

FULLY CONV-DECONV NETWORK FOR UNSUPERVISED SPECTRAL-SPATIAL FEATURE EXTRACTION OF HYPERSPECTRAL IMAGERY VIA RESIDUAL LEARNING

Lichao Mou, Pedram Ghamisi, Xiao Xiang Zhu

(1) Remote Sensing Technology Institute (IMF), German Aerospace Center (DLR), Germany
(2) Signal Processing in Earth Observation (SiPEO), Technical University of Munich (TUM), Germany

ABSTRACT

Supervised approaches classify input data using a set of representative samples for each class, known as *training samples*. The collection of such samples are expensive and time-demanding. Hence, unsupervised feature learning, which has a quick access to arbitrary amount of unlabeled data, is conceptually of high interest. In this paper, we propose a novel network architecture, fully Conv-Deconv network with residual learning, for unsupervised spectral-spatial feature learning of hyperspectral images, which is able to be trained in an end-to-end manner. Specifically, our network is based on the so-called encoder-decoder paradigm, i.e., the input 3D hyperspectral patch is first transformed into a typically lower-dimensional space via a convolutional sub-network (encoder), and then expanded to reproduce the initial data by a deconvolutional sub-network (decoder). Experimental results on the Pavia University hyperspectral data set demonstrate competitive performance obtained by the proposed methodology compared to other studied approaches.

Index Terms— Convolutional network, deconvolutional network, hyperspectral image classification, residual learning, unsupervised spectral-spatial feature learning.

1. INTRODUCTION

Along with the development of different Earth observation missions, hyperspectral imagery has been accessible at a reasonable cost over the last decade. Since hyperspectral images are characterized in hundreds of continuous observation bands, throughout the electromagnetic spectrum with high spectral resolution, such data have attracted considerable attention in the remote sensing community [1]. To benefit from this type of data, supervised hyperspectral image classification is among the most active research areas in the field of hyperspectral analysis.

There is an intensive literature on supervised classification models such as decision trees, random forests [2], and support vector machines (SVMs) [3]. However, these approaches are attributed as “shallow” models, which means that their ability to deal with nonlinear data, e.g., hyperspectral data demonstrate dense nonlinearity, is limited compared to the “deep”

ones. It is believed that, compared to the “shallow” models, deep learning architectures [4–6] are able to extract high-level, hierarchical, and abstract features, which are generally more robust to the nonlinear input data.

Convolutional neural network (CNN), which is regarded as an important branch of the deep learning family, has been attracting attention since they are capable of automatically discovering relevant contextual 2D spatial features in image categorization tasks. Very recently, a few supervised CNN-based models have been proposed for spectral-spatial classification of hyperspectral remote sensing images. Chen *et al.* [7] introduced a supervised, ℓ_2 regularized 3D CNN-based feature extraction model to extract efficient spectral-spatial features for the purpose of classification. Ghamisi *et al.* [8] proposed a self-improving CNN model, which combined a CNN with a fractional order Darwinian particle swarm optimization algorithm to iteratively select the most informative bands suitable for training the designed CNN.

Those CNNs, however, have been trained in a supervised manner via back-propagation which improved the state of the art performance on the hyperspectral image classification task. Despite the big success of the supervised CNNs, they have at least one potential drawback: there is an urgent need for an adequate amount of labeled training samples to be used for supervised training. However, these samples are difficult to be collected. Hence, unsupervised spectral-spatial feature learning, which has a quick access to arbitrary amount of unlabeled data, is potentially of high interest. In this paper, we aim to propose an end-to-end network, fully Conv-Deconv residual network, for unsupervised spectral-spatial feature learning of hyperspectral imagery. Basically, our network architecture is based on the so-called encoder-decoder paradigm. Specifically, the input is first transformed into a typically lower-dimensional space via a convolutional sub-network (encoder), and then expanded to reproduce the initial data by a deconvolutional sub-network (decoder). Moreover, the trained unsupervised Conv-Deconv network can be adapted for the classification of hyperspectral data by cutting off the deconvolutional sub-network, replacing the loss function, and fine-tuning it with respect to the new task, i.e., adjusting the weights using back-propagation. With this approach, typically, much smaller training sets are sufficient.

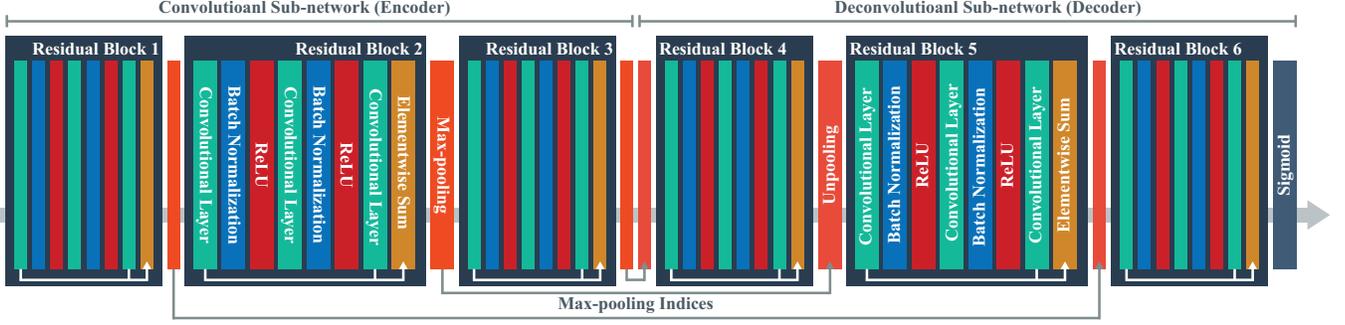


Fig. 1. We propose a network architecture which learns to extract spectral-spatial features by reconstructing the initial input 3D hyperspectral patches, being trained end-to-end. There are no fully connected layers and hence it is a fully Conv-Deconv network. The proposed network architecture is composed of two parts, i.e., convolutional sub-network and deconvolutional sub-network. The former corresponds to an encoder that transforms the input 3D hyperspectral patches to abstract feature representations, whereas the latter plays the role of decoder that reproduces the initial input data from the encoded features. Each layer in the convolutional sub-network has a corresponding decoder layer in the deconvolutional sub-network.

2. METHODOLOGY

2.1. Analysis and Modeling

Denote by $(\mathbf{x}, \mathbf{h}, \mathbf{y})$ random variables represent a 3D hyperspectral patch, its encoded feature representation, and the reconstructed output. The conditional probability distribution $p(\mathbf{y}|\mathbf{x})$ can be written as

$$p(\mathbf{y}|\mathbf{x}) = p(\mathbf{y}, \mathbf{h}|\mathbf{x}) = p(\mathbf{y}|\mathbf{h})p(\mathbf{h}|\mathbf{x}), \quad (1)$$

where $p(\mathbf{h}|\mathbf{x})$ indicates the distribution of the encoded feature representations given the input hyperspectral patches. As a special case, \mathbf{y} may be a deterministic function of \mathbf{x} . Ideally we would like to find $p(\mathbf{h}|\mathbf{x})$ and $p(\mathbf{y}|\mathbf{h})$, but direct application of Bayesian theory is not feasible. We, therefore, in this work resort to an estimate function $f(\mathbf{x})$ which minimizes the following mean squared error objective:

$$\mathbb{E}_{\mathbf{x}} \|\mathbf{x} - f(\mathbf{x})\|_2^2. \quad (2)$$

The minimizer of this loss is the conditional expectation:

$$\hat{f}(\mathbf{x}_0) = \mathbb{E}_{\mathbf{y}}[\mathbf{y}|\mathbf{h}] + \mathbb{E}_{\mathbf{h}}[\mathbf{h}|\mathbf{x} = \mathbf{x}_0], \quad (3)$$

that is the expected reconstructed output given a hyperspectral patch.

Given a set of unlabeled 3D hyperspectral patches $\{\mathbf{x}_i\}$, we learn the weights Θ of a network $f(\mathbf{x}; \Theta)$ to minimize a Monte-Carlo estimate of the loss (2):

$$\hat{\Theta} = \arg \min_{\Theta} \sum_i \|\mathbf{x}_i - f(\mathbf{x}_i; \Theta)\|_2^2. \quad (4)$$

This means that we train the network to reproduce the input results in learning high-level abstract features in an unsupervised manner.

In this paper, we propose a fully Conv-Deconv network (cf. Fig. 1) in which the desired output is the input data itself. The proposed network architecture is composed of two parts, i.e., the convolutional sub-network and deconvolutional sub-network. The former corresponds to an encoder that transforms the input 3D hyperspectral patch \mathbf{x}_i to abstract feature representation \mathbf{h}_i , whereas the latter plays the role of a decoder that reproduces the initial input data from the encoded feature. Each layer in the convolutional sub-network has a corresponding decoder layer in the deconvolutional sub-network.

2.2. Conv-Deconv Network with Residual Learning

The proposed Conv-Deconv network with residual learning is a modularized network architecture that stacks residual blocks. Similarly to the convolutional blocks, a residual block consists of several convolutional layers that are with the same feature map size and have the same number of filters. However, it performs the following calculation:

$$\varphi_l = g(\phi_l) + \mathcal{F}(\phi_l; \Theta_l), \quad (5)$$

$$\phi_{l+1} = f(\varphi_l). \quad (6)$$

Here, ϕ_l indicates the feature maps that are fed into the l -th residual block and satisfies $\phi_0 = \mathbf{x}$ where \mathbf{x} is the input 3D hyperspectral patch. $\Theta_l = \{\Theta_{l,k} | 1 \leq k \leq K\}$ represents a collection of weights associated with the l -th residual block, and K denotes that there are K convolutional layers in a residual block. Moreover, \mathcal{F} is the residual function, which is generally achieved by few stacked convolutional layers. The function f indicates the activation function such as a linear activation function or ReLU, and f works after element-wise addition. The function g is fixed to an identity mapping: $g(\phi_l) = \phi_l$.

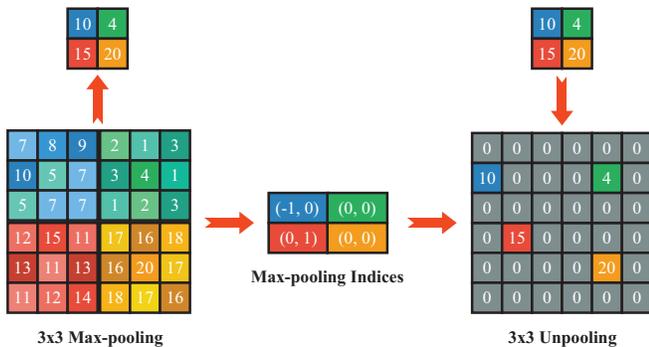


Fig. 2. An illustration of the unpooling operation in the Conv-Deconv residual network, using max-pooling indices which is capable of recording the location of the maximum value in each local pooling region during pooling in the convolutional sub-network.

If f adopts a linear activation function and also acts as an identity mapping, i.e., $\phi_{l+1} = \phi_l$, we can obtain the output of the l -th residual block by putting Eq. (5) into Eq. (6):

$$\phi_{l+1} = \phi_l + \mathcal{F}(\phi_l; \Theta_l). \quad (7)$$

Recursively like

$$\begin{aligned} \phi_{l+2} &= \phi_{l+1} + \mathcal{F}(\phi_{l+1}; \Theta_{l+1}) \\ &= \phi_l + \mathcal{F}(\phi_l; \Theta_l) + \mathcal{F}(\phi_{l+1}; \Theta_{l+1}), \end{aligned} \quad (8)$$

etc., we will get the following recurrence formula:

$$\phi_L = \phi_l + \sum_{i=l}^{L-1} \mathcal{F}(\phi_i; \Theta_i), \quad (9)$$

for any shallower block l and any deeper block L .

2.3. Unpooling

The convolutional sub-network is responsible for extracting high-level abstract spectral-spatial feature representation of the input 3D hyperspectral patch, by interleaving convolutional layers and max-pooling layers, i.e., spatially shrinking the feature maps layer by layer. Pooling is necessary to allow agglomerating information over large areas of feature maps, and more fundamentally, to make the network computationally feasible. However, pooling leads to reduced resolution of the feature maps, hence in order to reconstruct the initial input data we need unpooling to unpool the feature maps, i.e., to increase their spatial span, as opposed to the pooling implemented by the convolutional sub-network. Fig. 2 illustrates the details of the unpooling operation.

3. EXPERIMENTS

3.1. Data Description

We used the bench-mark Pavia University data set, which was captured by reflective optics system imaging spectrometer (ROSIS) covering the Engineering School at the University of Pavia. The available training samples present nine classes, mostly related to land-covers. The image is of 610×340 pixels with a spatial resolution of 1.3 m per pixel and was collected under the HySens project managed by the German Aerospace Agency (DLR). The hyperspectral imagery consists of 115 spectral channels ranging from 430 to 860 nm. In this paper, we made use of 103 spectral channels, after removing 12 noisy bands.

3.2. General Information

To validate the effectiveness of the proposed network architecture for the purpose of hyperspectral image classification, the novel classification method is compared with the most widely used supervised models, random forest and support vector machines (SVM). In addition, in this paper, the experiments making use of other supervised deep learning methods such as 1D CNN, 2D CNN [7], and SICNN [8] are also carried out to verify the validity of the results obtained by the proposed network.

For the network configuration, we leverage convolutional filters with a very small receptive field of 3×3 . In addition, the convolutional stride is fixed to 1 pixel; the spatial padding is also 1 pixel. Max-pooling is performed over 3×3 pixel windows with stride 3. All the convolutional layers are with ReLU as nonlinear activation function except the last layer that uses sigmoid activation. The fully Conv-Deconv residual network was trained using the Adam algorithm [9], and all the suggested default parameters were used for all the following experiments. Once the training of the Conv-Deconv residual network is complete, we can start to fine-tune the network for hyperspectral data classification. We made use of SGD with a fairly low learning rate of 0.0001, to fine-tune the network.

3.3. Fine-tuned Network for Hyperspectral Image Classification

The classification maps of the Pavia University data set obtained by the widely used classifiers (e.g., random forest and SVM), supervised CNNs, and our method are shown in Fig. 3, and the corresponding accuracy indexes are presented in Table 1. It can be seen that the proposed fine-tuned Conv-Deconv residual net achieves better scores for OA and Kappa coefficient compared to all other methods. It is worth noting that our method for feature learning is unsupervised, while 1D CNN, 2D CNN, and SICNN are supervised networks. Taking this into account, the performance of our approach is competitive and satisfactory.

Table 1. Classification accuracy Comparison for the Pavia University Data Set. The Best Accuracy in Each Row is Shown in Bold.

Class No.	Class Name	RF-200	SVM-RBF	1D CNN	2D CNN	SICNN	Conv-Deconv Net
1	Asphalt	80.85	80.80	83.73	70.64	84.21	82.81
2	Meadows	55.29	66.78	65.70	93.38	91.10	97.11
3	Gravel	52.93	73.18	67.03	62.60	64.36	60.31
4	Trees	98.79	95.17	94.03	94.22	95.53	95.59
5	Metal Sheets	99.26	99.55	99.41	100	97.70	97.55
6	Bare Soil	78.76	92.90	96.30	49.00	56.53	59.38
7	Bitumen	84.36	90.08	93.83	70.08	77.29	78.42
8	Bricks	91.58	91.20	93.56	94.19	95.57	96.50
9	Shadows	98.20	93.77	99.79	93.66	96.20	92.29
OA	-	71.37	78.82	80.51	82.75	85.25	87.82
AA	-	82.23	87.05	88.15	80.86	84.28	84.44
Kappa	-	0.6484	0.7358	0.7423	0.7697	0.8041	0.8363

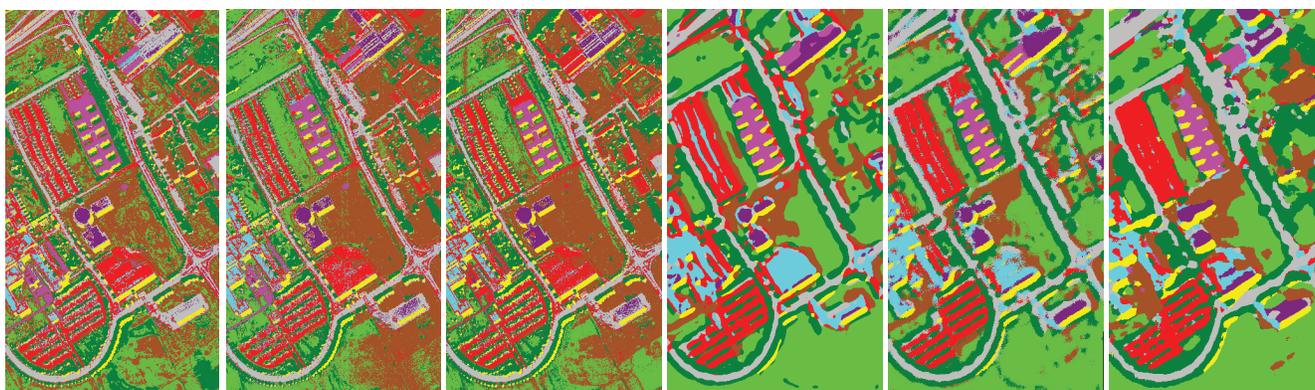


Fig. 3. Classification results obtained by different methods for the Pavia University scene: (a) RF (with 200 trees); (b) SVM-RBF (hyperplane parameters are estimated using five-fold cross-validation); (c) 1D CNN; (d) 2D CNN; (e) SICNN; and (f) Fine-tuned Conv-Deconv residual net. Note that we used the standard sets of training and test samples for the data sets.

4. CONCLUSION

In this paper, we proposed a novel end-to-end fully Conv-Deconv residual network architecture for unsupervised spectral-spatial feature extraction of hyperspectral images. In the future, further experiments and studies will be conducted to fully understand the “block box” of the proposed fully Conv-Deconv network with residual learning.

References

- [1] J. A. Benediktsson and P. Ghamisi, *Spectral-spatial classification of hyperspectral remote sensing images*, Artech House, INC, Boston, USA, 2015.
- [2] J. Ham, Y. Chen, M. M. Crawford, and J. Ghosh, “Investigation of the random forest framework for classification of hyperspectral data,” *IEEE Trans. Geos. Remote Sens.*, vol. 43, no. 3, pp. 492–501, 2005.
- [3] F. Melgani and L. Bruzzone, “Classification of hyperspectral remote sensing images with support vector machines,” *IEEE Trans. Geos. Remote Sens.*, vol. 42, no. 8, pp. 1778–1790, 2004.
- [4] H. Lyu, H. Lu, and L. Mou, “Learning a transferable change rule from a recurrent neural network for land cover change detection,” *Remote Sens.*, vol. 8, no. 6, pp. 506, 2016.
- [5] L. Mou, P. Ghamisi, and X. Zhu, “Deep recurrent neural networks for hyperspectral image classification,” *IEEE Trans. Geos. Remote Sens.*, in press.
- [6] L. Mou and X. Zhu, “Spatiotemporal scene interpretation of space videos via deep neural network and tracklet analysis,” in *IEEE IGARSS*, 2016.
- [7] Y. Chen, H. Jiang, C. Li, X. Jia, and P. Ghamisi, “Deep feature extraction and classification of hyperspectral images based on convolutional neural networks,” *IEEE Trans. Geos. Remote Sens.*, vol. 54, no. 10, pp. 6232–6251, 2016.
- [8] P. Ghamisi, Y. Chen, and X. Zhu, “A self-improving convolution neural network for the classification of hyperspectral data,” *IEEE Geos. Remote Sens. Lett.*, vol. 13, no. 10, pp. 1537–1541, 2016.
- [9] D. Kingma and J. Ba, “Adam: A method for stochastic optimization,” *arXiv:1412.6980v8*, 2015.