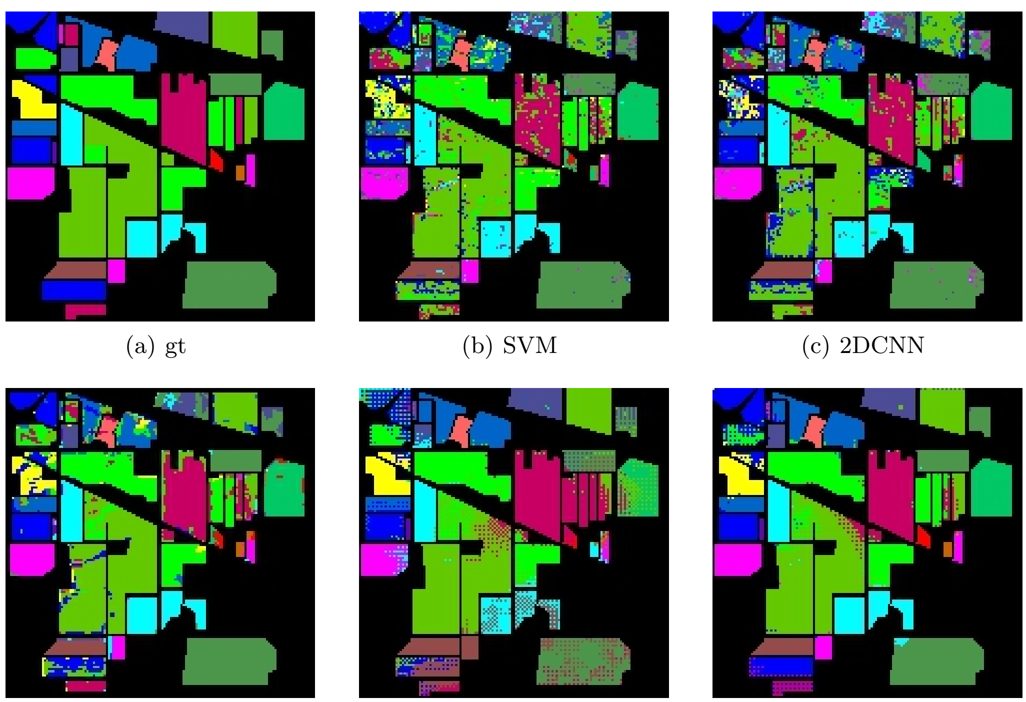
The classification results of the two datasets are shown in Table 3 and Table 4. Figure 4 and Figure 5 are visualizations of classification results. By analyzing the above tables and pictures, we can find that S^2HCN has the best performance on both datasets, which proves the effectiveness of our method. Meanwhile, as a relatively mature CNN method, 3DCNN performs better than 2DCNN and SVM. Compared with Funet-M, another GCN method that does not fuse spectral-spatial features, S^2HCN also has obvious advantages, which also proves the importance of spectral-spatial feature fusion.

4. CONCLUSION

In this paper, we propose a novel network S^2HCN for HSI classification. S^2HCN extracts spectral-spatial features first, and then utilizes hypergraph convolutional network for training. It is worth noting that we specifically propose the extraction method of spatial features, the generation method of hypergraph structure, and design a two-layer hypergraph convolutional network. Sufficient experiments on two datasets demonstrate the validity of S^2HCN.

Table 3. Per-class accuracy, Overall accuracy(OA), Average accuracy(AA), and Kappa coefficient Acquired by Different Method on Indian Pines dataset

Class No.	SVM	2DCNN	3DCNN[20]	FuNet-M[15]	S^2HCN
1	46.21	62.58	54.73	36.96	**100.00**
2	73.48	65.64	84.21	74.37	**88.66**
3	67.82	46.82	73.88	54.82	**95.06**
4	58.21	82.31	65.57	**99.70**	99.58
5	88.79	87.10	88.86	66.05	**96.27**
6	88.80	50.05	93.64	78.77	**99.59**
7	31.64	93.57	65.83	10.71	**100.00**
8	93.27	14.53	94.03	62.13	**100.00**
9	18.25	74.49	40.90	**100.00**	**100.00**
10	70.03	75.78	81.26	**86.11**	85.91
11	79.05	69.40	86.14	**87.94**	85.91
12	66.01	86.13	72.91	92.58	**96.46**
13	93.88	91.84	87.50	**100.00**	99.51
14	92.39	42.18	94.22	76.13	**99.37**
15	53.46	93.27	62.05	89.38	**98.70**
16	94.55	87.59	89.34	**100.00**	**100.00**
OA(%)	78.17	75.60	83.51	79.36	**92.75**
AA(%)	69.74	70.21	77.19	75.98	**96.56**
Kappa	0.7503	0.7214	0.8119	0.7612	**0.9099**



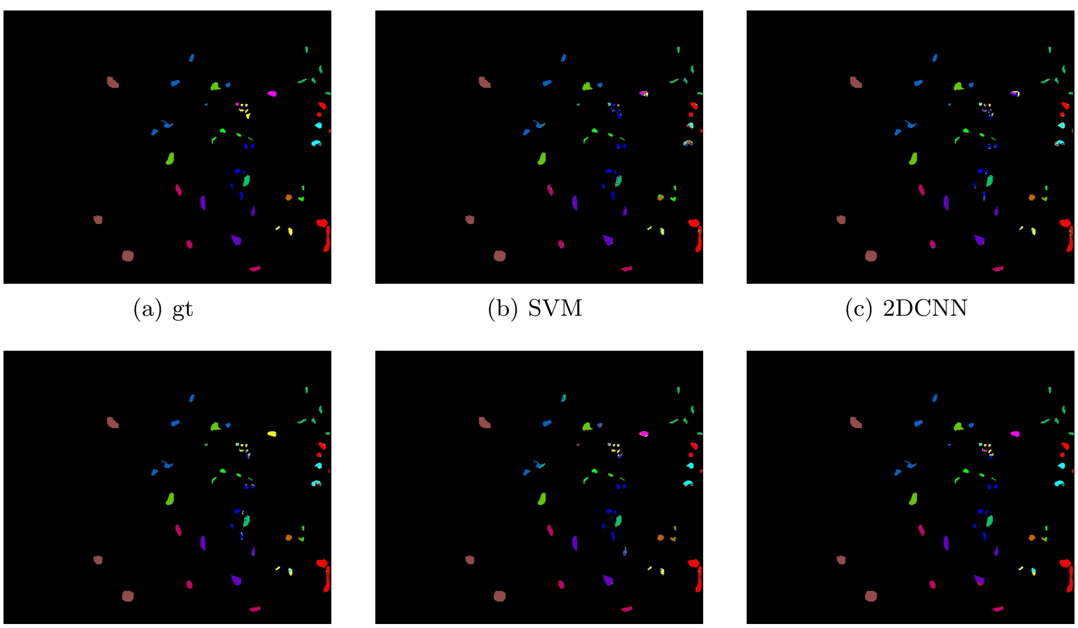
(a) gt (b) SVM (c) 2DCNN

(d) 3DCNN (e) FuNet-M (f) S^2HCN

Figure 4. Visualization of the classification results of different methods on Indian Pines dataset. (a) gt. (b) SVM. (c) 2DCNN. (d) 3DCNN. (e) Funet-M. (f) S^2HCN.

Table 4. Per-class accuracy, Overall accuracy(OA), Average accuracy(AA), and Kappa coefficient Acquired by Different Method on KSC dataset

Class No.	SVM	2DCNN	3DCNN[20]	FuNet-M[15]	S^2HCN
1	92.53	96.78	96.93	96.98	**100.00**
2	86.64	**89.91**	89.70	71.60	88.07
3	74.61	73.05	83.45	**100.00**	**100.00**
4	36.64	52.75	55.45	**66.27**	59.92
5	41.07	32.61	12.07	75.16	**81.89**
6	55.50	67.44	80.41	90.83	**100.00**
7	71.01	80.81	89.73	**100.00**	**100.00**
8	83.59	94.56	92.48	95.59	**99.53**
9	92.29	**98.30**	97.39	92.69	93.85
10	94.40	96.61	99.30	99.75	**100.00**
11	97.53	99.07	99.60	100.00	**100.00**
12	88.71	96.03	97.67	77.14	**100.00**
13	**100.00**	**100.00**	**100.00**	**100.00**	**100.00**
OA(%)	86.67	90.45	91.58	92.11	**96.30**
AA(%)	78.04	82.92	84.17	89.70	**94.10**
Kappa	0.8508	0.8943	0.9061	0.9107	**0.9498**



(a) gt (b) SVM (c) 2DCNN

(d) 3DCNN (e) FuNet-M (f) S^2HCN

Figure 5. Visualization of the classification results of different methods on KSC dataset. (a) gt. (b) SVM. (c) 2DCNN. (d) 3DCNN. (e) Funet-M. (f) S^2HCN.

ACKNOWLEDGMENTS

This work was supported in part by the National Key Research and Development Program of China (Grant No. 2019YFC1510905), and the Fundamental Research Funds for the Central Universities.

REFERENCES

[1] Landgrebe, D., "Hyperspectral image data analysis," *IEEE Signal processing magazine* **19**(1), 17–28 (2002).

[2] Harsanyi, J. C. and Chang, C.-I., "Hyperspectral image classification and dimensionality reduction: An orthogonal subspace projection approach," *IEEE Transactions on geoscience and remote sensing* **32**(4), 779–785 (1994).

[3] Rodarmel, C. and Shan, J., "Principal component analysis for hyperspectral image classification," *Surveying and Land Information Science* **62**(2), 115–122 (2002).

[4] Du, B., Zhang, L., Zhang, L., Chen, T., and Wu, K., "A discriminative manifold learning based dimension reduction method for hyperspectral classification," *International Journal of Fuzzy Systems* **14**(2), 272–277 (2012).

[5] Ma, L., Crawford, M. M., and Tian, J., "Local manifold learning-based k-nearest-neighbor for hyperspectral image classification," *IEEE Transactions on Geoscience and Remote Sensing* **48**(11), 4099–4109 (2010).

[6] Rajadell, O., Garcia-Sevilla, P., and Pla, F., "Spectral–spatial pixel characterization using gabor filters for hyperspectral image classification," *IEEE Geoscience and Remote Sensing Letters* **10**(4), 860–864 (2012).

[7] Fauvel, M., Benediktsson, J. A., Chanussot, J., and Sveinsson, J. R., "Spectral and spatial classification of hyperspectral data using svms and morphological profiles," *IEEE Transactions on Geoscience and Remote Sensing* **46**(11), 3804–3814 (2008).

[8] Pal, M. and Foody, G. M., "Feature selection for classification of hyperspectral data by svm," *IEEE Transactions on Geoscience and Remote Sensing* **48**(5), 2297–2307 (2010).

[9] Guo, B., Gunn, S. R., Damper, R. I., and Nelson, J. D., "Customizing kernel functions for svm-based hyperspectral image classification," *IEEE Transactions on Image Processing* **17**(4), 622–629 (2008).

[10] Kuching, S., "The performance of maximum likelihood, spectral angle mapper, neural network and decision tree classifiers in hyperspectral image analysis," *Journal of Computer Science* **3**(6), 419–423 (2007).

[11] Hu, W., Huang, Y., Wei, L., Zhang, F., and Li, H., "Deep convolutional neural networks for hyperspectral image classification," *Journal of Sensors* **2015** (2015).

[12] Chen, Y., Jiang, H., Li, C., Jia, X., and Ghamisi, P., "Deep feature extraction and classification of hyperspectral images based on convolutional neural networks," *IEEE Transactions on Geoscience and Remote Sensing* **54**(10), 6232–6251 (2016).

[13] Kipf, T. N. and Welling, M., "Semi-supervised classification with graph convolutional networks," in [*Proc. Int. Conf. Learn. Represent. (ICLR)*], 1–14 (2016).

[14] Qin, A., Shang, Z., Tian, J., Wang, Y., Zhang, T., and Tang, Y. Y., "Spectral–spatial graph convolutional networks for semisupervised hyperspectral image classification," *IEEE Geoscience and Remote Sensing Letters* **16**(2), 241–245 (2018).

[15] Hong, D., Gao, L., Yao, J., Zhang, B., Plaza, A., and Chanussot, J., "Graph convolutional networks for hyperspectral image classification," *IEEE Transactions on Geoscience and Remote Sensing* (2020).

[16] Wan, S., Gong, C., Zhong, P., Du, B., Zhang, L., and Yang, J., "Multiscale dynamic graph convolutional network for hyperspectral image classification," *IEEE Transactions on Geoscience and Remote Sensing* **58**(5), 3162–3177 (2019).

[17] Feng, Y., You, H., Zhang, Z., Ji, R., and Gao, Y., "Hypergraph neural networks," in [*Proceedings of the AAAI Conference on Artificial Intelligence*], **33**(01), 3558–3565 (2019).

[18] Bai, S., Zhang, F., and Torr, P. H., "Hypergraph convolution and hypergraph attention," *Pattern Recognition* **110**, 107637 (2021).

[19] Kingma, D. P. and Ba, J., "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980* (2014).

[20] Hamida, A. B., Benoit, A., Lambert, P., and Amar, C. B., "3-d deep learning approach for remote sensing image classification," *IEEE Transactions on geoscience and remote sensing* **56**(8), 4420–4434 (2018).