Faster Multiscale Capsule Network With Octave Convolution for Hyperspectral Image Classification

Qin Xu, Dongyue Wang, and Bin Luo, *Member, IEEE*

***Abstract*— Recently proposed capsule networks have revealed powerfulness in various visual tasks. However, the traditional CNNs adopted in the capsule layer of the capsule network have the problem of high parameter redundancy. In this letter, we propose a faster multiscale capsule network with octave convolution (MSOctCaps) for hyperspectral image classification. In the proposed MSOctCaps, we design multiple kernels of different sizes with parallel convolution to extract deep multiscale features. To feasibly reduce the redundancy of parameters and achieve high accuracy, the octave convolution is explored in the capsule layer, instead of the traditional convolution, which improves the accuracy of the capsule layer above predicted by the capsule layer below. The comparison experiments with six state-of-the-arts on two challenging contest data sets demonstrate the proposed MSOctCaps is able to produce competitive advantages in terms of both classification accuracy and computational time.**

***Index Terms*— Capsule networks, convolutional neural network (CNN), hyperspectral image (HSI) classification, octave convolution.**

I. Introduction

HYPERSPECTRAL images (HSIs) are composed of hundreds of continuous spectral bands from the visible to the near-infrared wavelength and have a wide range of applications, including global environmental monitoring, agricultural development management, and national defense security [1], [2]. In the HSIs processing techniques, HSI classification that attempts to assign a specific class to each pixel in the scene is a challenging task and has been considered to be a hot topic in the remote sensing community [3], [4].

There are many feasible classic methods to solve this problem. The support vector machine method, which achieves good classification performance with limited training samples, has once become a benchmark for HSI classification [5]. In [6], a self-paced joint sparse representation (JSR) model is proposed for HSI classification. This model is more accurate and robust than existing JSR methods [7], especially in the case of heavy noise. Some methods focus on band selection in order to select a small subset of hyperspectral bands to remove spectral redundancy and reduce computational costs [8].

In recent years, deep learning-based methods have shown promising performance in HSI classification. In [9], the application of deep learning methods in HSI classification was explored for the first time. It proposed a hybrid framework of principle component analysis, stacked autoencoders, and logistic regression. To capture both the spectral and spatial information in HSI, 3-D convolutional neural networks (CNNs) were used to extract the HSI spectral-spatial features for the HSI classification [10]. For the same season, [11] and [12] used new CNNs, such as deep residual networks and densely connected CNNs for HSI classification. Based on Hinton's capsule network [13], Paoletti et al. [14] proposed an extended CNN model in the aim to achieve highly accurate classification of HSI. In order to improve the feature extraction ability of CNNs, multiscale CNNs (MS-CNNs) [15] and multiscale capsule network (MS-CapsNet) [16] were developed. However, these two methods enhance multiscale representation strength by increasing the number of conventional parallel convolution. This process results in substantial redundancy in the spatial dimension of the feature maps. More recently, an octave convolution was proposed to reduce the spatial redundancy in the convolution process by condensing the low-frequency features and keeping the high-frequency features of images [17]. This helps each convolutional layer gain a larger receptive field to capture more contextual information [17].

Inspired by the aforementioned work, in this letter, we propose a faster MS-CapsNet with octave convolution, which we term MSOctCaps for short, for hyperspectral image (HSI) classification. Although the capsule network has shown efficiency in HSI classification, due to the existence of multiple parallel convolution operations in the capsule layer of capsule network, the number of parameters is relatively large, and the memory resource consumption is high. Therefore, we embed the octave convolution into the capsule network effectively to reduce the parameters and accelerate the computation. The octave convolution processes different frequency components with corresponding convolutions and effectively enlarges the receptive field in the original pixel space, which brings about the accuracy improvement of the capsule layer above predicted by the capsule layer below and reduction in computation. Furthermore, in the feature extraction phase, the multiscale convolution layer, which has multiple kernels of different sizes with parallel convolution, is designed to enhance the feature representation ability of the network. In consequence,

Manuscript received August 29, 2019; revised November 13, 2019; accepted January 24, 2020. Date of publication February 5, 2020; date of current version January 21, 2021. This work was supported in part by the National Natural Science Foundation of China under Grant 61502003, Grant 61671018, Grant 61860206004, and Grant 61976003, and in part by the Key Research Project of Humanities and Social Sciences in Colleges and Universities of Anhui Province under Grant SK2019A0013. *(Corresponding author: Qin Xu.)*

The authors are with the Key Laboratory of Intelligent Computing and Signal Processing of Ministry of Education, School of Computer Science and Technology, Anhui University, Hefei 230601, China (e-mail: xuqin@ahu.edu.cn; dongyuewangahu@163.com; luobin@ahu.edu.cn).

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

Digital Object Identifier 10.1109/LGRS.2020.2970079

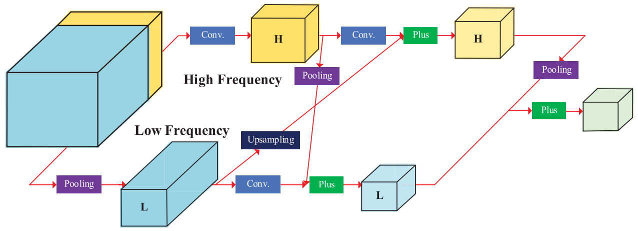

Fig. 1. Octave convolution diagram.

the proposed network structure enables a more efficient and stable classification of complex HSI data.

A detailed introduction of octave convolution and our proposed MSOctCaps network for HSI classification are described in Section II. In Section III, the data sets and the experimental results are presented and analyzed. Section IV concludes this letter.

II. Methodology

This section will first introduce the octave convolution and then develop an MSOctCaps network for HSIs classification, where the capsule network is primarily responsible for the encoding and decoding, and the octave convolution is used to optimize the capsule network and reduce the parameter redundancy.

A. Octave Convolution

As the natural images contain different frequencies, higher frequencies are usually encoded with fine details, and lower frequencies are usually encoded with global structures. Thus, the spatial resolution of the low-frequency group can be safely reduced by sharing information between neighboring locations to reduce spatial redundancy. The octave convolution attempts to factorize the mixed feature maps by their frequencies and processes the feature maps that vary spatially slower at a lower spatial resolution to reduce both memory and computation cost [17]. The octave convolution diagram is shown in Fig. 1. An input image is decomposed into two components according to the frequency of the mixed feature map during the convolution process. The smoothly varying low-frequency maps are stored in the low-resolution tensors to reduce spatial redundancy. Different convolution kernels are used for high-frequency and low-frequency features, respectively. The features of the same frequency are updated, and the features of different frequencies are exchanged; in other words, the new high-frequency feature map is obtained by adding the convolution of the original high-frequency part and the low-frequency part obtained by upsampling, meanwhile, the new low-frequency feature map is obtained by adding the pooled original high-frequency part and the convolved low-frequency part.

B. Proposed MSOctCaps Network

To learn high-level spectral-spatial features and improve the accuracy of the capsule layer above predicted by the capsule layer below, a faster MSOctCaps for HSI classification is developed. The flowchart of the MSOctCaps network, which includes the encoding and decoding, is shown in Fig. 2.

In the encoding phase of the network, we adopt the 3-D cube as the initial input data which is then entered into multiple parallel convolutional layers. To effectively extract and fuse a multiscale image feature, convolution kernels of different sizes are designed. The multiple convolutional layers are all activated by the ReLU function, and then the output feature maps are concatenated. The fused feature is inputted to the convolutional layers with the batch normalization algorithm and the ReLU activation function. The spectral dimension size of the convolution kernel is equal to the number of spectral bands of input data to reduce the spectral redundancy information.

Then, we design a capsule layer that consists of eight-octave convolution capsules (octconv-caps) to encode the feature map into a multidimensional vector whose dimensions are equal to the total number of capsules in parallel. In this layer, we define α as the high-frequency proportionality factor, then the number of the high-frequency feature map channels are the channels of the input feature map multiplied by α and the number of the low-frequency feature map channels is the channels of the input feature map multiplied by $1 - \alpha$. The new high-frequency feature is obtained by pooling and convolving the initial high-frequency feature and upsampling the initial low-frequency feature. Similarly, the new low-frequency feature is obtained by convolving the initial low-frequency feature and pooling the initial high-frequency feature. Finally, we pool the new high-frequency feature and add it to the new low-frequency feature to get the whole final feature which is flattened into a vector.

In addition, the output vector of octconv-caps layer is the input of dynamic routing capsules (routing-caps) layer where the number of routing-caps is equal to the number of categories. Let $\mathbf{u_i}$ denote the output vector of the octconv-cap i, and the predicted output vector $\mathbf{s_j}$ of the routing-cap j is represented by

$$\mathbf{s_j} = \sum_i c_{ij} \mathbf{W_{ij} u_i} \tag{1}$$

where the weight matrix $\mathbf{W_{ij}}$ is learned during the back propagation of the network and c_{ij} is the correlation coefficient between the octconv-cap i and the routing-cap j. It is calculated by

$$c_{ij} = \frac{e^{b_{ij}}}{\sum_k e^{b_{ik}}} \tag{2}$$

where k is the number of capsules in the routing-caps layer, and the log probabilities b_{ij} are set to zero at the initial stage of the routing by agreement process. The final output vector $\mathbf{v_j}$ of the routing-cap j is computed by a nonlinear squashing function

$$\mathbf{v_j} = \frac{\|\mathbf{s_j}\|^2}{1 + \|\mathbf{s_j}\|^2} \frac{\mathbf{s_j}}{\|\mathbf{s_j}\|}. \tag{3}$$

The log probabilities b_{rmij} are updated by the actual outputs of the $\mathbf{v_j}$ as following:

$$b_{ij} \leftarrow b_{ij} + \mathbf{v_j W_{ij} u_i}. \tag{4}$$

In the decoding process, the output vector in the encoding phase is used as the input data that are inputted into the

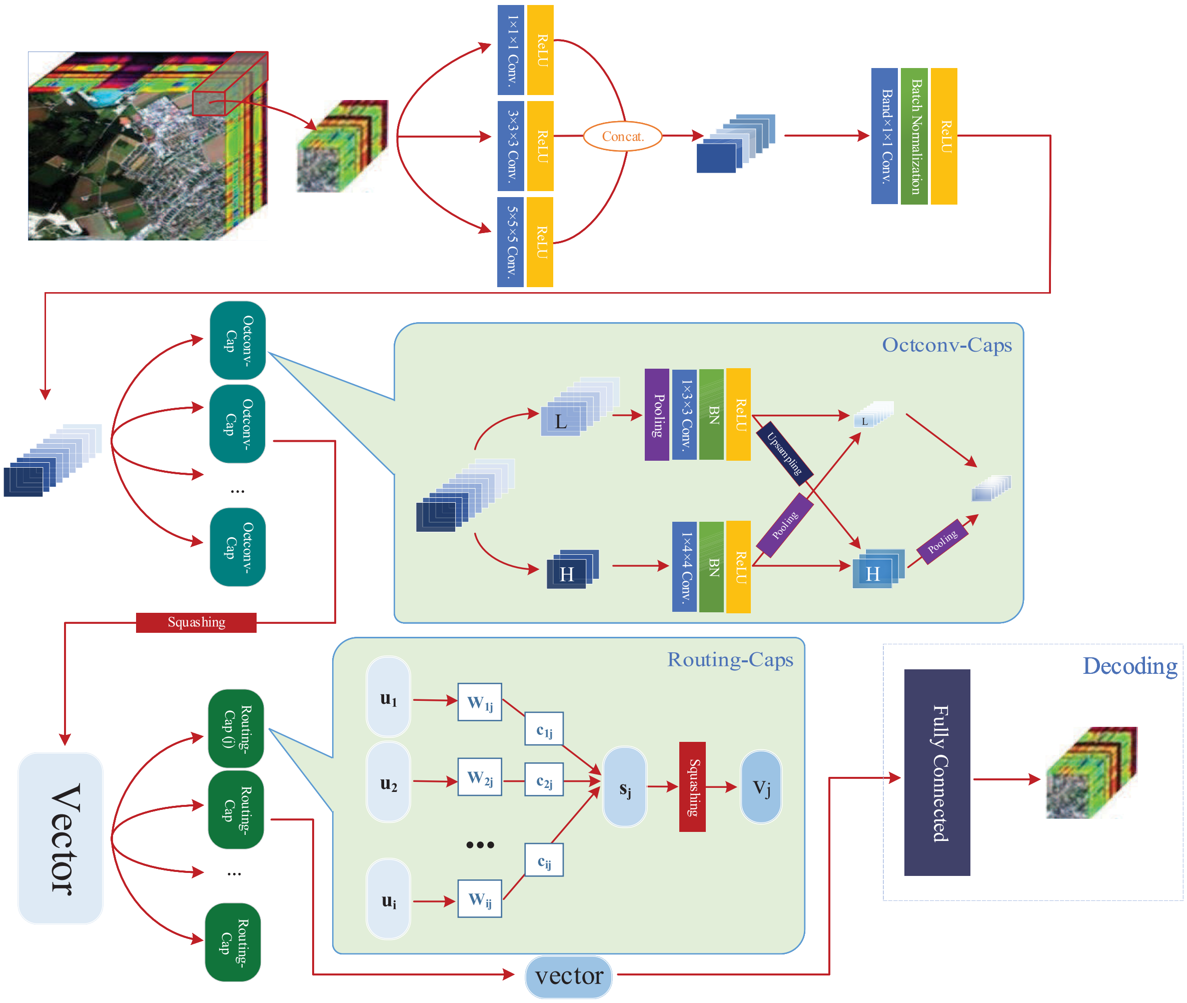

Fig. 2. Flowchart of the proposed MSOctCaps network for HSI classification.

fully connected layer to reconstruct the original input data. This process is used as a regularization term L_r which is the reconstruction loss determined by the square of the difference between the output of fully connected layer and the original input. The total loss function L_t given by

$$L_t = L_k + \theta L_r \tag{5}$$

where θ is a regularization factor to balance the two loss metrics and L_k is the edge loss computed by

$$L_k = T_k \max(0, m^+ - \|\mathbf{v_k}\|)^2 + \lambda(1 - T_k)\max(0, \|\mathbf{v_k}\| - m^-)^2 \tag{6}$$

where $m^+ = 0.9$ and $m^- = 0.1$ are default free parameters, the parameters λ guarantees the final convergence, and if the k class has been recognized, then $T_k = 1$, otherwise $T_k = 0$.

III. Experiments

In this section, we demonstrate the validity of the proposed MSOctcaps on two HSI data sets by comparing with CNNs [9], 3DCNNs [10], MS3DCNNs [15], ResNet [18], Caps [14], and MSCaps in classification accuracy and time consumption. The networks are written in the TensorFlow architecture. All experiments are executed on i7-8700 CPU and GTX 1060 GPU.

A. Description of Data Sets

1) Grss_dfc_2013: The Grss_dfc_2013 data set [19] is a public hyperspectral data set used in the 2013 IEEE GRSS Data Fusion Contest. It is acquired over the University of Houston, Houston, TX, USA, and the neighboring urban area, with 349 × 1905 pixels size and 2.5-m spatial resolution. The HSI has 144 spectral bands in the 0.4–1.0-lm region. All 15029 marked pixels are divided into 15 classes.

2) Grss_dfc_2014: The Grss_dfc_2014 data set [20] is a more challenging hyperspectral data set which is used in the 2014 IEEE GRSS Data Fusion Contest. It covers an urban area near Thetford Mines, QC, Canada, with 84 channels and 1-m spatial resolution. It is acquired from long-wave infrared (thermal infrared) bands between 7.8- and 11.5-lm wavelengths. The size of this data set is 795 × 564 pixels with 22 532 pixels labeled. A ground truth with seven land cover classes is provided.

B. Experimental Setup

In all experiments, the input data cube size is 12 (height) × 12 (width) × 80 (the number of spectral bands), where the high-dimensional spectral bands are processed by PCA. The number of training batches is 128. An Adam solver with a

TABLE I

CLASSIFICATION RESULTS FOR THE GRSS_DFC_2013 DATA SET

Class	Samples	Methods						
	Train/Test	CNNs	3DCNNs	MS3DCNNs	ResNet	Caps	MSCaps	MSOctCaps
Healthy grass	200/1051	99.09 ± 0.42	99.33 ± 0.57	98.86 ± 0.67	**100.0 ± 0.00**	99.67 ± 0.24	99.90 ± 0.10	99.81 ± 0.19
Stressed grass	200/1054	97.01 ± 2.60	99.38 ± 0.43	99.48 ± 0.14	99.91 ± 0.00	99.67 ± 0.05	**99.29 ± 0.24**	99.81 ± 0.19
Synthetic grass	200/497	99.59 ± 0.20	99.90 ± 0.10	99.90 ± 0.10	**100.0 ± 0.00**	99.80 ± 0.00	99.70 ± 0.10	99.80 ± 0.00
Trees	200/1044	95.83 ± 2.25	**100.0 ± 0.00**	99.81 ± 0.10	99.71 ± 0.19	99.38 ± 0.34	99.95 ± 0.05	99.43 ± 0.02
Soil	200/1042	98.32 ± 0.14	100.0 ± 0.00	99.81 ± 0.19	100.0 ± 0.00	99.04 ± 0.00	100.0 ± 0.00	**100.0 ± 0.00**
Water	200/125	97.60 ± 2.40	100.0 ± 0.00	**100.0 ± 0.00**	100.0 ± 0.00	98.80 ± 0.40	100.0 ± 0.00	99.20 ± 0.19
Residential	200/1068	90.68 ± 0.14	97.10 ± 0.56	99.06 ± 0.66	99.11 ± 0.61	99.67 ± 0.33	98.27 ± 0.89	**99.95 ± 0.05**
Commercial	200/1044	85.49 ± 1.96	98.52 ± 0.43	98.08 ± 0.38	96.36 ± 1.25	97.89 ± 0.19	98.47 ± 1.15	**99.57 ± 0.43**
Road	200/1052	84.70 ± 5.32	99.24 ± 0.19	97.48 ± 0.71	97.43 ± 0.29	99.14 ± 0.38	99.14 ± 0.10	**98.57 ± 0.57**
Highway	200/1027	92.36 ± 0.83	99.17 ± 0.83	99.81 ± 0.19	99.90 ± 0.10	99.61 ± 0.00	**100.0 ± 0.00**	99.85 ± 0.15
Railway	200/1035	87.15 ± 1.64	98.99 ± 0.72	99.66 ± 0.14	99.66 ± 0.34	98.16 ± 0.39	**99.76 ± 0.24**	99.32 ± 0.29
Parking Lot 1	200/1033	83.45 ± 1.36	99.23 ± 0.29	98.60 ± 1.31	99.27 ± 0.73	99.37 ± 0.53	**99.95 ± 0.05**	99.71 ± 0.10
Parking Lot 2	200/269	68.59 ± 0.93	99.81 ± 0.19	99.81 ± 0.19	99.81 ± 0.19	96.65 ± 0.37	**100.0 ± 0.00**	97.77 ± 2.23
Tennis Court	200/228	99.78 ± 0.22	100.0 ± 0.00	100.0 ± 0.00	100.0 ± 0.00	100.0 ± 0.00	100.0 ± 0.00	**100.0 ± 0.00**
Running Track	200/460	99.46 ± 0.11	100.0 ± 0.00	100.0 ± 0.00	100.0 ± 0.00	100.0 ± 0.00	99.89 ± 0.11	**100.0 ± 0.00**
OA(%)		91.77 ± 0.35	99.20 ± 0.08	99.18 ± 0.06	99.24 ± 0.09	99.47 ± 0.10	99.52 ± 0.05	**99.59 ± 0.06**
AA(%)		91.30 ± 0.13	99.27 ± 0.04	99.12 ± 0.29	98.97 ± 0.05	99.59 ± 0.08	99.51 ± 0.15	**99.62 ± 0.06**
$\kappa\times 100$		91.06 ± 0.38	99.13 ± 0.09	99.11 ± 0.06	99.18 ± 0.10	99.43 ± 0.11	99.48 ± 0.06	**99.56 ± 0.06**

TABLE II

CLASSIFICATION RESULTS FOR THE GRSS_DFC_2014 DATA SET

Class	Samples	Methods						
	Train/Test	CNNs	3DCNNs	MS3DCNNs	ResNet	Caps	MSCaps	MSOctCaps
Road	200/4243	90.20 ± 5.68	99.58 ± 0.02	99.26 ± 0.11	99.95 ± 0.05	**99.96 ± 0.01**	99.94 ± 0.06	99.93 ± 0.00
Trees	200/893	32.81 ± 2.13	85.44 ± 4.82	86.73 ± 5.88	88.13 ± 2.69	85.61 ± 1.06	93.11 ± 2.74	**96.36 ± 1.85**
Red roof	200/1654	62.88 ± 5.02	89.42 ± 1.15	93.02 ± 0.33	89.60 ± 0.60	93.65 ± 0.91	97.19 ± 0.70	**98.67 ± 0.12**
Grey roof	200/1926	32.92 ± 0.67	88.42 ± 3.79	87.33 ± 2.02	93.87 ± 0.31	91.10 ± 0.08	98.00 ± 1.12	**98.91 ± 0.10**
Concrete roof	200/3688	44.59 ± 5.25	92.99 ± 0.20	94.27 ± 0.58	95.13 ± 1.40	92.94 ± 0.56	95.76 ± 2.07	**97.91 ± 1.36**
Vegetation	200/7157	46.97 ± 4.34	82.60 ± 3.19	86.87 ± 2.10	82.78 ± 1.28	89.25 ± 0.97	93.37 ± 3.35	**94.84 ± 1.45**
Bare soil	200/1571	54.68 ± 2.67	94.27 ± 2.67	95.86 ± 1.02	94.46 ± 0.57	94.27 ± 1.97	**98.44 ± 0.22**	97.87 ± 0.48
OA(%)		55.17 ± 3.87	89.88 ± 1.78	91.85 ± 0.99	91.02 ± 0.69	92.80 ± 0.36	96.37 ± 0.60	**97.37 ± 0.15**
AA(%)		51.42 ± 3.66	84.59 ± 2.11	86.78 ± 1.61	86.01 ± 0.79	88.14 ± 0.41	93.58 ± 1.34	**95.00 ± 0.48**
$\kappa\times 100$		46.31 ± 4.43	87.46 ± 2.16	89.86 ± 1.21	88.87 ± 0.83	91.02 ± 0.44	95.46 ± 0.73	**96.70 ± 0.18**

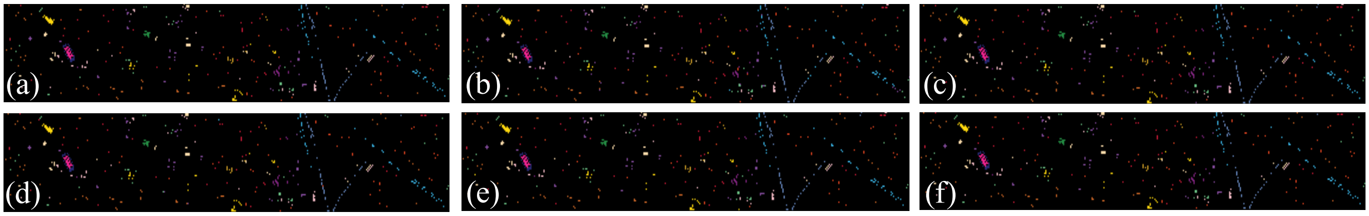

Fig. 3. Grss_dfc_2013 data set classification maps. (a) GT. (b) CNNs. (c) MS3DCNNs. (d) ResNet. (e) CAPs. (f) MSOctCaps.

learning rate of 1e-3 is adopted as an optimizer. Two hundred samples in each category are randomly selected as the training samples. In our network, high-frequency proportionality factor α is 0.25, and the final total loss function parameter θ takes a value of 0.006.

C. Results and Analysis

The detailed classification results of Grss_dfc_2013 and Grss_dfc_2014 are shown in Tables I and II, respectively. The evaluation criteria include the overall accuracy (OA) of all test samples, the average accuracy of each class, and the kappa coefficient (κ) to measure data consistency. From Tables I and II, we can see that our MSOctCaps outperform the CNNs, 3DCNNs, MS3DCNNs, ResNet, Caps, and MSCaps. Compared with other methods, our method can provide a better classification accuracy for the classes that have a large number of test samples, for example, in Grss_dfc_2014 class "vegetable" has 7157 test samples, which indicates that the proposed network is more robust and stable.

The specific time consumption of the two data sets is shown in Table III. From the table, we can see that the time consumption of MSOctCaps is much less than the capsule network and MSCaps, and comparable with the MS3DCNNs.

Figs. 3 and 4 show the classification maps obtained by CNNs, MS3DCNNs, ResNet, Caps, and our MSOctCaps on the Grss_dfc_2013 data set and Grss_dfc_2014 data set. From

TABLE III
TIME CONSUMPTION OF ALL METHODS

Time(min)	Methods				
Dataset	3DCNNs	MS3DCNNs	Caps	MSCaps	MSOctCaps
Grss_dfc_2013	6.67	9.98	44.37	58.84	12.65
Grss_dfc_2014	7.40	11.22	42.29	56.75	10.90

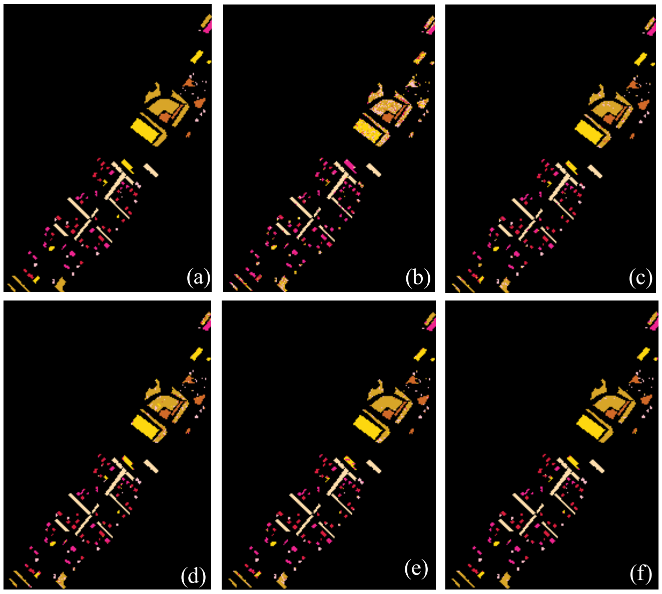

Fig. 4. Grss_dfc_2014 data set classification maps. (a) GT. (b) CNNs. (c) MS3DCNNs. (d) ResNet. (e) CAPs. (f) OctCaps.

TABLE IV
INFLUENCE OF DIFFERENT PARAMETERS

Parameters	Size		
Input patches	8 × 8	12 × 12	16 × 16
OA(%)	94.44	97.37	**98.01**
Band size	20	60	80
OA(%)	95.21	95.67	**97.37**
α	0.25	0.50	0.75
OA(%)	**97.37**	97.02	96.49
θ	6e-2	6e-3	6e-4
OA(%)	97.29	**97.37**	96.72

Table I, we can see that MSOctCaps outperforms others slightly in comparison with the state-of-the-art methods. This is because the between-class scatter of the Grss_dfc_2013 data set is larger and the within-class scatter is smaller. From Table II, MSOctCaps exhibits significant improvement on the Grss_dfc_2014 data set.

We also do more experiments on the Grss_dfc_2014 data set to explore the influence of different parameters, as shown in Table IV. Experiment results show that a larger spatial size can give an improved OA, and the PCA dimension has little effect on the OA. When α equals 0.25 and θ is equals to 6e-3, we get a relatively better classification result.

IV. CONCLUSION

In this letter, we have proposed a faster multiscale capsule network with octave convolution for HSI classification. The network utilizes octave convolution instead of traditional convolution in the capsule layer to reduce the parameter redundancy and improve the representation ability of spectral-spatial features of HSI. The MSOctCaps reduces the chance of overfitting and is more robust. The classification experiments on the Grss_dfc_2013 and Grss_dfc_2014 data sets also demonstrate that the proposed network has a faster learning speed and can provide a promising classification accuracy when the available training samples are limited.

REFERENCES

[1] S. Li, W. Song, L. Fang, Y. Chen, P. Ghamisi, and J. A. Benediktsson, "Deep learning for hyperspectral image classification: An overview," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 9, pp. 6690–6709, Sep. 2019.

[2] D. Landgrebe, "Hyperspectral image data analysis," *IEEE Signal Process. Mag.*, vol. 19, no. 1, pp. 17–28, Jan. 2002.

[3] C.-I. Chang, *Hyperspectral Data Exploitation: Theory and Applications*. Hoboken, NJ, USA: Wiley, 2007.

[4] B. Pan, Z. Shi, X. Xu, T. Shi, N. Zhang, and X. Zhu, "CoinNet: Copy initialization network for multispectral imagery semantic segmentation," *IEEE Geosci. Remote Sens. Lett.*, vol. 16, no. 5, pp. 816–820, May 2019.

[5] 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.

[6] J. Peng, W. Sun, and Q. Du, "Self-paced joint sparse representation for the classification of hyperspectral images," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 2, pp. 1183–1194, Feb. 2019.

[7] S. Shekhar, V. M. Patel, N. M. Nasrabadi, and R. Chellappa, "Joint sparse representation for robust multimodal biometrics recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 1, pp. 113–126, Jan. 2014.

[8] W. Sun and Q. Du, "Hyperspectral band selection: A review," *IEEE Geosci. Remote Sens. Mag.*, vol. 7, no. 2, pp. 118–139, Jun. 2019.

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

[10] Y. Li, H. Zhang, and Q. Shen, "Spectral–spatial classification of hyperspectral imagery with 3D convolutional neural network," *Remote Sens.*, vol. 9, no. 1, p. 67, Jan. 2017.

[11] L. Mou, P. Ghamisi, and X. X. Zhu, "Unsupervised spectral–spatial feature learning via deep residual conv–deconv network for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 1, pp. 391–406, Jan. 2018.

[12] M. Paoletti, J. Haut, J. Plaza, and A. Plaza, "Deep&dense convolutional neural network for hyperspectral image classification," *Remote Sens.*, vol. 10, no. 9, p. 1454, 2018.

[13] S. Sabour, N. Frosst, and G. E. Hinton, "Dynamic routing between capsules," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 3856–3866.

[14] M. E. Paoletti *et al.*, "Capsule networks for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 4, pp. 2145–2160, Apr. 2019.

[15] Z. Gong, P. Zhong, Y. Yu, W. Hu, and S. Li, "A CNN with multiscale convolution and diversified metric for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 6, pp. 3599–3618, Jun. 2019.

[16] C. Xiang, L. Zhang, Y. Tang, W. Zou, and C. Xu, "MS-CapsNet: A novel multi-scale capsule network," *IEEE Signal Process. Lett.*, vol. 25, no. 12, pp. 1850–1854, Dec. 2018.

[17] Y. Chen *et al.*, "Drop an octave: Reducing spatial redundancy in convolutional neural networks with octave convolution," 2019, *arXiv:1904.05049*. [Online]. Available: https://arxiv.org/abs/1904.05049

[18] Z. Zhong, J. Li, L. Ma, H. Jiang, and H. Zhao, "Deep residual networks for hyperspectral image classification," in *Proc. IEEE Int. Geosci. Remote Sens. Symp. (IGARSS)*, Jul. 2017, pp. 1824–1827.

[19] *2013 IEEE GRSS Data Fusion Contest*. Accessed: Jun. 23, 2012. [Online]. Available: http://www.grss-ieee.org/community/technical-committees/data-fusion/

[20] *2014 IEEE GRSS Data Fusion Contest*. Accessed: Jan. 27, 2014. [Online]. Available: http://www.grss-ieee.org/community/technical-committees/data-fusion/