

R-VCANet: A New Deep Learning-Based Hyperspectral Image Classification Method

Bin Pan, Zhenwei Shi and Xia Xu

Abstract

Deep learning-based methods have displayed promising performance for hyperspectral image (HSI) classification, due to their capacity of extracting deep features from HSI. However, these methods usually require a large number of training samples. It is quite difficult for deep learning model to provide representative feature expression for HSI data when the number of samples are limited. In this paper, a novel simplified deep learning model, R-VCANet, is proposed, which achieves higher accuracy when the number of training samples is not abundant. In R-VCANet, the inherent properties of HSI data, spatial information and spectral characteristics, are utilized to construct the network. And by this means the obtained model could generate more powerful feature expression with less samples. Firstly, spectral and spatial information are combined via the rolling guidance filter (RGF), which could explore the contextual structure features and remove small details from HSI. More importantly, we have designed a new network called Vertex Component Analysis Network (VCANet) for deep features extraction from the smoothed HSI. Experiments on three popular datasets indicate that the proposed R-VCANet based method reveals better performance than some state-of-the-art methods, especially when the training samples available are not abundant.

Index Terms

Hyperspectral image classification, R-VCANet, limited samples, deep learning.

I. INTRODUCTION

Hyperspectral sensors could provide images containing hundreds of data bands with high spatial and spectral resolution. Abundant spatial and spectral information turns hyperspectral image (HSI) into a powerful tool in many fields such as geological prospecting [1], precision agriculture [2] and environment monitoring [3]. Hyperspectral image classification is one of the key technologies in HSI processing. However, HSI classification is still challenging

The work was supported by the National Natural Science Foundation of China under the Grants 61671037, the Beijing Natural Science Foundation under the Grant 4152031, the funding project of State Key Laboratory of Virtual Reality Technology and Systems, Beihang University under the Grant BUAA-VR-16ZZ-03, and the Fundamental Research Funds for the Central Universities under Grant YWF-16-BJ-J-30. (*Corresponding author: Zhenwei Shi.*)

Bin Pan, Zhenwei Shi (Corresponding Author) and Xia Xu are with Image Processing Center, School of Astronautics, Beihang University, Beijing 100191, China, and with Beijing Key Laboratory of Digital Media, Beihang University, Beijing 100191, China, and also with State Key Laboratory of Virtual Reality Technology and Systems, School of Astronautics, Beihang University, Beijing 100191, China, (e-mail: panbin@buaa.edu.cn; shizhenwei@buaa.edu.cn; xuxia@buaa.edu.cn).

due to the complex characteristics of HSI data. The large number of spectral bands may bring noise to HSI, and the high dimensionality of HSI may produce the Hughes phenomenon [4]. Therefore, using spectral signatures directly may not be suitable for the task of HSI classification [5].

During the last decade, many feature extraction methods have been proposed to handle this problem. A popular idea is reducing the dimension of HSI. In [6], principal component analysis (PCA) is discussed. Some non-linear dimension reduction methods such as manifold learning [7], [8] are also utilized for HSI classification. In [9], Sun *et al.* proposed a band selection method based on improved sparse subspace clustering. In [10], Persello *et al.* presented a kernel-based feature selection method to obtain a subset of the original hyperspectral data. To further improve the classification performance, many researchers have worked on spectral–spatial feature extraction. In [11], the extended morphological profile was proposed to combine the spectral and spatial information. In [12], spectral-spatial classification methods based on attribute profiles were surveyed. Li *et al.* developed a discontinuity preserving relaxation strategy for HSI classification [13]. Khodadadzadeh *et al.* presented a spectral-spatial classifier for HSI which specifically deals with the issue of mixed pixel [14]. A detailed overview of the spectral-spatial based HSI classification is available in [15].

Recently, deep learning methods have achieved excellent performance in many fields such as data dimensionality reduction [16] and image classification [17]. In 2014, deep learning method even achieved surpassing human-level face recognition performance [18]. Deep learning methods aim at learning the representative and discriminative features in a hierarchical-manner from the data. In [19], Zhang *et al.* provided a technical tutorial for the application of deep learning in remote sensing. In [20], Chen *et al.* employed deep learning method to handle HSI classification for the first time, where a stacked autoencoder (SAE) was adopted to extract the deep features in HSI. Based on this work, some improved autoencoder-based methods were proposed, including stacked denoising autoencoder [21], [22], stacked sparse autoencoder [23], convolutional autoencoder [24]. Research indicates that convolutional neural network (CNN) could also provide effective deep features for HSI classification. Some promising works were reported in [25], [26], [27], [28], [29]. However, deep learning-based methods usually need to train networks with complex structure, leading to time-consuming training process. In [30], Chan *et al.* proposed a simplified deep learning baseline called PCA network (PCANet). Compared with CNN, PCANet is a much simpler network. PCA filters are chosen as the convolution filter bank in each layer, a binary quantization is used as the nonlinear layer, and feature pooling layer is replaced by the block-wise histograms of the binary codes. Though simple, experiments indicate that PCANet is already quite on par with, and often better than, state-of-the-art deep learning-based features in many image classification task [30]. In [31], Pan *et al.* developed a simplified deep learning model for HSI classification based on PCANet.

However, deep learning-based methods may perform poor when the number of training samples is small [29]. In [31], the authors considered that one of the further works for deep learning-based HSI classification is reducing the required number of training samples. Generally, tens of thousands to tens of millions training samples are necessary to get a deep learning model with powerful feature representation capability [32], but this is nearly impossible for the task of HSI classification. According to the authors' knowledge, when training samples are limited, some

state-of-the-art traditional methods still outperform deep learning-based ones in several popular datasets. Though deep learning-based methods are promising, the problem of limited samples must be overcome.

In this paper, we propose a novel deep learning framework based on rolling guidance filter and vertex component analysis network (R-VCANet for short), which is able to achieve high classification accuracy with much less training samples than traditional deep learning-based methods. The key ideas of our work contain the following two aspects: First, take full advantage of the spatial information of hyperspectral data; Second, construct a network model considering the physical characteristics of HSIs. Based on the above two strategies, the R-VCANet could achieve satisfying classification accuracy while significantly reduce the number of training samples required.

Different from some computer vision tasks, in HSI classification, the spatial relativity between neighboring pixels could provide discriminative classification information. In this paper, we adopt an effective edge-preserving filter, rolling guidance filter (RGF) [33], to make full use of the spatial structure information. RGF is usually used to remove noise and small details in an image, while the overall structure of the image is preserved. Based on RGF, the spatial context information could be successfully exploited. RGF has been used by some recent studies in different ways. In [34], guidance filter is used to tackle the probabilistic map of SVM. In [35], Xia *et al.* combined independent component analysis and RGF via an ensemble strategy (E-ICA-RGF). In this paper, we utilize RGF to smooth the original HSI directly, and the result is considered as the input of the following steps. The RGF could not only reduce the noise in HSI but also extract the spatial structure information. Therefore, the smoothed HSIs by RGF could provide a powerful basis for extracting more representative features.

Moreover, we design a simplified deep learning model, VCANet, based on the work of vertex component analysis (VCA) [36] and PCANet [30]. VCA is a popular endmember extraction method, which is usually used to find *pure* materials in HSI. In VCANet, we first extract the pure materials signatures (such as alfalfa, wheat and bare soil) by VCA. Then, instead of collecting patches around pixels, we directly use extracted endmembers by VCA to generate the convolution filter bank and train a multi-layer network for HSI classification. The major motivation of this strategy is that we want to construct a network utilizing some spectral characteristics of HSI. PCANet is originally designed for single image classification, however, the target of HSI classification is giving a label for each pixel. Therefore, we improve the PCANet via adding spectral characteristics of HSI.

Combining RGF and VCANet, the proposed R-VCANet could provide more representative features with limited training samples. The major novelty of our work is summarized as follows.

- We propose a simplified deep learning-based framework, R-VCANet, which achieves promising performance in HSI classification with limited training samples. The proposed R-VCANet could contribute to the application of deep learning methods in HSI classification.
- We conduct a spectral-spatial strategy to take full use of the spatial structure information and improve the quality of the input data.
- A spectral characteristics-based network, VCANet, is constructed to extract more representative features.

The remainder of this paper is organized as follows. In Section II, we describe the proposed R-VCANet in details. Experimental results and the discussion are presented in Section III. We conclude this paper in Section IV.

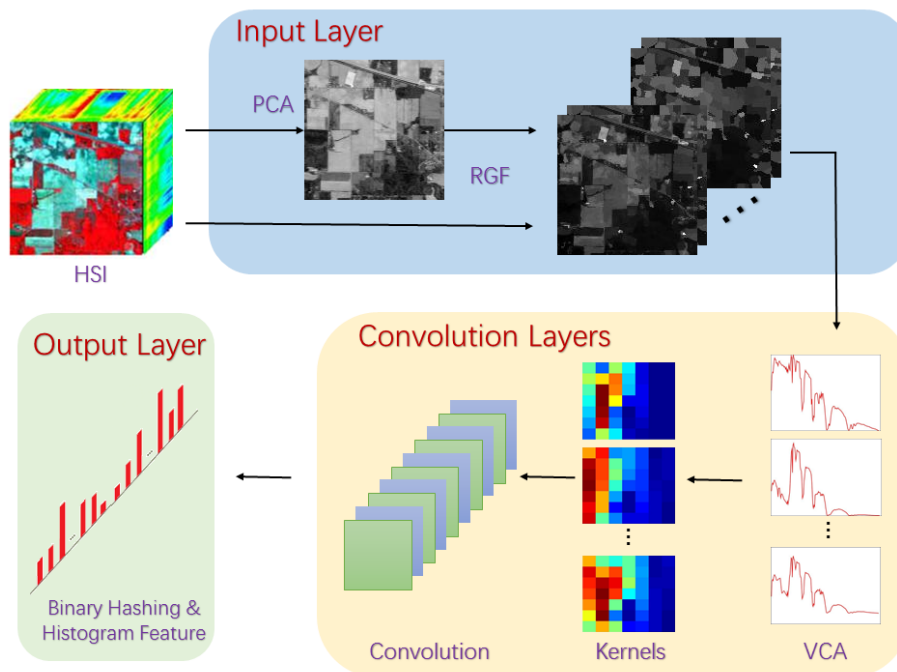


Fig. 1: Feature extraction by R-VCANet.

II. R-VCANET FOR HYPERSPECTRAL IMAGE CLASSIFICATION

Recently, deep learning-based methods have achieved promising performance in HSI classification [20]. However, a large number of training samples have to be used. Here, we develop the R-VCANet where only limited samples are necessary. R-VCANet is a simplified deep learning model, because the network structure is much simpler than some popular convolutional deep learning models such as CNN. In CNN, users have to utilize a large number of training samples to learn the parameters in convolution kernels. However, in R-VCANet, the convolution kernels are obtained by VCA. And this is also one of the most important reasons why only limited samples are necessary in R-VCANet. The R-VCANet contains two parts: RGF-based HSI smoothing and VCANet-based feature extraction. RGF is used for utilizing the joint spectral-spatial information, and VCANet is proposed to extract discriminative features. The flowchart of R-VCANet-based HSI feature extraction is shown in Fig. 1. In this section, we first describe the process of HSI smoothing by RGF, then, present the detailed structure of VCANet. At last, the HSI classification process by R-VCANet is revealed.

A. RGF-based smoothing

RGF [33] is a recently proposed edge-preserving filter, which could smooth away small textures while retaining spatial structure information. In HSIs, the neighboring pixels usually have strong relationship. Studies have shown that edge-preserving filters are effective approaches to utilize the spatial information for the task of HSI classification [34], [35]. In R-VCANet, we smooth each band of a HSI based on RGF so as to remove the spatial variability and

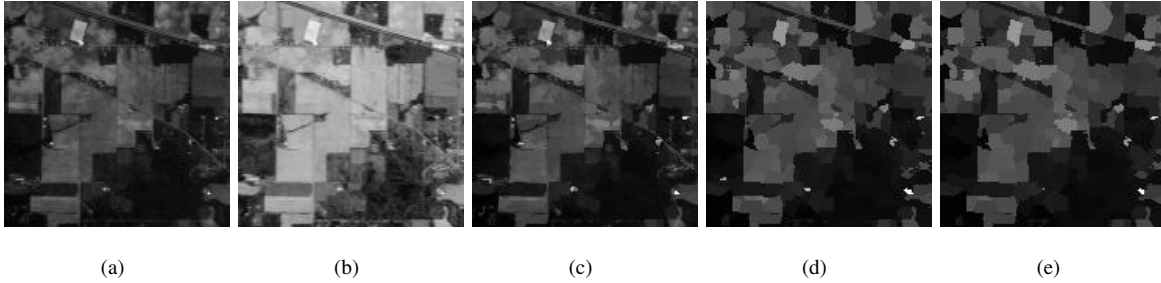


Fig. 2: Some results of RGF. (a) Input image. *Indian Pines* dataset, band 10. (b) Guidance image. (c) $T=1$. (d) $T=4$. (e) $T=8$.

image noise.

RGF is developed from guidance filter, which is based on a local linear model. The guidance filter assumes that in a local window ω_k with size $(2r + 1) \times (2r + 1)$, the output Q of a filter can be expressed by a linear transform of a guidance image:

$$Q_i = a_k G_i + b_k, \forall i \in \omega_k \quad (1)$$

where i is one of a pixel in the window ω_k , G is a guidance image, and a_k and b_k are coefficients of the linear transform. Theoretically, any image can serve as a guidance image. Here, to improve the computational efficiency, we use the first principal component of the original HSI by PCA decomposition as the guidance image. a_k and b_k can be obtained by minimize the following energy function:

$$E(a_k, b_k) = \sum ((a_k G_i + b_k - p_i)^2 + \epsilon a_k^2) \quad (2)$$

where ϵ is a parameter controlling the degree of blurring, and p denotes the input image. Running the same filtering on the output image Q (rolling), we can get the result of RGF. Some results of RGF are displayed in Fig. 2, where T denotes the rolling times.

In R-VCANet, we carry out the RGF 30 times for each band of the HSI data, and the results are considered as the input of the following VCANet. The parameter analysis is presented in the experimental section. By this means the spatial information of HSI is extracted, and the quality of input data could be improved.

B. VCANet-based feature extraction

VCANet is the key point of the proposed R-VCANet. Based on VCANet, we could extract more representative features from the smoothed HSI data. VCANet, which is also a simplified deep learning model, is developed from PCANet [30]. However, the PCANet is originally designed for single image classification (2-D data), rather than spectral vector classification (1-D vectors). Furthermore, in PCANet, the convolutional kernels are obtained from the principal components of sliding patches, which may not reflect the spectral characteristics of HSI. In the task of HSI classification, a basic assumption is that most of the pixels are not mixed. In other words, there are *pure*

materials (also called endmembers) could be observed in the image. Therefore, we attempt to utilize the materials spectra extracted from the HSI data to construct a new network. VCA is a popular endmember extraction method [36]. In VCANet, we replace the convolutional kernels in PCANet with the spectra extracted from the HSI by VCA, thus the obtained network could better embody the particularity of HSI classification. Here, we first give a brief description about PCANet and VCA, and then introduce how VCANet works.

Algorithm 1 R-VCANet for HSI feature extraction

Input Layer

RGF-based smoothing

Input: HSI data

1. Guidance filter on HSI data by Eq.(1)
2. Rolling T times
3. Data imaging by Eq.(4)

Output: Single images as samples

Convolution Layer

VCANet-based feature extraction

Input: Smoothed HSI data and imaged samples

1. Extract endmembers by VCA
2. Generate convolution kernels by Eq.(4)
3. Construct the structure of the network

Output: Convolution network

Output Layer

Feature expression

Input: Samples and the convolution network

1. Binary hashing based on Eq. (7)
2. Histogram features based on Eq. (8)

Output: Deep feature expression for each sample

1) *PCANet*: PCANet is a cascaded linear network, where four layers could be observed: The input, two convolution and output layers. The input layer contains training or testing samples. Note that all the samples are single images. Let \mathbf{I} denote an input 2-D image, \mathbf{W}_l^1 be the l th filter which is obtained by PCA decomposition, then the output of the first convolution layer can be expressed by

$$\mathbf{I}_1^l \doteq \mathbf{I} * \mathbf{W}_l^1, l = 1, 2, \dots, L_1, \quad (3)$$

where $*$ denotes 2-D convolution, and L_1 is the number of filters in the first convolution layer. Using similar strategy we can get the output of the second convolution layer. If there are L_2 filters in the second convolution layer, then for each input image \mathbf{I} , totally $L_1 \times L_2$ output images can be generated by the network. Though more

convolution layers are also available, in [30] the authors suggested that two layers were enough to achieve satisfying performance.

The output layer is composed of binary hashing for all the outputs in the second convolution layer, followed by a histogram feature extraction operation. The histogram feature is the final feature representation for an input image. At last, a linear SVM [37] is used for classification.

2) *VCA*: VCA has been widely used in endmember extraction and hyperspectral unmixing [36], [38]. Endmembers refer to the pure materials' spectra in HSIs, and endmember extraction is a process of finding the spectra of all the endmembers. The VCA algorithm begins with a randomly selected endmembers set, and then iteratively projects the HSI data onto a direction which is orthogonal to the subspace spanned by the selected endmembers. The extreme of the projection corresponds to a new endmember. The algorithm iterates for each endmember until all of them are determined.

3) *VCANet*: VCANet is developed from PCANet, which also contains four layers: The input, two convolution and output layers. Here, we improve it by redesigning the input and the two convolution layers. In PCANet, the input samples must be single images, and the convolution kernels are obtained by PCA decomposition for many sliding patches. However, compared with natural image classification, there are two special properties for HSI classification: First, the samples are 1-D spectral signatures, rather than 2-D images; Second, the spectral characteristics in HSI could contribute to the classification results. In VCANet, two strategies are proposed to tackle the above two problems respectively: data imaging and VCA-based kernels construction.

Data imaging refers to transform a spectral vector to an image. For a pixel spectrum \mathbf{x} , the data imaging operation can be expressed by

$$\mathbf{X} = \text{mat}_{m \times m}(\mathbf{x}) \in \mathbb{R}^{m \times m}, \quad (4)$$

where $\text{mat}_{m \times m}(\mathbf{x})$ is a function that maps \mathbf{x} to a matrix \mathbf{X} , and $m \times m$ denotes the size of \mathbf{X} . We set the height and width the same only for convenience. Data imaging would not destroy the physical characteristics of spectra. Instead, the spectral differences between pixels could be represented by different texture of each image.

More importantly, we replace the convolution kernels in PCANet by the endmembers spectra. VCA algorithm is conducted on the smoothed HSI data, and the extracted endmembers are used to construct the convolution kernels in the convolution layer. Let \mathbf{X}_i denotes the i th imaged spectrum in a hyperspectral image. We define

$$\mathbf{A}^1 = [\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_{L_1}] \in \mathbb{R}^{p \times L_1}, \quad (5)$$

where \mathbf{A}^1 denotes the endmembers matrix obtained by VCA in the first convolution layer, $\mathbf{a}_l \in \mathbb{R}^{p \times 1}$ is the l th endmember, p is the number of bands, then the l th convolution kernel can be expressed by

$$\mathbf{W}_l^1 \doteq \text{mat}_{k \times k}(\mathbf{m}_l) \in \mathbb{R}^{k \times k}, \quad (6)$$

where $k \times k$ is the kernel size. Using the imaged endmembers as the convolution kernel directly is not appropriated, because in this case the size of convolution kernel is the same as the input data. Therefore, a dimension reduction strategy is necessary. In [39], the authors have shown that averaging method performed well in removing redundant



Fig. 3: The convolution kernels learned by VCANet from *Indian Pines* dataset.

spectral information. Motivated by this, we reduce the spectral dimension by averaging fusion. After the dimension reduction, we also conduct data imaging based on Eq. (4) so as to generate convolution kernels. Then, the obtained kernels are used to convolve the smoothed as well as imaged HSI data. The same strategy is conducted in the next convolution layer.

Similar to PCANet [30], in the output layer, the binary hashing and histogram feature are also used in VCANet to obtain the final feature representation for each pixel. In the second convolution layer, there are totally L_1 input images, each of which generates L_2 outputs. That is to say, there are L_1 groups output images. We define \mathbf{O}^j ($j = 1, 2, \dots, L_1$) as the j th group. For each pixel, viewing the vector of L_2 binary bits as a decimal number, we can convert each \mathbf{O}^j to a single image:

$$\mathbf{T}^j \doteq \sum_{\ell=1}^{L_2} 2^{\ell-1} H(\mathbf{O}^j), \quad (7)$$

where $H(\cdot)$ is a Heaviside step function whose value is 1 for positive inputs and 0 otherwise, \mathbf{T}^j is an integer-valued image with pixel value in the range $[0, 2^{L_2}-1]$. Partitioning each \mathbf{T}^j into B blocks, then we extract the histogram feature in each block and combine all the B histograms into a single vector, denoted by $\text{Bhist}(\mathbf{T}^j)$. At last, the final feature can be expressed by:

$$\mathbf{f} \doteq [\text{Bhist}(\mathbf{T}^1), \text{Bhist}(\mathbf{T}^2), \dots, \text{Bhist}(\mathbf{T}^{L_1})]. \quad (8)$$

After the feature extraction process by VCANet, each pixel \mathbf{x} in a hyperspectral image is transformed it into a new feature space, and represented by \mathbf{f} .

C. R-VCANet for classification

The overall structure of the R-VCANet is shown in Fig. 1. The spectral and spatial information are combined in the input layer. The convolution layers, where convolution kernels are extracted from the smoothed HSI by based on VCA, are used for deep feature extraction. And the final feature representation is obtained in the output layer. At last, a linear SVM with regularization parameter set as 1 is conducted to get the classification results. Note that the linear SVM is also adopted in [30]. We also give a pseudocode to describe the structure and working process of the proposed R-VCANet, as shown in Algorithm 1.

III. EXPERIMENTAL RESULTS

Experimental results are shown in this section. We first give an introduction about the datasets used in experiments. Then, the influence of parameters in R-VCANet is analyzed. Finally, the experimental results are shown and discussed, by comparing with some related state-of-the-art methods. Three widely used metrics, namely, overall

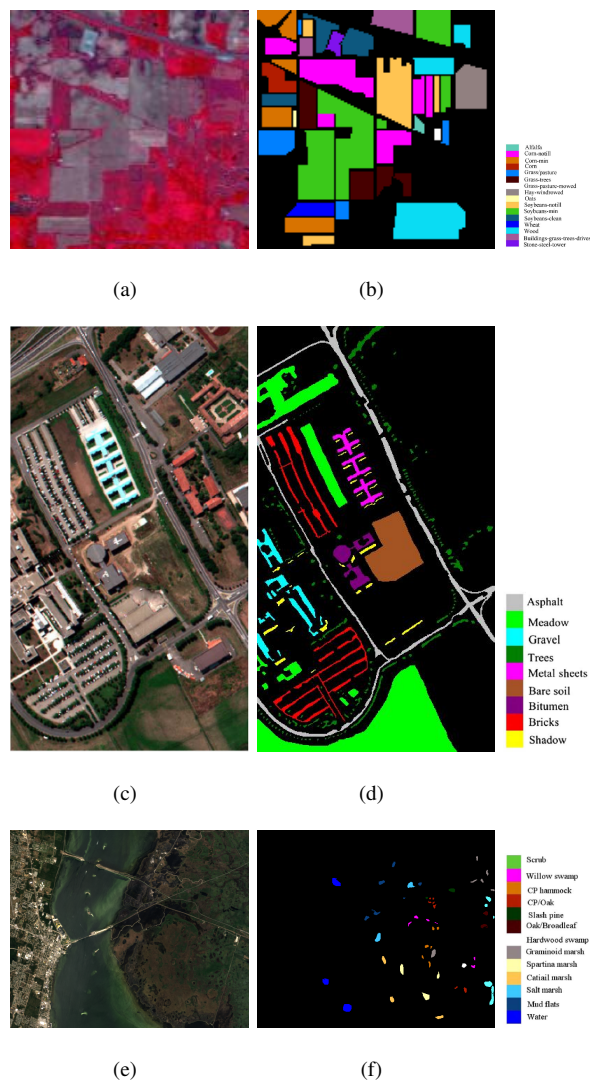


Fig. 4: Three test datasets and corresponding groundtruths. (a) False color composite image (R-G-B=band 50-27-17) for *Indian Pines* dataset. (b) The groundtruth image with 16 land-cover classes. (c) False color composite image (R-G-B=band 10-27-46) for *Pavia University* dataset. (d) The groundtruth image with 9 land-cover classes. (e) False color composite image (R-G-B=band 28-9-10) for *KSC* dataset. (f) The groundtruth image with 13 land-cover classes.

accuracy (OA), average accuracy (AA) and Kappa coefficient (κ) are adopted as the evaluation criteria. Three popular HSI datasets are used to evaluate the performance of R-VCANet and some state-of-the-art methods: *Indian Pines*, *Pavia University* and *Kennedy Space Center (KSC)*¹. Specially, because we try to handle the problem that

¹All of them are available online: http://www.ehu.es/ccwintco/index.php?title=Hyperspectral_Remote_Sensing_Scenes

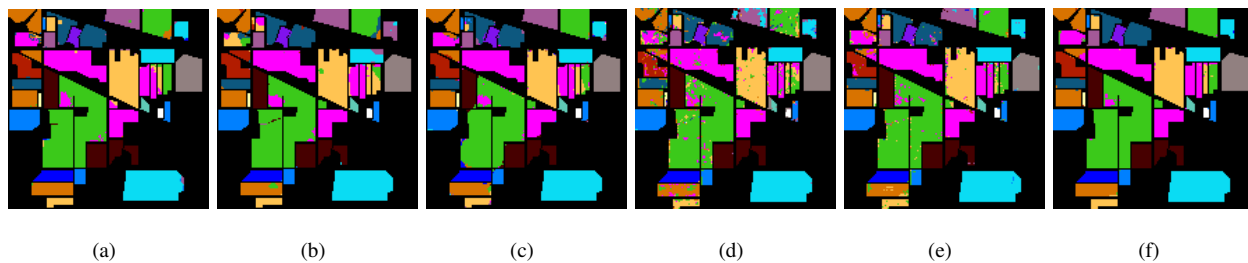


Fig. 5: Classification results by different methods on *Indian Pines* dataset. (a) IFRF (b) EPF-G (c) E-ICA-RGF (d) NSSNet (e) R-PCANet and (f) R-VCANet.

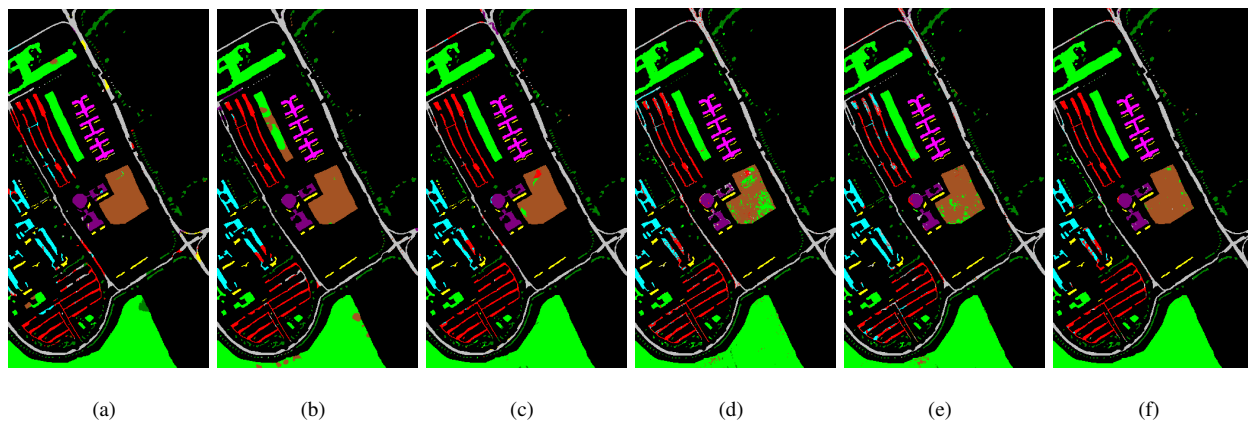


Fig. 6: Classification results by different methods on *Pavia University* dataset. (a) IFRF (b) EPF-G (c) E-ICA-RGF (d) NSSNet (e) R-PCANet and (f) R-VCANet.

lots of training samples are essential in deep learning-based methods, here we only randomly selected 10% (*Indian Pines*), 1% (*Pavia University*) and 3% (*KSC*) of all labeled pixels in each class for training, and the others are used for testing. The detailed information about the number of samples is shown in Table I-III. Although this ratios may be still large for some traditional methods [40], it has been reduced significantly, compared with some deep learning-based methods such as SAE-LR [20], DBN-LR [41] and NSSNet [31], where half of all labeled pixels are used for training. We also implement statistical evaluation to verify that the improvements in results are significant. In addition, some important parameters, rolling times T , kernels number K and kernels size n , are set as 30, 8 and 7, respectively.

A. Datasets

- *Indian Pines* dataset was acquired by airborne visible/infrared imaging spectrometer (AVIRIS) in Northwestern Indiana, with 145×145 pixels size and 20m spatial resolution. The wavelength ranges from 0.4 to 2.5 μm . The bands covering the region of water absorption are removed from this image, and 200 spectral bands remain. The groundtruth data are composed of 10249 labeled pixels which classified into 16 classes. Fig. 4(a)(b) provide

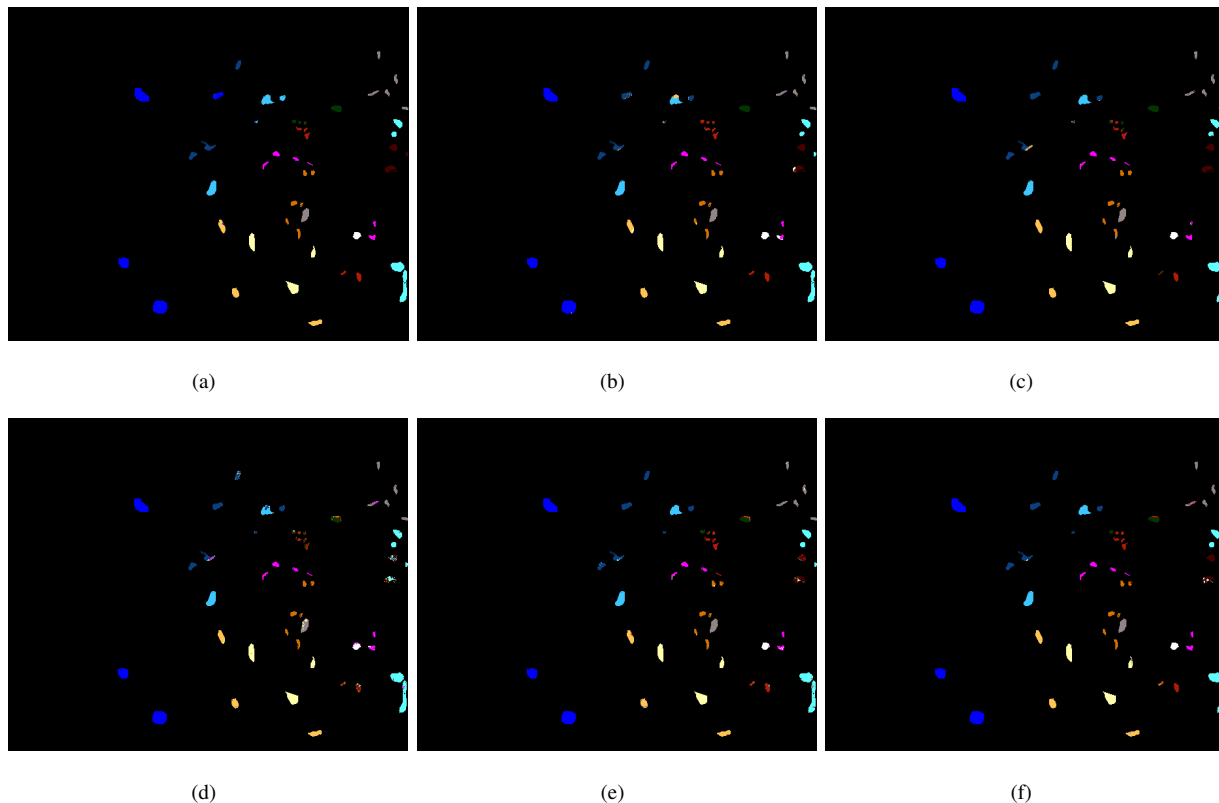


Fig. 7: Classification results by different methods on *KSC* dataset. (a) IFRF (b) EPF-G (c) E-ICA-RGF (d) NSSNet (e) R-PCANet and (f) R-VCANet.

a false color composite image and the groundtruth for this data.

- *Pavia University* image was collected by reflective optics system imaging spectrometer (ROSIS-3) sensor over the city of Pavia, Italy. The size of this image is 610×340 with 1.3m spatial resolution, and 103 bands are preserved after removing the noise bands. There are totally 42776 labeled pixels available in the ground truth, containing 9 different classes. A false color composite image and the groundtruth image are shown in Fig. 4(c)(d). Compared with Indian Pines dataset, this dataset have more labeled samples and higher spatial resolution.
- *Kennedy Space Center (KSC)* image was collected by AVIRIS in 1996. It contains 176 bands after removing water absorption and low SNR bands, ranging from 0.4 to 2.5 μm wavelength. The *KSC* data have 18m spatial resolution and 512×614 pixels size. Totally 5211 labeled pixels belonging to 13 land-cover classes are observed in the groundtruth image. Fig. 4(e)(f) give a false color composite image and the groundtruth for this data.

B. Parameter Analysis

In R-VCANet, the number of training samples is an important concern. We present some experiments about the effect of training samples. Furthermore, since our method is developed from PCANet, all the parameters in

PCANet are also observed in R-VCANet. However, in [30], the authors have provided a detailed analysis about the parameters in PCANet, and demonstrated that the influence of different parameters is limited. Therefore, we mainly focus on the discussion about the particular parameters in R-VCANet: the rolling times T , the number of convolution kernels K and the kernels size n . In this section, OA or κ is selected as the metric. All of the results here are obtained by the average value of running 30 times.

1) *Training samples*: Deep learning-based methods could achieve excellent performance when training samples are abundant. However, in HSI classification, the training samples are limited. In [31] and [41], 50% of all labeled pixels are selected as training samples.

Fig. 8 shows the influence of training samples on *Indian Pines* dataset. We use *Indian Pines* dataset here because the number of training samples used in this dataset is larger than the other, so it is more easy to depict the tendency. R-VCANet, IFRF and E-ICA-RGF present the best performance among all the compared methods, so they are displayed in Fig. 8. This results are obtained by the average value of 30 times running. We note that all of them have achieved above 92% κ with only 4% labeled pixels for training. When the ratio of training samples is below 10%, the R-VCANet slightly outperforms the others. Continue increasing the number of training samples could further improve the accuracies, but it is not obvious. Therefore, we may conclude that 10% is enough to learning a powerful classification model for *Indian Pines* dataset. Fig. 8 indicates that compared with traditional deep learning models the R-VCANet could achieve good results with less samples.

2) *Rolling times*: RGF is the first step of R-VCANet. More rolling times could generate smoothed data, as depicted in Fig. 2. However, too many rolling times may lead to information loss, as well as the increasing of computing cost. Therefore, selecting an appropriate rolling times T is necessary. Fig. 9 shows the influence of rolling times on AA in the three datasets. We can see that there is a significant increase after using RGF ($T = 1 - 10$). When T is above 10, the AA in all the three datasets tend to keep stable. However, there is still 1% around improvement can be observed when T is set as 30. Though more rolling times may contribute to better results, the improvement is slight. Therefore, we may draw the conclusion that $T = 30$ is an available setting, since continuing increasing T contributes little to the accuracies.

3) *Kernels number*: The number of convolution kernels could also affect the final results. According to the structure of the R-VCANet, more convolution kernels could generate higher dimensional features and higher computing complexity. Different from other parameters, there is not a sharp increase at first in Fig. 10. Instead, it is observed that the accuracies keep slow growth until the kernels number is up to 6. Subsequently, the three metrics seem stable, and only 0.5% around variation could be observed. Continuing adding convolution kernels could slightly contribute to the results, but the computing cost, especially for RAM, will be unacceptable. Therefore, in R-VCANet, we set the number of convolution kernels as 8.

4) *Kernels size*: After extracting the endmembers by VCA, we should implement dimension reduction so as to obtain the convolution kernels. Here, we set the kernels size from 3×3 to 9×9 to analyse the difference, as shown in Fig. 11. We note that after a sharp increase at from 3×3 to 4×4 , the values of κ present steady tendency. Even when the kernel size is set as 3×3 , the κ are still above 94%. Since there are only 103 bands available in *Pavia*

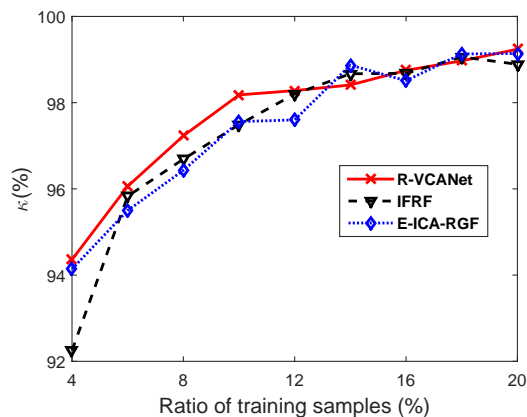


Fig. 8: The influence of training samples number on the *Indian Pines* dataset.

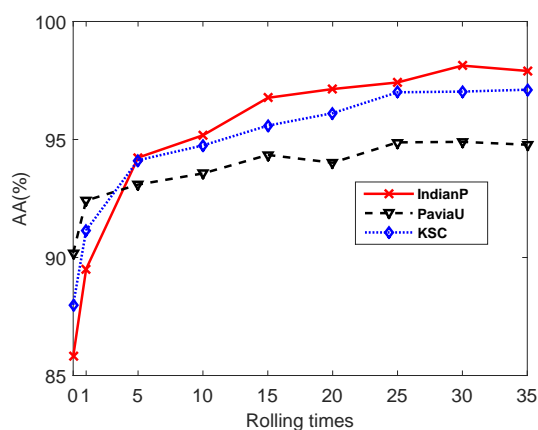


Fig. 9: The influence of rolling times on the three datasets.

University dataset, the convolution operation does not work if the kernels size were too large. This experiment indicates that kernels size is not an important impact factor in R-VCANet. Actually, from these experiments we may come to a conclusion that the network parameters have little influence to the final classification results.

C. Compared Methods

We compare the R-VCANet with some related and state-of-the-art HSI classification methods: IFRF [39], EPF-G [34], E-ICA-RGF [35], SAE-LR [20] and NSSNet [31]. All the compared methods were designed for HSI classification, and were proposed in recent years. IFRF is a classical method where spatial and spectral information is combined via image fusion and recursive filtering. EPF-G is a joint spectral-spatial HSI classification method, where edge-preserving filters (guidance filter) are used to extract the spatial structure information. E-ICA-RGF is a recently developed HSI classification algorithm where RGF is introduced for the first time. SAE-LR is a representative deep learning-based HSI classification method. Note that because deep learning-based methods may

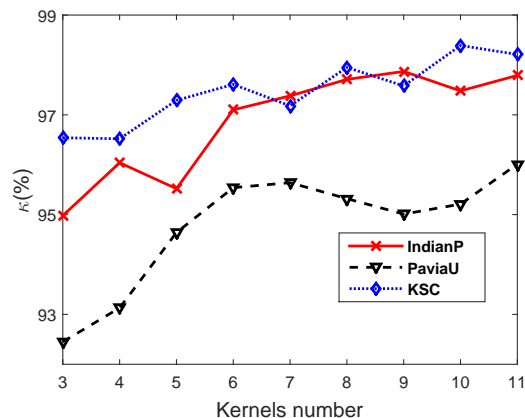


Fig. 10: The influence of convolution kernels number on the three datasets.

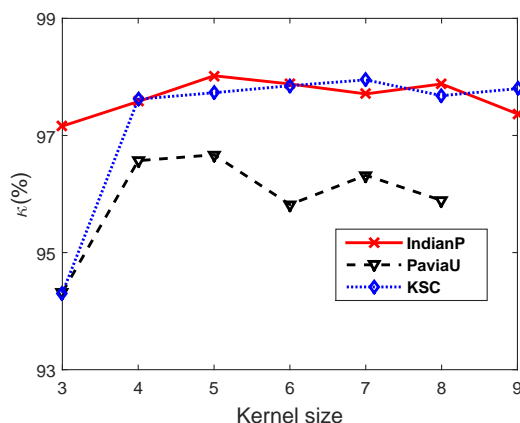


Fig. 11: The influence of kernels size on the three datasets.

perform poor when training samples are not enough, in the comparison experiments the number of training samples in SAE-LR is set as 60% of all labeled pixels for *indian pines dataset*. SAE-LR are not compared in the other datasets, because the training samples are too small. NSSNet is also a simplified deep learning model which has presented promising performance. Furthermore, since the major contribution of the proposed method is the VCANet which is developed from PCANet, to verify its effectiveness, we supplement another experiment: using the results of RGF and the original PCANet [30] (R-PCANet for short). We adopt the default parameters of the compared methods which were presented in the corresponding references. The number of training sample of the above methods (except SAE-LR) is set as 10%, 1% and 3% of all labeled pixels for *Indian Pines*, *Pavia University* and *KSC* datasets, respectively.

TABLE I: CLASSIFICATION ACCURACIES OF DIFFERENT METHODS ON *INDIAN PINES* DATASET.

Class	Samples		Methods						
	Train	Test	IFRF	EPF-G	E-ICA-RGF	SAE-LR	NSSNet	R-PCANet	R-VCANet
Alfalfa	5	41	96.00±2.63	95.85±11.2	98.66±2.07	94.79±2.55	48.94±12.6	93.00±5.11	98.94±1.65
Corn-notill	143	1285	95.29±2.13	93.95±3.08	95.78±1.97	90.41±0.94	83.58±1.87	90.32±1.92	95.34±1.68
Corn-mintill	83	747	96.03±2.64	96.25±2.95	95.93±1.98	87.08±1.14	79.88±3.51	91.63±2.72	96.17±1.52
Corn	24	213	94.82±3.75	67.00±9.15	99.27±1.32	89.32±3.60	76.80±6.18	89.82±4.85	97.38±2.87
Grass-pasture	48	435	97.77±2.90	98.17±1.25	97.67±1.62	97.31±1.63	92.68±2.46	93.54±2.43	97.80±1.65
Grass-trees	73	657	98.78±0.58	97.97±1.12	98.74±0.68	96.85±0.96	99.15±0.55	99.11±0.61	99.83±0.17
Grass-pasture -mowed	3	25	96.18±12.2	100.0±0.00	99.28±2.17	78.57±0.01	62.80±17.7	90.80±8.92	96.00±5.35
Hay-windrowed	48	430	100.0±0.00	99.99±0.04	99.93±0.13	99.82±0.42	99.95±0.11	99.28±0.73	99.98±0.05
Oats	2	18	90.52±13.5	99.14±3.41	97.62±6.31	72.21±9.29	59.25±16.0	89.07±13.2	96.29±6.41
Soybean-notill	97	975	94.97±1.85	80.85±4.36	96.25±2.17	92.79±1.14	82.86±2.18	89.53±2.05	96.13±1.49
Soybean-mintill	246	2209	98.11±1.27	95.32±2.08	95.80±1.51	93.95±0.65	88.87±1.21	94.18±1.16	98.71±0.76
Soybean-clean	59	534	96.79±2.02	87.23±6.66	95.61±1.60	84.92±3.65	84.76±2.89	92.13±2.72	96.90±1.74
Wheat	21	184	96.90±2.42	100.0±0.00	99.73±0.44	98.75±0.00	99.34±0.48	98.85±0.72	99.58±0.42
Woods	127	1138	99.90±0.32	99.25±0.92	98.99±0.75	97.31±0.16	98.07±0.81	98.99±0.57	99.83±0.14
Buildings-Grass -Trees-Drives	39	347	94.90±3.27	78.80±6.70	99.35±0.45	69.39±2.63	71.38±3.78	89.70±3.58	98.58±1.20
Stone-Steel- Towers	9	84	95.82±5.74	87.36±5.49	99.42±3.14	97.46±1.24	93.13±4.49	96.34±2.23	99.08±1.11
OA(%)			97.21±0.44	92.43±1.18	97.00±0.43	92.42±0.27	88.22±0.50	93.86±0.47	97.90±0.32
AA(%)			96.42±1.29	92.32±1.27	98.01±0.54	90.06±0.52	82.59±1.68	93.52±1.06	97.91±0.58
$\kappa \times 100$			96.78±0.51	91.33±1.35	96.54±0.50	91.34±0.30	86.54±0.57	93.01±0.53	97.60±0.37

D. Results and Discussion

Fig. 5-7 present the visual results of all the compared methods. Though some mistakes are still observed, the overall performance of R-VCANet is good. Table I-III display the objective evaluation of the proposed and compared methods in the three datasets. Three popular metrics, OA, AA and κ , are used to give quantitative evaluation for different methods.

1) *Results on Indian Pines dataset:* In this dataset, all the compared methods have shown close results, and the R-VCANet outperforms other methods slightly. Because the number of training samples in this dataset is relatively large, this results indicate that though R-VCANet is a simplified deep learning model, it still possesses the most important characteristics of traditional deep learning methods: Abundant training samples will lead to better performance. Moreover, our method has shown better performance than SAE-LR, while the number of training samples used in R-VCANet is just 1/6 of that used in SAE-LR. Experiments on this dataset demonstrate that the R-VCANet could reduce the required samples effectively, compared with other deep learning-based methods.

TABLE II: CLASSIFICATION ACCURACIES OF DIFFERENT METHODS ON PAVIA UNIVERSITY DATASET.

Class	Samples		Methods					
	Train	Test	IFRF	EPF-G	E-ICA-RGF	NSSNet	R-PCANet	R-VCANet
Asphalt	66	6565	91.47±3.27	97.35±1.94	92.59±3.22	95.22±1.03	90.28±2.05	94.73±1.78
Meadows	186	18463	98.98±0.45	98.54±0.77	97.03±2.32	98.62±0.47	98.58±0.76	99.71±0.19
Gravel	21	2078	87.18±4.81	93.19±6.24	93.12±2.75	73.82±4.51	84.68±5.04	89.33±5.25
Trees	31	3033	88.81±8.17	87.48±10.1	91.77±1.60	90.41±1.77	89.88±2.28	90.38±3.04
Painted metal sheets	13	1332	99.73±0.43	96.77±3.27	99.04±0.60	99.85±0.13	99.75±0.32	99.89±0.15
Bare	50	4979	94.68±4.09	83.85±8.33	97.38±2.11	74.63±4.21	88.04±3.64	96.81±2.21
Bitumen	13	1317	90.19±3.67	88.23±9.07	97.68±1.30	86.09±3.42	89.25±6.64	93.68±3.41
Self-Blocking Bricks	37	3645	85.19±4.82	91.01±3.53	94.96±2.02	88.83±3.48	86.79±4.44	95.09±1.79
Shadows	9	938	77.24±10.5	99.06±0.86	91.14±2.43	97.10±1.99	95.13±3.15	97.06±2.47
OA(%)			93.73±1.46	93.86±1.76	95.59±1.10	92.19±0.83	93.37±0.87	96.77±0.91
AA(%)			90.38±2.35	92.83±1.95	94.97±0.58	89.40±1.19	91.38±1.34	95.19±1.29
$\kappa \times 100$			91.74±1.89	91.96±2.25	94.17±1.43	89.56±1.13	91.21±1.16	95.71±1.21

TABLE III: CLASSIFICATION ACCURACIES OF DIFFERENT METHODS ON KSC DATASET.

Class	Samples		Methods					
	Train	Test	IFRF	EPF-G	E-ICA-RGF	NSSNet	R-PCANet	R-VCANet
Scrub	20	741	97.71±4.19	99.23±1.47	98.17±1.96	98.07±1.29	97.74±1.56	99.36±0.74
Willow swamp	6	237	98.59±5.10	99.29±1.65	96.67±3.60	92.03±6.96	80.24±9.58	94.08±8.70
CP hammock	10	246	97.05±7.43	97.33±4.52	96.15±4.56	85.48±8.99	89.28±5.63	96.35±2.73
Slash pine	5	247	97.67±3.78	85.67±10.9	82.61±10.6	54.30±14.4	78.04±6.98	87.84±9.64
Oak/Broadleaf	3	158	92.41±8.55	88.46±14.8	89.99±7.56	58.93±11.2	77.00±6.50	90.51±9.66
Hardwood	7	222	85.95±11.2	92.79±10.6	96.47±3.57	47.71±14.0	71.44±9.19	93.42±4.89
Swamp	4	101	83.70±6.02	77.05±13.8	97.82±2.90	66.76±20.6	86.79±18.3	99.57±1.66
Graminoid marsh	13	418	93.49±8.21	87.77±10.8	93.45±6.82	85.86±6.15	94.73±5.21	98.96±1.55
Spartina marsh	19	501	99.49±2.74	95.63±5.28	97.34±4.18	97.62±3.31	98.55±2.10	97.97±4.81
Cattail marsh	16	388	100.0±0.00	95.16±7.29	99.36±1.17	98.87±1.07	95.11±2.84	99.97±0.10
Salt marsh	19	400	94.85±5.94	96.52±4.21	98.38±1.81	91.70±4.70	97.38±3.08	99.62±0.80
Mud flats	13	490	90.19±13.5	97.24±5.14	96.01±3.10	90.83±5.02	94.78±3.86	99.44±0.73
Water	21	906	100.0±0.00	99.98±0.08	100.0±0.00	100.0±0.00	99.76±0.44	100.0±0.00
OA(%)			95.70±2.34	94.96±2.22	96.69±0.99	89.14±2.60	93.21±0.81	97.90±0.66
AA(%)			95.40±1.54	93.24±3.08	95.57±1.15	82.17±4.32	89.30±1.79	96.70±1.10
$\kappa \times 100$			95.21±2.60	94.38±2.48	96.32±1.10	87.87±2.92	92.44±0.91	97.66±0.74

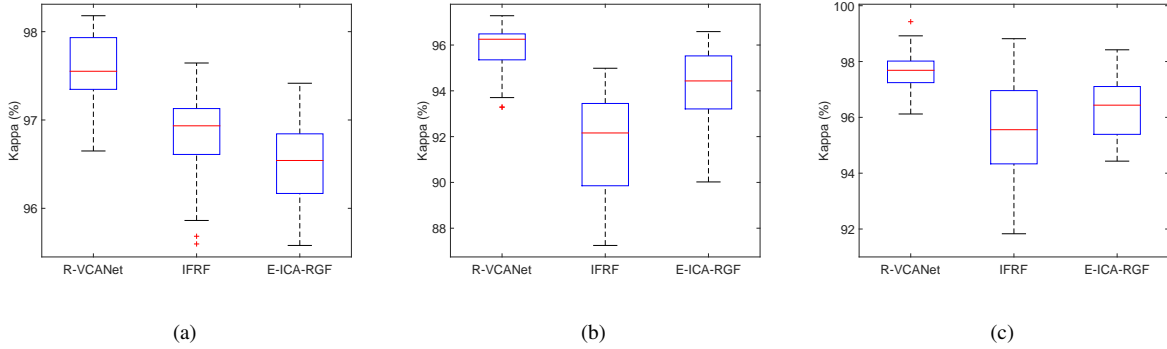


Fig. 12: Box plot of κ of different methods on (a) *Indian Pines*, (b) *Pavia University* and (c) *KSC* datasets. The center line is the median value, the edges of the box are the 25th and 75th percentiles, the whiskers extend to the most extreme points, and the abnormal outliers are plotted by “+”.

2) *Results on Pavia University dataset*: 1% of all labeled pixels are used for training in this dataset. Samples imbalance still exists, which lead to the situation that OA is higher than AA in all the methods. R-VCANet surpasses NSSNet by 4%-5% percentage in all the three metrics. SAE-LR method is removed in this experiment, because it works poor with such few training samples. Compared with three traditional methods (IFRF, EPF-G and E-ICA-RGF), R-VCANet achieves about 1% advantage in OA, and more in AA and κ . This results may indicate that compared with some state-of-the-art methods the proposed model is also predominant.

3) *Results on KSC dataset*: Only 156 samples are used for training in this dataset, which is the least among all of the three. However, the R-VCANet works well. IFRF and E-ICA-RGF present close performance to R-VCANet, but more than 1.5% gap could be still observed. In addition, the accuracy for each class is important as well to judge on the performance of the proposed method, and we can see that there are only several training samples in some classes (*e.g.*, Swamp and Oak). Except R-VCANet, all the other compared methods have shown lower than 85% accuracy in one or several classes. By comparison, the lowest accuracy of R-VCANet is 87.84% in Slash pine, and all the other classes achieve higher than 90% accuracy. Furthermore, comparison with R-PCANet could verify the effectiveness of the proposed feature extraction strategy, VCANet. And similar results could also be observed in the other two experiments.

4) *Statistical evaluation about the results*: To further validate whether the observed increase in κ is statistically significant, we use paired t-test to show the statistical evaluation about the results. T-test is popular in many related works [20], [41], [31]. We accept the hypothesis that the mean κ of R-VCANet is larger than a compared method only if Eq. (9) is valid:

$$\frac{(\bar{a}_1 - \bar{a}_2)\sqrt{n_1 + n_2 - 2}}{\sqrt{(\frac{1}{n_1} + \frac{1}{n_2})(n_1 s_1^2 + n_2 s_2^2)}} > t_{1-\alpha}[n_1 + n_2 - 2], \quad (9)$$

where \bar{a}_1 and \bar{a}_2 are the means of κ of R-VCANet and a compared method, s_1 and s_2 are the corresponding standard deviations, n_1 and n_2 are the number of realizations of experiments reported which is set as 30 in this

paper. E-ICA-RGF and IFRF are selected for evaluation, because they present the closest results to R-VCANet. Paired t-test shows that the increases on κ are statistically significant in all the three datasets (at the level of 95%), and it can be also observed in Fig. 12.

Overall, the experiments on the three popular datasets could imply that the proposed method is an effective deep learning-based HSI classification method, while only limited training samples are required.

IV. CONCLUSION

Deep learning models have been discussed in recent research for the task of HSI classification. These methods can extract the deep feature from original HSI data, and have presented promising performance. However, to achieve satisfying results, a large number of training samples are necessary. In this paper, a simplified deep learning model, R-VCANet, is proposed to overcome this problem. We utilize the inherent properties of HSI data, namely spatial contextual information and spectral characteristics, to improve the feature expression capacity of the network.

The R-VCANet contains four layers: The input layer, two convolution layers and the output layer. In the input layer, RGF is used to combine the spectral and spatial information of the original HSI data. Based on the result of RGF, we design the convolution layers to explore the deep information in the HSI data. At last, an output layer is used to determine the feature expression for each pixel.

We have conducted some experiments on three popular datasets for parameters analysis and comparison with other methods. We may conclude based on the experimental results that the R-VCANet is a promising approach to handle the HSI classification task with limited training samples. Moreover, the parameters analysis and discussion indicate that the R-VCANet is not very sensitive to parameters variation.

Although the R-VCANet has much simpler structure than classical convolutional deep learning model, it is still a time-consuming model, compared with some traditional methods. In the future work, we will focus on further simplifying the network structure, at the same time, improving the classification accuracies.

V. ACKNOWLEDGEMENT

The authors would like to thank Prof. Junshi Xia, Prof. Yi Ma, Prof. Yushi Chen and Dr. Xudong Kang for sharing their codes. The authors would also like thank the Associate Editor and six anonymous reviewers for the very insightful comments and suggestions which have significantly improved the quality of this work.

REFERENCES

- [1] F. D. V. D. Meer, H. M. A. V. D. Werff, F. J. A. V. Ruitenbeek, C. A. Hecker, W. H. Bakker, M. F. Noomen, M. V. D. Meijde, E. J. M. Carranza, J. B. D. Smeth, and T. Woldai, "Multi- and hyperspectral geologic remote sensing: A review," *International Journal of Applied Earth Observation and Geoinformation*, vol. 14, no. 1, pp. 112–128, 2012.
- [2] C. Zhang and J. M. Kovacs, "The application of small unmanned aerial systems for precision agriculture: a review," *Precision Agriculture*, vol. 13, no. 6, pp. 693–712, 2012.
- [3] B. Pan, Z. Shi, Z. An, and Z. Jiang, "A novel spectral-unmixing-based green algae area estimation method for goci data," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 2016.

- [4] G. Hughes, "On the mean accuracy of statistical pattern recognizers," *IEEE Transactions on Information Theory*, vol. 14, no. 1, pp. 55–63, 1968.
- [5] X. Zhang, Y. Liang, Y. Zheng, and J. An, "Hierarchical discriminative feature learning for hyperspectral image classification," *IEEE Geoscience and Remote Sensing Letters*, vol. 13, no. 4, pp. 594–598, 2016.
- [6] S. Prasad and L. M. Bruce, "Limitations of principal components analysis for hyperspectral target recognition," *IEEE Geoscience and Remote Sensing Letters*, vol. 5, no. 4, pp. 625–629, 2008.
- [7] C. M. Bachmann, T. L. Ainsworth, and R. A. Fusina, "Improved manifold coordinate representations of large-scale hyperspectral scenes," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 44, no. 10, pp. 2786–2803, 2006.
- [8] B. Du, L. Zhang, T. Chen, and K. Wu, "A discriminative manifold learning based dimension reduction method for hyperspectral classification," *International Journal of Fuzzy Systems*, vol. 14, no. 2, pp. 272–277, 2012.
- [9] W. Sun, L. Zhang, B. Du, and W. Li, "Band selection using improved sparse subspace clustering for hyperspectral imagery classification," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 8, no. 6, pp. 2784–2797, 2015.
- [10] C. Persello and L. Bruzzone, "Kernel-based domain-invariant feature selection in hyperspectral images for transfer learning," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 54, no. 5, pp. 2615–2626, 2016.
- [11] J. A. Benediktsson, J. A. Palmason, and J. R. Sveinsson, "Classification of hyperspectral data from urban areas based on extended morphological profiles," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 43, no. 3, pp. 480–491, 2005.
- [12] B. J. A. Ghamisi P, Dalla Mura M, "A survey on spectral-spatial classification techniques based on attribute profiles," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 53, no. 5, pp. 2335–2353, 2015.
- [13] J. Li, M. Khodadadzadeh, A. Plaza, and X. Jia, "A discontinuity preserving relaxation scheme for spectral-spatial hyperspectral image classification," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 9, no. 2, pp. 625–639, 2016.
- [14] M. Khodadadzadeh, J. Li, A. Plaza, H. Ghassemian, J. M. Bioucas-Dias, and X. Li, "Spectral-spatial classification of hyperspectral data using local and global probabilities for mixed pixel characterization," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 52, no. 10, pp. 6298–6314, 2014.
- [15] M. Fauvel, Y. Tarabalka, J. A. Benediktsson, J. Chanussot, and J. C. Tilton, "Advances in spectral-spatial classification of hyperspectral images," *Proceedings of the IEEE*, vol. 101, no. 3, pp. 652–675, 2013.
- [16] G. E. Hinton and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks." *Science*, vol. 313, no. 5786, pp. 504–507, 2006.
- [17] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Advances in Neural Information Processing Systems*, vol. 25, no. 2, p. 2012, 2012.
- [18] Y. Sun, D. Liang, X. Wang, and X. Tang, "Deepid3: Face recognition with very deep neural networks," *Computer Science*, 2015.
- [19] L. Zhang, L. Zhang, and B. Du, "Deep learning for remote sensing data: A technical tutorial on the state of the art," *IEEE Geoscience and Remote Sensing Magazine*, vol. 4, no. 2, pp. 22–40, 2016.
- [20] 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 and Remote Sensing*, vol. 7, no. 6, pp. 2094–2107, 2014.
- [21] X. Ma, J. Geng, and H. Wang, "Hyperspectral image classification via contextual deep learning," *EURASIP Journal on Image and Video Processing*, vol. 2015, no. 1, pp. 1–12, 2015.
- [22] Y. Liu, G. Cao, Q. Sun, and M. Siegel, "Hyperspectral classification via deep networks and superpixel segmentation," *International Journal of Remote Sensing*, vol. 36, no. 13, pp. 3459–3482, 2015.
- [23] C. Tao, H. Pan, Y. Li, and Z. Zou, "Unsupervised spectral-spatial feature learning with stacked sparse autoencoder for hyperspectral imagery classification," *IEEE Geoscience and Remote Sensing Letters*, vol. 12, no. 12, pp. 2438–2442, 2015.
- [24] 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, no. 13, pp. 3368–3379, 2015.
- [25] K. Makantasis, K. Karantzalos, A. Doulamis, and N. Doulamis, "Deep supervised learning for hyperspectral data classification through convolutional neural networks," in *Geoscience and Remote Sensing Symposium (IGARSS), 2015 IEEE International*. IEEE, 2015, pp. 4959–4962.
- [26] W. Hu, Y. Huang, L. Wei, F. Zhang, and H. Li, "Deep convolutional neural networks for hyperspectral image classification," *Journal of Sensors*, vol. 2015, 2015.

- [27] H. Liang and Q. Li, "Hyperspectral imagery classification using sparse representations of convolutional neural network features," *Remote Sensing*, vol. 8, no. 2, p. 99, 2016.
- [28] A. Romero, C. Gatta, and G. Camps-Valls, "Unsupervised deep feature extraction for remote sensing image classification," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 54, no. 3, pp. 1349–1362, 2016.
- [29] Y. Chen, H. Jiang, C. Li, X. Jia, and P. Ghamisi, "Deep feature extraction and classification of hyperspectral images based on convolutional neural networks," 2016.
- [30] T. H. Chan, K. Jia, S. Gao, J. Lu, Z. Zeng, and Y. Ma, "Pcanet: A simple deep learning baseline for image classification?" *IEEE Transactions on Image Processing A Publication of the IEEE Signal Processing Society*, vol. 24, no. 12, pp. 5017–5032, 2015.
- [31] B. Pan, Z. Shi, N. Zhang, and S. Xie, "Hyperspectral image classification based on nonlinear spectral–spatial network," *IEEE Geoscience and Remote Sensing Letters*, 2016.
- [32] J. Deng, W. Dong, R. Socher, L. J. Li, K. Li, and F. F. Li, "Imagenet: A large-scale hierarchical image database," 2009, pp. 248–255.
- [33] Q. Zhang, X. Shen, L. Xu, and J. Jia, "Rolling guidance filter," in *European Conference on Computer Vision*. Springer, 2014, pp. 815–830.
- [34] X. Kang, S. Li, and J. A. Benediktsson, "Spectral–spatial hyperspectral image classification with edge-preserving filtering," *IEEE transactions on geoscience and remote sensing*, vol. 52, no. 5, pp. 2666–2677, 2014.
- [35] J. Xia, L. Bombrun, T. Adali, Y. Berthoumieu, and C. Germain, "Spectral–spatial classification of hyperspectral images using ica and edge-preserving filter via an ensemble strategy," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 54, no. 8, pp. 4971–4982, 2016.
- [36] J. M. P. Nascimento and J. M. B. Dias, "Vertex component analysis: a fast algorithm to unmix hyperspectral data," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 43, no. 4, pp. 898–910, 2005.
- [37] R. E. Fan, K. W. Chang, C. J. Hsieh, X. R. Wang, and C. J. Lin, "LIBLINEAR: A library for large linear classification," *Journal of Machine Learning Research*, vol. 9, no. 9, pp. 1871–1874, 2008.
- [38] J. M. Bioucas-Dias, A. Plaza, N. Dobigeon, M. Parente, Q. Du, P. Gader, and J. Chanussot, "Hyperspectral unmixing overview: Geometrical, statistical, and sparse regression-based approaches," *Selected Topics in Applied Earth Observations and Remote Sensing IEEE Journal of*, vol. 5, no. 2, pp. 354–379, 2012.
- [39] X. Kang, S. Li, and J. A. Benediktsson, "Feature extraction of hyperspectral images with image fusion and recursive filtering," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 52, no. 6, pp. 3742–3752, 2014.
- [40] F. Li, L. Xu, P. Siva, A. Wong, and D. A. Clausi, "Hyperspectral image classification with limited labeled training samples using enhanced ensemble learning and conditional random fields," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 8, no. 6, pp. 1–12, 2015.
- [41] 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 and Remote Sensing*, vol. 8, no. 6, pp. 2381–2392, 2015.