



HAL
open science

Parallel Dimensionality-Variied Convolutional Neural Network for Hyperspectral Image Classification

Haicheng Qu, Xiu Yin, Xuejian Liang, Wanjun Liu

► **To cite this version:**

Haicheng Qu, Xiu Yin, Xuejian Liang, Wanjun Liu. Parallel Dimensionality-Variied Convolutional Neural Network for Hyperspectral Image Classification. 2nd International Conference on Intelligence Science (ICIS), Nov 2018, Beijing, China. pp.302-309, 10.1007/978-3-030-01313-4_32 . hal-02118826

HAL Id: hal-02118826

<https://inria.hal.science/hal-02118826>

Submitted on 3 May 2019

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution 4.0 International License

Parallel Dimensionality-varied Convolutional Neural Network for Hyperspectral Image Classification

Haicheng Qu¹, Xiu Yin¹, Xuejian Liang¹, Wanjun Liu¹

¹ Liaoning Technical University, College of Software, 188, longwan south street, Xing-Cheng, Huludao, China

quhaicheng@lntu.edu.cn
lntuyinxiu@hotmail.com
liangxuejian2015@hotmail.com
liuwanjun@lntu.edu.cn

Abstract. Many spectral-spatial classification methods of HSI based on convolutional neural network (CNN) are proposed and achieve outstanding performance recently. However, these methods require tremendous computations with complex network and excessively large model. Moreover, single machine is obviously weak when dealing with big data. In this paper, a parallel dimensionality-varied convolutional neural network (DV-CNN) is proposed to address these issues. The dimensionalities of feature maps extracted vary with stages in DV-CNN, and DV-CNN reduces the dimensionalities of feature maps to simplify the computation and the structure of network without information loss. Besides, the parallel architecture of DV-CNN can obviously reduce the training time. The experiments compared with state-of-the-art methods are performed on Indian Pines and Pavia University scene datasets. The results of experiments demonstrate that parallel DV-CNN can obtain better classification performance, reduce the time consuming and improve the training efficiency.

Keywords: parallel computing, dimensionality variation, convolutional neural network, hyperspectral image classification

1 Introduction

Hyperspectral data contain spectral and spatial information simultaneously [1], which have been used in various fields widely. In hyperspectral image (HSI) processing, dimensionality reduction is one of the most difficult and most necessary problem to be solved. In order to improve the accuracy of HSI classification, numerous methods have been proposed in the past few years.

Traditional HSI classification methods which generally focused on spectral bands included k-nearest neighbours, logistic regression, maximum likelihood, etc [2]. Spectral-spatial classification could improve the accuracy of classification prominently [3]. And deep learning promoted the development of HSI processing and many methods

based on convolutional neural network (CNN) were proposed for classification. Gu et al. proposed a Representative multiple kernel learning (RMKL) method to achieve good classification accuracy and interpretability [4]. Chan et al. proposed PCANet which learned extremely easily and efficiently [5]. Moreover, a stacked autoencoder (SAE) method [6] and a Deep Belief Network (DBN) method [7] were proposed by Chan et al. to deal with HSI classification, which were demonstrated the effectiveness of deep learning in HSI classification. The nonlinear spectral-spatial network (NSSNet), which was based on PCANet and proposed by Pan obtained better performance [8] than other methods. Some CNN-based methods for HSI classification such as 2D-CNN [9] and 3D-CNN [10] were proposed and achieved better performance. However, traditional methods barely made full use of HSI information and CNN-based methods required more tremendous computations and more complex networks because of the increase of kernels for more abundant and accurate features. In addition, the single worker was unable to meet the needs of big data, and parallel computing of cluster can reduce the computing time and improve the efficiency.

In this paper, a parallel dimensionality-varied convolutional neural network (DV-CNN) is proposed. The parallel DV-CNN is designed to solve tremendous amount of HSI data and improve the accuracy of classification. The results of experiments show that the parallel DV-CNN achieves 99.14% of overall accuracy (OA) on Indian Pines and 99.84% of overall accuracy (OA) on Pavia University scene, which is higher than the second best accuracy in CNN-based methods or other state-of-the-art methods. Besides, the parallel architecture of DV-CNN obviously reduces the training time consuming.

The rest of this paper is organized into three sections. In Section 2, a detailed description about parallel computing of DV-CNN is presented. Section 3 elaborates the experiments of HSI classification and parallel computing, and the comparisons with state-of-the-art methods are also present. Finally, Section 4 presents the conclusions.

2 Proposed method

This section elaborates the structure of parallel DV-CNN, which include the dimensionality variation of feature maps in four stages of DV-CNN, and the architecture description of parallel computing is elaborated in detail. The structure of parallel DV-CNN is shown in Fig. 1.

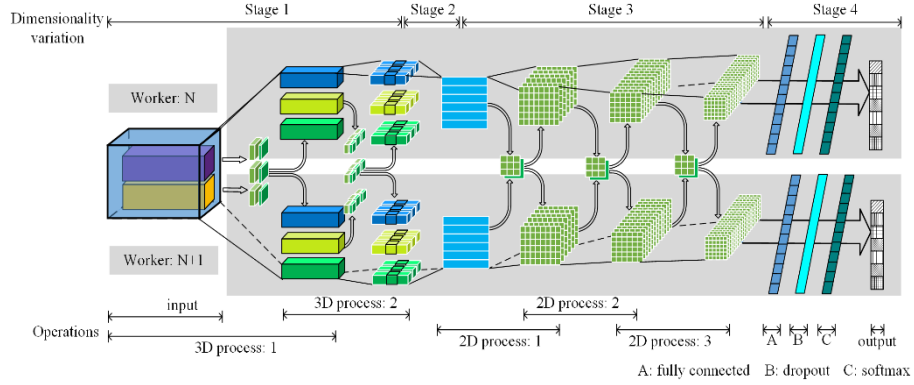


Fig. 1. Structure of parallel DV-CNN

2.1 Dimensionality-varied CNN

Parallel DV-CNN has four stages (shown in Fig. 1) and each of them relates to dimensionality variation. There are two 3D process and three 2D process in the feature extraction of DV-CNN. In stage 1, both 3D feature extraction contain 3D convolution and maxpooling operations. And the size of convolution kernel is related to specific dataset. One of the functions of stage 1 is the spectral-spatial features fusion. In stage 2, some 1D vectors are generated after stage 1 and assembled in to a 2D feature map as the input of the first 2D process. Stage 3 and stage 4 make up a general 2D-CNN for image processing. There are three 2D process for feature extraction which contain 2D convolution, activation and maxpooling operations in stage 3. The main components of stage 4 consists of a fully connected layer, a dropout layer and a softmax layer. The output of DV-CNN is also in stage 4. The dimensionality variation of feature map in DV-CNN reduces the computations, and the simplified structure of network improves the classification accuracy.

The convolution in first 3D process can be formulated as

$$\text{output}(m_{l,i}) = f\left(\text{conv}\left(\text{input}(m_{(l-1)}), k_{l,i}\right) + b_{l,i}\right) \quad (1)$$

where l indicates the layer that is considered, and $m_{(l-1)}$ is the input. $k_{l,i}$ is the convolution kernels with i being the index of feature map generated, and $b_{l,i}$ is the bias corresponded to $k_{l,i}$. The function $\text{conv}(\cdot)$ represents convolution operation, and activation function $f(\cdot)$ represents Rectified Linear Unites (ReLU), which is a non-saturation function and faster than other saturating function. $m_{l,i}$ is the result of convolution.

The convolution in second 3D process can be formulated as

$$\text{output}(m_{(l+1),i,j}^h) = \sum_{x=1}^L \sum_{y=1}^W \sum_{z=1}^H \text{input}(n_{l,i}^{g(x),g(y),g(z)}) k_{(l+1),i,j}^{x,y,z} \quad (2)$$

$$g(x) = x + s(h-1) \quad (3)$$

In Eq.(2) and Eq. (3), $k_{l,i,j}^{x,y,z}$ is one of the elements in convolution kernel $k_{l,i,j}$ with j being the index of kernel, and $n_{(l-1),i}^{g(x),g(y),g(z)}$ is one of the elements in $n_{(l-1),i}$. (x, y, z) indicates the position of elements in them respectively, and i is the index of input 3D feature map $n_{(l-1),i}$ in $(l-1)$ th layer with L , W and H being the length, width and height separately. The $m_{(l+1),i,j}^h$ is one of the elements in 1D vector generated by convolution in the second 3D process with h being the index. Besides, s stands for the stride of convolution in bands.

2.2 Parallel computing

In order to avoid excessive consumption of time, the DV-CNN is distributed on different workers of a cluster, and each worker typically has a same task for training. Besides, the parallel DV-CNN (show in Fig. 1) has a master server and some worker servers in the cluster which share parameters and datasets. All servers update parameter asynchronously to make the most of computing ability of each worker. The master server is focus on the initialization of parameters, the deployment of parallel network and the test of classification accuracy by sampling every a certain times.

The input of parallel DV-CNN is a cube of HSI which consists of spectral-spatial information. Each worker in this cluster selects different bands of a neighbour region to extract features, and feature maps are generated by 3D/2D processes. After four stages, the parameters in convolution kernels are updated by workers. Thus, every worker contributes to the update of parameters, so that the network is easier to converge and save training time. The time consuming T can be formulated as

$$T = T_d + T_w + T_c \quad (4)$$

where T_d is the time consuming of parameters initialization and tasks deployment, T_w is the computing time consuming of worker in cluster, and T_c is the communication time between workers in the network and I/O time consuming.

3 Experiments

In this paper, we used two real HSI datasets, namely Indian Pines and Pavia University scene for experiments. And the superiority of DV-CNN on accuracy and time consuming was based on these datasets. Besides, the ratio of labelled samples for training and testing was randomly divided into 1: 1.

The dropout and learning rate of network were set as 0.8 and 1×10^{-3} respectively, and the value of cross entropy was set as 1×10^{-15} which meant all values in classification results were set as 1 if they were smaller than 1×10^{-15} . The neighbour region selected from HSI cube was 10×10 for Indian Pines dataset and 8×8 for Pavia University scene dataset. Besides, the batch size input into network was 40 for Indian pines and 150 for Pavia University scene. The other parameters are displayed in Table 1 and Table 2.

Table 1. Parameters of 3D feature extraction of DV-CNN

data set	3D feature extraction			
	kernel number, kernel size		pooling size	
	process: 1	process: 2	process:1	process:2
Indian Pines	8, $1 \times 5 \times 5$	36, $21 \times 3 \times 3$ $1 \times 3 \times 3$	$1 \times 2 \times 2$	$2 \times 1 \times 1$
Pavia University scene	5, $1 \times 3 \times 3$	36, $4 \times 3 \times 3$	$1 \times 2 \times 2$	$2 \times 1 \times 1$

Table 2. Parameters of 2D feature extraction of DV-CNN

data set	2D feature extraction			
	kernel number, kernel size			pooling size
	process: 1	process: 2	process: 3	process: all
Indian Pines	6, 5×5	12, 5×5	36, 5×5	2×2
Pavia University scene	6, 5×5	12, 5×5	36, 5×5	2×2

The parallel DV-CNN were based on Tensorflow architecture and implemented on servers with Intel Core i7-6700@3.4 GHz CPU, 16 GB of RAM. And there was a NVIDIA GTX 1080 Ti GPU card, a NVIDIA GTX 1070 GPU card and two NVIDIA GTX 1060 GPU cards in cluster.

3.1 Classification accuracy of methods

Classification accuracy of methods based on CNN. We compared some methods based on CNN (2D-CNN, 3D-CNN) for HSI classification to evaluate the classification performance of parallel DV-CNN, and all of them were based on spectral-spatial feature extraction. The convolution kernels is two dimensions for 2D-CNN and three dimensions for 3D-CNN in all feature extractions, which are different from kernels in DV-CNN. The experiment parameters of DV-CNN were the same as other methods. The results of comparison were shown in Table 3.

In Table 3, 3D-CNN and DV-CNN achieved better classification performance than 2D-CNN because of the simultaneous spectral-spatial features extraction, which was different from 2D-CNN. Besides, the performance of DV-CNN was better than 3D-CNN. The OA of DV-CNN was up to 99.14% on Indian Pines and 99.84% on Pavia University scene, which was 0.07% and 0.45% higher than 3D-CNN. Moreover, both average accuracy (AA) and kappa coefficient (κ) achieved similar results as OA.

Table 3. Results of comparison with CNNs

	2D-CNN	3D-CNN	DV-CNN
Indian Pines			
OA(%)	95.97±0.0938	99.07±0.0345	99.14±0.1175
AA(%)	93.23 ±0.7629	98.66±0.0345	98.71±0.65606
κ	0.9540±0.0012	0.9893±0.0004	0.9926±0.0010
Pavia University scene			
OA(%)	99.03±0.0142	99.39±0.0098	99.84±0.0343
AA(%)	98.19±0.0268	99.85±0.0609	99.74±0.0693
κ	0.9817±0.0001	0.9920±0.0002	0.9979±0.0005

Classification accuracy with state-of-the-art methods. We also compared some traditional and other deep learning method for HSI classification. RMKL, PCANet, SAE-LR, DBN-LR, and NSSNet were performed an optimized effect in HSI classification, and introduced to experiments to compare with DV-CNN. The results of comparison were shown in Table 4.

Table 4. Comparison results of classification with other methods

	Indian Pines			Pavia University scene		
	OA(%)	AA(%)	κ	OA(%)	AA(%)	κ
RMKL	95.61	94.20	0.9499	96.06	94.48	0.9443
PCANet	86.58	85.16	0.8471	93.20	91.01	0.8997
SAE-LR	92.58	90.38	0.9152	98.69	98.17	0.9829
DBN-LR	95.95	95.45	0.9539	99.05	98.48	0.9875
NSSNet	96.08	96.40	0.9547	99.50	99.03	0.9910
DV-CNN	99.14	98.71	0.9926	99.84	99.74	0.9979

As shown in Table 4, DV-CNN obtained the best classification results of OA, AA and κ on both Indian Pines and Pavia University scene datasets. And the OA of DV-CNN was 3.18% higher than the second best obtained by NSSNet on Indian Pines dataset and 0.34% higher on Pavia University scene dataset.

3.2 Time consuming of methods

In order to verify the efficiency of parallel computing for big data in HSI classification, we conducted some comparison experiments on both Indian Pines dataset and Pavia University dataset. The experiments were divided into four categories based on the number of servers in parallel cluster. We tested the classification efficiency of NVIDIA GTX 1060 GPU cards (1060-1, 1060-2) and NVIDIA GTX 1080 GPU cards (1080Ti) for single server computing, two NVIDIA GTX 1060 GPU cards for parallel computing (parallel-2), two NVIDIA GTX 1060 GPU cards and a NVIDIA GTX 1080Ti GPU card (as master) for parallel computing (parallel-3), and all of the GPU cards in cluster for parallel computing (parallel-4). Moreover, we also tested the impact of network on computing time. The parallel-3-1 represented that the parallel computing was in a 1000Mb/s network and parallel-3-2 was in a 100Mb/s network. We counted the time consuming every 100 iterations. The curve of Indian Pines was shown in Fig. 2 and the curve of Pavia University scene was shown in Fig. 3.

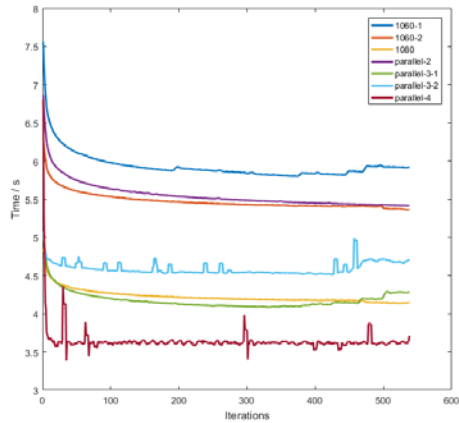


Fig. 2. The curve of time consuming on Indian Pines

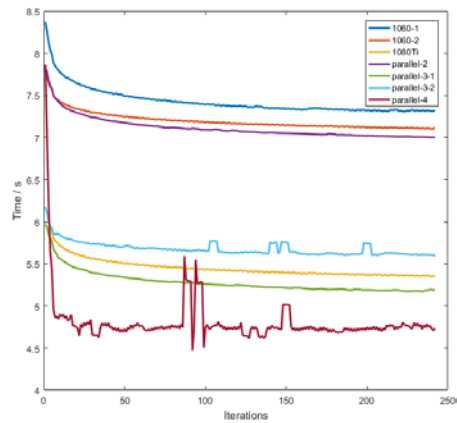


Fig. 3. The curve of time consuming on Pavia University scene

As shown in Fig. 2 and Fig. 3, the time consuming of each model in the beginning is much more than in other iteration points, which is resulting from the task deployment and initialization of parallel network. In terms of single computing, 1080Ti is much faster than 1060s. According to the curves of parallel-2, parallel-3-1 and parallel-4, we can infer that increasing the number of worker in cluster can improve the average computing speed and save lots of time. Besides, parallel-3-1 takes less computing time than parallel-3-2, which demonstrates that the hardware of network can influence the parallel efficiency. Moreover, the fluctuation of the parallel-4 curve is much larger, and the impact of network on the parallel computing is more obvious than others. Thus, the parallel efficiency increases when the number of worker in cluster increases, but it decreases when the number of worker is exceeds a certain number.

4 Conclusions

In this paper, we have proposed a parallel DV-CNN framework for HSI classification. The DV-CNN can make full use of spectral-spatial information contained in HSI to extract more accurate features. The experiments compared with the state-of-the-art methods demonstrate that DV-CNN can improve the classification accuracy. DV-CNN also simplifies the complexity and reduces the computations of CNN-based model. Besides, the parallel architecture is feasible for HSI classification methods based on CNNs. It has been shown that parallel DV-CNN is a suitable approach for HSI classification dealing with big data.

However, the communication of workers in cluster is a major factor to influence the time consuming of parallel computing, and the consumption of communication time is obviously influenced by the increased of worker number. In the further research, we mainly research on the specific effect of increasing number of workers in cluster on parallel time consuming.

Acknowledgments

This paper was supported by the National Nature Science Foundation of China (41701479).

References

1. Zhao W. Z., Du S. H., "Spectral-spatial feature extraction for hyperspectral image classification: A dimension reduction and deep learning approach." *IEEE Transactions on Geoscience and Remote Sensing*. 54(8), 4544-4554 (2016).
2. Qu H. C., Liang X. J., Liang S. C., Liu W. J., "Dimensionality-varied deep convolutional neural network for spectral-spatial classification of hyperspectral data," *Journal of Applied Remote Sensing*. 12(1), 016007 (2018).
3. Qian Y. T., Ye M. C., "Hyperspectral imagery restoration using nonlocal spectral-spatial structured sparse representation with noise estimation," *IEEE Journal of Selected Topics in Applied Earth Observations & Remote Sensing*. 6(2), 499-515 (2013).
4. Gu Y. F., Wang C., You D et al., "Representative multiple kernel learning for classification in hyperspectral imagery." *IEEE Transactions on Geoscience and Remote Sensing*. 50(7), 2852-2865 (2012).
5. Chan T. H., Jia K., Gao S et al., "PCANet: A simple deep learning baseline for image classification?." *IEEE Transactions on Image Processing*. 24(12), 5017-5032 (2015).
6. Chen Y. S., Lin Z. H., Zhao X et al., "Deep learning-based classification of hyperspectral data." *IEEE Journal of Selected topics in applied earth observations and remote sensing*. 7(6), 2094-2107 (2014).
7. Chen Y. S., Zhao X., Jia X. P., "Spectral-spatial classification of hyperspectral data based on deep belief network." *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*. 8(6), 2381-2392 (2015).

8. Pan B., Shi Z. W., Zhang N., "Hyperspectral image classification based on nonlinear spectral-spatial network." *IEEE Geoscience and Remote Sensing Letters*. 13(12) 1782-1786 (2016).
9. Yue J., Zhao W., Miao S., "Spectral-spatial classification of hyperspectral images using deep convolutional neural networks." *Remote Sensing Letters*. 6(6), 468-477 (2015).
10. Li Y., Zhang H., Shen Q., "Spectral-spatial classification of hyperspectral imagery with 3d convolutional neural network." *Remote Sensing*. 9(1), 67 (2017).