A CNN WITH MULTISCALE CONVOLUTION FOR HYPERSPECTRAL IMAGE CLASSIFICATION USING TARGET-PIXEL-ORIENTATION SCHEME

Jayasree Saha, Yuvraj Khanna, and Jayanta Mukhopadhyay

Indian Institute of Technology, Kharagpur, West Bengal, India

ABSTRACT

Recently, convolution neural network (CNN)-based hyperspectral image (HSI) classification has gained attention due to its remarkable performance when the number of training samples is sufficiently large. Existing approaches involve 1-D, 2-D or 3D CNN based architecture. 3D CNN involves heavy computation and others are inefficient in using spatial and spectral information jointly. In this work, we propose to use pointwise 3-D convolution to extract spectral feature, followed by 2-D convolution to extract spectral-spatial features jointly in an end to end manner. We have shown a variation in inception-like high level architecture for feature extraction. This is guided by the fact that averaging a large number of feature may lose some unique information. On the other hand, existence of spatial variability within a class and similarity among different classes incurs degradation in HSI classification. To combat with that, we propose a novel target-patch-orientation (TPO) scheme to form a spatial-spectral neighborhood of a pixel. The Experimental results reveal that TPO scheme has positive impact on the proposed model.

Index Terms— Target-Pixel-Orientation, multi-scale convolution, deep learning, hyperspectral classification.

1. INTRODUCTION

Hyperspectral image (HSI) classification has received considerable attention in recent years for a variety of application using neural network-based techniques. Processing a large number of spectral dimension in HSI suffers from the curse of dimensionality. Also, a few properties of the dataset bring challenges in the classification of HSIs, such as i) limited training examples, and ii) large spatial variability of spectral signature. In general, contiguous spectral bands may contain some redundant information which leads to the Hughes phenomenon [5]. It causes accuracy drop in classification when there is an imbalance between the high number of spectral channels and scarce training examples. A few attempts have been made to handle large spectral information in deep learning community [3]. On the other hand, HSI classification task gets more complicated with the following facts: i) spectral signature of objects belonging to the same class may be different, and ii) the spectral signature of objects belonging to different classes may be the same. Therefore, spectral profile of a target pixel may not be sufficient for classification. In this paper, we use neighboring spectral information to generate spectral feature for the target pixel. Recent studies prove that incorporation of spatial context along with spectral components improves classification task considerably. Multi-scale convolution is widely popular in literature for exploring complex spatial context. In most cases, outputs of different kernels are either concatenated or summed [4] together and further processed for feature extraction and classification. However, there exists no study on whether concatenating a large number of filter banks of different scales result in improved classification. In the present context, we are more interested in scrutinizing various CNN architectures for the current problem. Our work is motivated by the following observations.

1. Substantial spectral information brings redundancy, and also taking all spectral bands together decreases the performance of classification. Hence, band reduction is required in hyperspectral image classification.

2. Due to complex reflectance property in HSI, pixels of the same class may have different spectral signature and pixels of different classes may have similar spectral signature. Spatial neighborhood plays a key role in improving classification accuracies in such a scenario.

3. A suitable network architecture is needed to incorporate spatial and spectral feature in order to achieve better classification accuracy for hyperspectral images.

Section 2 gives a detailed description of the proposed approach. Experimental results are discussed in Section 3. Section 4 concludes the paper.

2. PROPOSED APPROACH

The proposed HSI classification framework is shown in Fig 1. It mainly consists of three tasks: organizing a Target-Pixel-Orientation (TPO) scheme to generate a neighborhood of a target pixel, constructing a CNN-based architecture to extract spectral-spatial feature, and determining the label of testing samples.
2.1. Target-Pixel-Orientation

Consider a hyperspectral data set $H$ with $d$ spectral bands. We have $N$ labeled samples denoted as $P = \{p_i\}_{i=1}^N$ in an $R^{d \times 1}$ feature space and class labels $y_i \in \{1, 2, \cdots, C\}$ where $C$ is the number of class. Let $n_c$ be the number of available labeled samples in the $c$-th class and $\sum_{c=1}^C n_c = N$. We propose a Target-Pixel-Orientation (TPO) scheme. In this scheme, we consider a $k \times k$ window whose center pixel is the target pixel. We select eight 2-D neighbors of the target pixel by simply shifting the window toward eight different directions in a clockwise manner. Fig 2 shows one example of how we prepare eight neighbors of a target pixel of size $3 \times 3$. We mark the target pixel in blue color which is surrounded by a $3 \times 3$ neighboring window shown in a red border. First sub-image in Fig 2 depicts the $3 \times 3$ window when the target pixel occupies the center position of that window. Other eight sub-images are the neighbors of the first sub-image which are numbered by 1 to 8. We consider each of nine windows as a single integrated view for the target pixel. However, we have shown the TPO view with one spectral channel to make the illustration simple. In our proposed system, we consider $d$ spectral channels. Therefore, input to the CNN model is a 4-dimensional tensor. We perform the following operation to form the input for our models.

$$S\{S(j)i \in d\} \forall j \in V$$  \hspace{1cm} (1)

Where $S$ is a function which is responsible for stacking of inputs and $s_{ji}$ represents $k \times k$ patch of $j$-th spectral channel in $i$-th view. $V$ represents nine views in the TPO scheme. We have converted labeled samples $P$ to $I = \{i_1, i_2, \cdots, i_N\}$ such that each $i_c$ has $9 \times d \times k \times k$ dimension.

2.2. Network Architecture

It consists of mainly three blocks, namely, spectral-feature extraction, spatial-spectral feature extraction, and classification. The TPO scheme extracts samples from the given dataset as described in Section 2.1. The label of each sample is that of the pixel located in the center of the first view among the nine views (discussed in Sec 2.1).

2.2.1. Spectral Feature Extraction

Aggregation of neighboring pixels may give more generalized feature. It may reduce the ambiguity lies in spectral signature as discussed in Section 1. In this paper, we consider a 3-D neighborhood. To collect only the spectral feature, point-wise 3-D convolution is the obvious choice. A point-wise 3-D convolution transforms an input $X \in R^{V \times d \times k \times k}$ to a feature map $U \in R^{V' \times d' \times k \times k}$. In the notation, $f = [f_1, f_2, \cdots, f_{V'}]$ denotes the learned set of filter kernels, where $f_i$ refers to the parameters of the $i$-th filter. The outputs can be written as $U = [u_1, u_2, \cdots, u_{V'}]$, where $u_c = f_c \ast X = \sum_{i=1}^{V} f_i c \ast X^i$ Here, $\ast$ denotes convolution, $f_c = [f_{c1}^i, f_{c2}^i, \cdots, f_{cV'}^i]$, and $X = [X^1, X^2, \cdots, X^{V'}]$, and $u_c \in R^{V' \times d' \times k \times k}$. $f_i \in R^{T \times 1 \times 1}$ is the point-wise kernel corresponds to the $i$-th view of $X$. To simplify the notation, bias terms are omitted. Block-1 in Figure 1 represents the network architecture adopted for spectral feature extraction. This block contains three consecutive “BasicConv3d” layers. The designed “BasicConv3d” layer contains 3-D batch normalization layer and rectified linear unit (ReLU) layer sequentially after 3-D point-wise convolution layer. Parameters of 3-D convolution layer are the input channel, output channel, and the kernel size.

2.2.2. spatial-spectral feature extraction

Our assumption is that a shallow but wider network, i.e., “multi-scale filter bank” extracts more appropriate features from small patches. Hence, we use an inception-module for feature extraction. We use two variations in inception architecture (leading to two models MS-CNN1 and MS-CNN2) to experiment the quality of aggregation of large features. Inception layer of MS-CNN1 is shown in Figure 3. The inception layer in MS-CNN2 was introduced in our earlier work [6]. MS-CNN1 uses a multi-scale filter bank that locally convolves the input sample with four parallel blocks with different filter sizes in convolution layer. Each parallel...
block consists of either one or many “BasicConv2D” layer and pooling layer. The $3 \times 3$ and $5 \times 5$ filters in inception layer are used to exploit local spatial correlations of the input sample while the $1 \times 1$ filters are used to address correlations among nine views and their respective spectral information. The outputs of the $MS-CNN1$ feature extraction layer are combined at concatenation layer to form a joint view-spatio-spectral feature map used as input to the subsequent layers. In $MS-CNN1$, we have used an adaptive average pooling layer sequentially after the concatenation layer. However, the inception architecture of $MS-CNN1$ is splitted into three small inception layers in $MS-CNN2$.

2.2.3. Classification

Outputs of feature extraction block are flattened and fed to the fully connected layers whose output channel is the number of classes. The fully connected layers is followed by 1-D Batch-Normalization layer and a softmax activation function. In general, the classification layer can be defined as $p = \text{softmax}(BN(Wa + b))$ where $a$ is the input of the fully connected layer, and $W$ and $b$ are the weights and bias of the fully connected layer, respectively. $BN(\cdot)$ is the 1-D Batch-Normalization layer. $p$ is the $C$-dimensional vector which represents the probability that a sample belongs to the $c^{th}$ class.

We have trained the proposed networks by minimizing cross-entropy loss function. Adam optimizer with a batch size of 512 samples is used with a weight decay of 0.0001. We initially set a base learning rate as 0.0001. All the layers are initialized from a uniform distribution.

3. EXPERIMENTS

The performance of HSI classification is observed by experimenting with the University of Pavia scene and the Salinas dataset. We have selected 200 labeled pixels from each class to prepare a training set for each of the datasets. We normalize them using $x - \mu$, where $x$ denotes the pixel value of a given spectral channel and $\mu$ and $\sigma$ provide mean and standard deviation of the complete dataset. We have chosen four state of the art methods namely: 1) CNN-PPF [2], 2) DR-CNN [4], 3) BASS [3], and 4) DPP-ML [1] for comparison. CNN-PPF uses a pixel pair strategy to increase the number of training samples and feeds them into a deep network having 1-D convolutional layers. DR-CNN exploits diverse region-based 2-D patches from the image to produce more discriminative features. However, BASSNET extracts band specific spatial-spectral features. In DPP-ML, convolutional neural networks with multiscale convolution are used to extract deep multiscale features from the HSI. We have presented best reported results of comparative method from the respective paper in Tables 1 and 2.

3.1. Results and Discussion

The performance of the $MS-CNN1$ and $MS-CNN2$ on test-samples are compared with the aforementioned deep learning-based classifiers in Tables 1 and 2. $MS-CNN2$ produces better results compared to $MS-CNN1$ in all the datasets as shown in Table 1 to Table 2 respectively. The results signify that the arrangements of multi-scale convolutions in $MS-CNN2$ is able to extract more useful features for the classification compared to $MS-CNN1$. In order to judge how the TPO strategy, as described in Section 2.1 affects the performance of the classifier, we compare the classification results using a single view where the position of the target pixel is the center of the given window. To measure the performance, Overall Accuracy (OA) is considered. It is the ratio of the total number of correctly classified samples to the total number of samples.
all classes. We observe that OA increases by 6.09%, 1.37% in MS-CNN1 model and 9.99%, 2.08% in MS-CNN2 model for classification of U. Pavia, Salinas, respectively. In brief, the TPO-scheme improves results in every dataset for both the models. Again, we observe MS-CNN2 provides better results compared to MS-CNN1. We have highlighted misclassified pixels of U. Pavia with white color code in Figure 4.

**Table 1**: Class-specific Accuracy(%) and Overall Accuracy (OA) for the University of Pavia dataset. Results over 10 independent experiments are reported for our proposed network.

| #Class | CNN-PFF | BASS | DR-CNN | DPP-M | MS-CNN1 (µ±σ) | MS-CNN2 (µ±σ) |
|--------|---------|------|--------|------|----------------|----------------|
| 1      | 97.42   | 97.71| 98.43  | 99.38| 99.38 ± 1.71  | 99.97 ± 0.03  |
| 2      | 95.76   | 97.93| 99.45  | 99.59| 99.99 ± 6.10  | 99.97 ± 0.02  |
| 3      | 94.05   | 94.95| 97.33  | 99.63| 99.63 ± 0.91  | 99.93 ± 0.13  |
| 4      | 97.52   | 97.80| 99.50  | 99.31| 96.88 ± 8.31  | 99.75 ± 0.19  |
| 5      | 100     | 100  | 100    | 100  | 100 ± 0.00    | 100 ± 0.00    |
| 6      | 99.13   | 96.60| 100    | 99.99| 99.94 ± 0.09  | 99.99 ± 0.00  |
| 7      | 96.19   | 98.14| 99.70  | 99.85| 97.80 ± 6.91  | 100 ± 0.00    |
| 8      | 93.62   | 95.46| 99.55  | 99.02| 98.94 ± 3.14  | 99.96 ± 0.04  |
| 9      | 99.60   | 100  | 100    | 100  | 100 ± 0.00    | 100 ± 0.00    |
| OA     | 96.48   | 99.68| 99.46  | 99.72| 98.62 ± 4.05  | 99.96 ± 0.01  |

**Table 2**: Class-specific Accuracy(%) and and Overall Accuracy (OA) for the Salinas dataset. Results over 10 independent experiments are reported for our proposed network.

| #Class | CNN-PFF | BASS | DR-CNN | DPP-M | MS-CNN1 (µ±σ) | MS-CNN2 (µ±σ) |
|--------|---------|------|--------|------|----------------|----------------|
| 1      | 100     | 100  | 100    | 100  | 99.89 ± 0.28  | 100 ± 0.00    |
| 2      | 99.88   | 99.97| 100    | 100  | 99.96 ± 0.08  | 99.98 ± 0.03  |
| 3      | 99.60   | 99.98| 100    | 100  | 99.90 ± 0.28  | 99.99 ± 0.04  |
| 4      | 99.49   | 99.89| 99.25  | 99.99| 99.99 ± 0.03  | 100 ± 0.00    |
| 5      | 98.34   | 99.59| 99.83  | 99.44| 99.83 ± 0.16  | 99.46 ± 1.16  |
| 6      | 99.97   | 100  | 100    | 100  | 100 ± 0.00    | 100 ± 0.00    |
| 7      | 100     | 99.91| 99.87  | 99.87| 99.99 ± 0.03  | 99.97 ± 0.06  |
| 8      | 88.68   | 90.11| 94.14  | 95.36| 90.81 ± 1.44  | 95.28 ± 0.95  |
| 9      | 98.33   | 99.73| 99.99  | 100  | 99.98 ± 0.02  | 99.98 ± 0.05  |
| 10     | 98.60   | 97.46| 99.20  | 98.85| 99.01 ± 0.73  | 99.69 ± 0.58  |
| 11     | 99.54   | 99.08| 99.77  | 99.77| 99.95 ± 0.10  | 99.98 ± 0.04  |
| 12     | 100     | 100  | 100    | 100  | 100 ± 0.00    | 100 ± 0.00    |
| 13     | 100     | 99.44| 99.44  | 99.86| 99.02 ± 0.00  | 100 ± 0.00    |
| 14     | 98.96   | 100  | 100    | 99.77| 99.96 ± 0.08  | 99.99 ± 0.04  |
| 15     | 83.53   | 83.94| 95.52  | 90.50| 92.11 ± 3.37  | 95.45 ± 1.97  |
| 16     | 99.31   | 99.38| 99.72  | 98.94| 100 ± 0.00    | 99.99 ± 0.02  |
| OA     | 94.80   | 95.36| 98.33  | 97.51| 96.83±0.41    | 98.30 ± 0.27  |

**4. CONCLUSION**

In this paper, we propose a strategy (namely, target-pixel orientation (TPO)) to incorporate spatial and spectral information of HSI. In general, classification accuracy degrades due to a class having different spectral signatures in HSI. With incorporation of spectral and spatial information together, classification accuracy can be improved. Our approach attempts to exploit these two factors in a combination by using the orientation of the Target-pixel-view. Our architectural design of neural network exploits point-wise 3-D convolutions for spectral feature extraction whereas we adopt multi-scale 2-D inception like architecture for spatial-spectral feature extraction. Both the models perform competitively with the existing approaches, but MS-CNN2 has shown better performance compared to MS-CNN1.

**5. REFERENCES**

[1] Gong et al., “A CNN With multiscale convolution and diversified metric for hyperspectral image classification,” in IEEE Transactions on Geoscience and Remote Sensing, vol. 57, no. 6, pp. 3599-3618, 2019.

[2] Li et al., “Hyperspectral image classification using deep pixel-pair features,” in IEEE Transactions on Geoscience and Remote Sensing, vol. 55, no. 2, pp. 844-853, Feb. 2017.

[3] Santara et al., “BASS Net: Band-Adaptive Spectral-Spatial feature learning neural Network for hyperspectral image classification,” in IEEE Transactions on Geoscience and Remote Sensing, vol. 55, no. 9, pp. 5293-5301, 2017.

[4] Zhang et al., “Diverse region-based CNN for hyperspectral image classification,” in IEEE Transactions on Image Processing, vol. 27, no. 6, pp. 2623-2634, June 2018.

[5] Hughes, “On the mean accuracy of statistical pattern recognizers”, in IEEE Transactions on Information Theory, vol. 14, no. 1, pp. 55-63, 1968.

[6] Saha et al., “From supervised to unsupervised learning for land cover analysis of Sentinel-2 multispectral images”, in IEEE International Geoscience and Remote Sensing Symposium (IGARSS), 2020.