TRANSFERRED TENSOR DECOMPOSITION-BASED DEEP LEARNING FOR HYPERSPECTRAL ANOMALY DETECTION

Yulei Wang[1,2], Fengchao Wang[1], Qingyu Zhu[1], Meiping Song[1], and Chunyan Yu[1]

[1] Center for Hyperspectral Imaging in Remote Sensing (CHIRS), Information Science and Technology College, Dalian Maritime University, Dalian 116026, China;
[2] State Key Laboratory of Integrated Services Networks, School of Telecommunications Engineering, Xidian University, Xian 710071, China

ABSTRACT

This paper proposes a new hyperspectral anomaly detection method based on transferred deep learning and tensor decomposition. Firstly, since there is no labeled input data for training in anomaly detection, the detection model is obtained by training convolutional neural network with transferred learning. Then the model is decomposed to increase the number of convolution layers, that is, the depth of the network, so as to give more accurate results without over fitting. At the same time, the spatial information of the input data is extracted in order to make full use of the existing data for detection. Finally, combining the spectral and spatial information of the current pixel, the detection result is given. Experiments on two hyperspectral datasets show that the proposed algorithm has excellent performance.

Index Terms—Anomaly detection (AD), hyperspectral image (HSI), tensor representation, Tucker decomposition.

1.INTRODUCTION

Anomaly detection (AD) has received considerable interest in recent years, and the main idea of AD is to detect pixels whose spectral information is significantly different from the surrounding background, so the distribution of the outliers is usually sparse. One of the characteristics of AD is that the spectral information of anomaly target is usually unknown, which is a great challenge for AD.

The most widely used AD method is the Reed-Xiaoli (RX) algorithm, which calculates the Mahalanobis Distance based on the assumption of background distribution [1]. An algorithm based on background dictionary and joint sparse representation was proposed in [2]. Recently, methods based on deep learning and tensor are applied in hyperspectral processing. For example, a convolutional neural network (CNN) based on tensor anomaly detection algorithm was proposed in [3]. Li et al proposed the transferred learning algorithm [4], but the network only uses the spectral information for training and detection, ignoring the spatial information. Besides, the designed CNN is a little shallow, and the features learned are insufficient. Generally speaking, shallow neural network may not be able to extract the features of training data very well, the deeper the layers of CNN, the better the effect of the model. However, the lack of training data also leads to great resistance in training deep neural network because of over fitting.

In this paper, a transferred deep learning hyperspectral anomaly detection algorithm based on tensor decomposition is proposed, the original one convolution layer is decomposed into three convolution layers, which increases the number of convolution layers and network depth, and increases the number of parameters greatly, which is conducive to the network to better show the learned characteristics. Due to the lack of training hyperspectral data, it is hard to design and train the deeper network, so, to a certain extent, this method could address this problem effectively. Originally, only spectral information of input data was used for detection, ignoring the spatial information, which may lead to poor results. Tucker decomposition is then applied in this paper, in the core convolution layer obtained by decomposition, conv2d function was used for two-dimensional convolution, which can combine spatial and spectral information.

2. PROPOSED METHOD

In this section, the flow and principle of the proposed algorithm are introduced in three parts: firstly, train a CNN model on a labeled dataset, and then the convolution layer in the model is decomposed, changing the structure of convolution layer and increasing the depth of the network. Then the detection results are obtained by combining spectral and spatial information. Finally, the number of network parameters before and after decomposition is briefly analyzed.

2.1 Transferred CNN model

Consecutive convolution layers are used to extract hyperspectral data features. After that, the spectral information of the current pixel is mapped to a specific value by 1×1 convolution and full connection layer. The convolutional neural network is trained to learn the spectral features, the specific steps are as follows: Firstly, a known hyperspectral image data set is used to select pixels of the same category and different categories, and then they are paired and sent to network for learning, so that the network can judge whether the input pixel pairs are similar. The specific process of network learning is realized by designing loss function. When the same type of pixel pairs is inputted, the loss function will make the network continuously attenuate towards the direction of output 0; when different types of pixel pairs are inputted, the loss function will make the network continuously attenuate towards the direction of output 1. After certain iterations, the convolutional neural network has the ability to distinguish the similarity of pixel pairs. Finally, the test image is inputted for detection, and 16 pixels around the pixel to be detected are selected to form pairs of pixels. The network gives their similarity score, and the average value is taken as the output. The pixels whose score are greater than the threshold are judged as abnormal, while those pixels whose score are less than the threshold are considered as background. The well-trained model is saved for subsequent decomposition [4].

2.2 Tucker Decomposition

Tucker-2 decompositions are used, the original input 3-way tensor is decomposed into core tensor and factor matrices, every convolution layer is replaced by three new convolution layers, referred to as: first layer, core layer and last layer. The weight parameters of the convolution layer in the model are extracted as input, tensors obtained by decomposition will be assigned to three convolution layers as their weights, respectively.

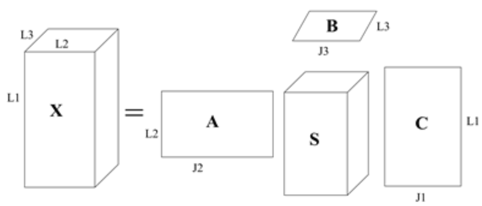


Fig.1. Structure of 3-way Tucker decomposition.

Convolution operations map input data X of size $H \times W \times S$ to output data Y of size $H' \times W' \times T$.

$$Y_{h',w',t} = \sum_{m=1}^{D}\sum_{n=1}^{D}\sum_{s=1}^{S} W_{m,n,s,t} X_{h_m,w_n,s} \tag{1}$$

The kernel tensor W is decomposed to:

$$W_{m,n,s,t} = \sum_{s=1}^{S}\sum_{t=1}^{T} P_{m,n,s,t} Q_s Q_t \tag{2}$$

Where P is a 4-way core tensor, $Q_s Q_t$ are factor matrices. Then three continuous convolution formulas can be obtained after decomposition [5], they form three new convolution layers.

$$M_{h,w,s} = \sum_{s=1}^{S} Q_s X_{h,w,s} \tag{3}$$

$$M'_{h',w',t} = \sum_{m=1}^{D}\sum_{n=1}^{D}\sum_{s=1}^{S} P_{m,n,s,t} M_{h,w,s} \tag{4}$$

$$Y_{h',w',t} = \sum_{t=1}^{T} Q_t M'_{h',w',t} \tag{5}$$

After tucker decomposition, the number of channels in convolution layers will change, the structure of every layer is designed as follows:

1) first layer:

One dimensional convolution is used to convolute the spectral vector on the first layer, since the input data has 189 bands, the size of 1 × 1 convolution kernel is 1×1×189. The first layer maps the input data (one-dimensional column vector) to an element. The number of output channels will be reduced, generally set to one third of the input channels.

2) core layer:

After convolution of the first layer, the spatial information of the data does not change and still corresponds to the original input, $D \times D$ convolution kernel is used to extract spatial information, the size of convolution kernel is $D \times D \times 189$. Here, D is equal to the size of convolution kernel used in convolution layer before decomposition.

3) last layer:

At last, the 1×1 convolution kernel is still used, but the output channel will be expanded to three times of the input to restore the size of the original convoluted data.

After the convolution layer is decomposed, the decomposed model is saved, each convolution layer is decomposed into three corresponding convolution layers, and the decomposition tensor is used as the parameter of the new convolution layer, which deepens the network depth and anomaly detection is carried out on the testing data by combining with spatial and spectral information.

2.3 Analysis of network parameters

A convolution layer of undecomposed neural network is taken as an example in Fig.2, the size of convolution kernel is 1×3, the number of input and output channels is both 20. Therefore, the number of convolution layer parameters is equal to 20×3×20=1200. In our network, the layer with the most parameters has 40 input and output channels, so it has 40×3×40=4800 parameters.

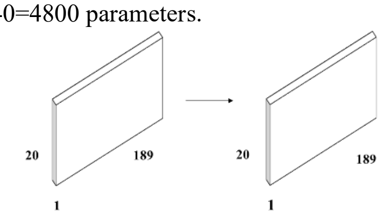


Fig.2. Convolution layer architecture

The three convolution layers obtained from the above convolution layer decomposition are shown by the arrows in Fig.3. As mentioned above, S' and T' are equal to one third of S and T, respectively. Three convolution kernels of different sizes are used to change the convolution mode. Total number of convolution layer parameters is equal to $1\times1\times S\times S/3+D\times D\times S/3\times T/3+1\times1\times T/3\times T$, where D is the convolution kernel size used in the convolution layer of the original model, D equals to 3, S and T are the number of input and output channels. For the current convolution layer, they are equal to 189, which is a significant difference from original. Due to the change of the structure of the convolution layer and the way of reading data, the dimension is no longer equal to 20. This makes total number of parameters expand many times, it equals to 59535.

This operation increases the amount of calculation greatly and leads to the extension of running time. On the other hand, the increase of parameters is helpful to improve the detection accuracy to a certain extent without over fitting.

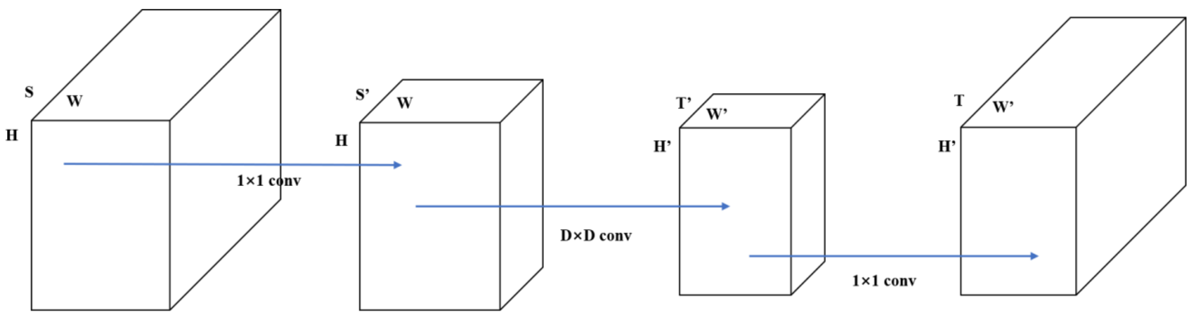


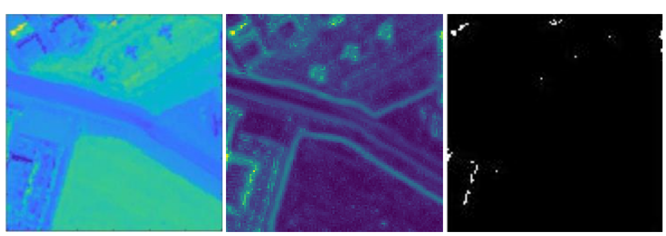
Fig.3. Flowchart of Tucker-2 decompositions

3.EXPERIMENTAL RESULT

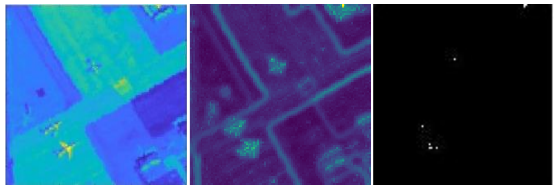
Since the experiment is based on transferred deep learning, we use training and test data from the same sensor. Training data comes from the Airborne Visible/Infrared Imaging Spectrometer (AVIRIS) sensor, it is an area in California over Salinas Valley. The image has 224 bands ranging from 400 to 1800 nm, and every band has 512×217 pixels. It contains a large number of labeled samples. It mainly contains 16 classes and hundreds of labeled samples per class. For example, it contains bare soils, vegetables, and vineyard fields [4].

The testing data was also collected by AVIRIS sensor. They are parts of hyperspectral datasets of SanDiego Naval Base. The original image has 224 bands, after removing water absorption and other bands, there are 189 bands left. Each band has 400×400 pixels originally, we use the three planes parts which has 120×120 and 100×100 pixels separately.

Fig.4 shows the image of 90th band HSI of the testing data, detection results as well as binary graphs. Receiver-operating-characteristic (ROC) curve is usually used as an indicator of quantitative analysis, the area under curve is called AUC. AUC value shows the effect of the algorithm. All the experimental results are obtained from an Intel Core i5-9500 central processing unit machine with 8 GB of random access memory. In our method, the AUC values of the two datasets are 0.9702 and 0.9730 and the original values (without using tucker decomposition) are only 0.9180 and 0.9086 in Fig.5. The performance has improved a lot obviously. The time consumption of our method is 1886.89s and 1291.83s. The running time is also acceptable.

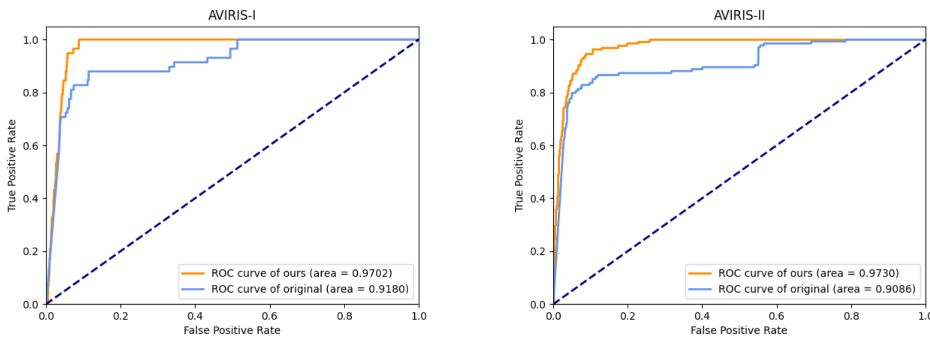


(a) 90th band of SanDiego1 HSI data, detection result and binary graph of detection result (120×120 pixels)



(b) 90th band of SanDiego2 HSI data, detection result and binary graph of detection result (100×100 pixels)

Fig.4. Two HSI data sets of SanDiego and detection results



(a) Detection performance of SanDiego1 data set

(b) Detection performance of SanDiego2 data set

Fig.5. Ours ROC curves and original ROC curves of the above two HSIs in Fig.4.

4. CONCLUSION

In this paper, a deep learning hyperspectral anomaly detection algorithm based on tensor decomposition is proposed. Firstly, convolution neural network is designed for transferred learning, and data sets with rich sample information are selected for training, each convolution layer in the model trained by transferred learning is decomposed into three new convolution layers, which increases the depth of the network. After decomposition, two-dimensional convolution is used to extract the spatial information of the current pixel. Finally, the anomaly target is detected by combining the spectral information and spatial information. Experimental results show that the proposed algorithm has advanced performance, the average increase was 5.8 percentage points. But the algorithm is time-consuming, how to maintain the accuracy and shorten the running time is the next problem to be solved.

5. ACKNOWLEDGEMENTS

This work is supported by the National Nature Science Foundation of China (61801075, 61971082), China Postdoctoral Science Foundation (2020M670723), Open Research Funds of State Key Laboratory of Integrated Services Networks, Xidian University (ISN20-15) and the Fundamental Research Funds for the Central Universities (3132019341).

REFERENCES

[1] S. Reed and X. Y u, “Adaptive multiple-band CFAR detection of an optical pattern with unknown spectral distribution,” IEEE Transactions on Acoustics, Speech, and Signal Processing, vol. 38, no. 10, pp. 1760–1770, October 1990.
[2] Huyan N, Zhang X, Zhou H, et al. Hyperspectral Anomaly Detection via Background and Potential Anomaly Dictionaries Construction[J]. IEEE Transactions on Geoscience and Remote Sensing, 2019, 57(4):2263-2276.
[3] L. Zhang and B. Cheng, "Transferred CNN Based on Tensor for Hyperspectral Anomaly Detection," in IEEE Geoscience and Remote Sensing Letters, vol. 17, no. 12, pp. 2115-2119, Dec. 2020.
[4] W. Li, G. Wu and Q. Du, "Transferred deep learning for hyperspectral target detection," 2017 IEEE International Geoscience and Remote Sensing Symposium (IGARSS), Fort Worth, TX, 2017, pp. 5177-5180.
[5] Kim Y D, Park E, Yoo S, et al. Compression of Deep Convolutional Neural Networks for Fast and Low Power Mobile Applications[J]. Computer Science, 2015, 71(2):576-584.
[6] Ma N, Peng Y, Wang S, et al. An Unsupervised Deep Hyperspectral Anomaly Detector[J]. Sensors, 2018, 18(3):693.
[7] Zhang G, Zhao S, Li W, et al. HTD-Net: A Deep Convolutional Neural Network for Target Detection in Hyperspectral Imagery[J]. Remote Sensing, 2020, 12(9):1489.
[8] Chang S, Du B, Zhang L. A Sparse Autoencoder Based Hyperspectral Anomaly Detection Algorithm Using Residual of Reconstruction Error[C]. IGARSS 2019 - 2019 IEEE International Geoscience and Remote Sensing Symposium. IEEE, 2019.
[9] Li Z, Zhang Y. Hyperspectral Anomaly Detection Based on Improved RX with CNN Framework[C]. IGARSS 2019 - 2019 IEEE International Geoscience and Remote Sensing Symposium. IEEE, 2019.
[10] Xiong J, Ling Q, Lin Z, et al. Kernel Sparse Representation for Anomaly Detection in Hyperspectral Imagery[C]. The 2nd International Conference. 2018.