Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 129 (2018) 219–223

Procedia Computer Science

www.elsevier.com/locate/procedia

2017 International Conference on Identification, Information and Knowledge in the Internet of Things

Guided filter based Deep Recurrent Neural Networks for Hyperspectral Image Classification

Yanhui Guo[a,b], Siming Han[a,*],Han Cao[a], Yu Zhang[a,*], Qian Wang[a]

[a] *Shaanxi Normal University, Xi'an Shaanxi710119, China*
[b] *Shandong Women's University, Ji'nan Shandong, 250300, China*

Abstract

Hyperspectral image(HSI) classification has been a hot topic in the remote sensing community. A large number of methods have been proposed for HSI classification. However, most of them are based on the extraction of spectral feature, which leads to information loss. Moreover, they rarely consider the correlation among the spectrums. In this paper, we see spectral information as a sequential data which is relevant with each other. We introduce long short-term memory model, which is a typical recurrent neural network (RNN), to deal with HSI classification. In order to solve the problem of difficult to reach the steady state of the model, we proposed a novel guided filter based RNN model. Also, we proposed a method for modeling hyperspectral sequential data, which is very useful for future research work. The experimental results show that our proposed method can improve the classification performance as compared to other methods in two popular hyperspectral datasets.

Selection and peer-review under responsibility of the scientific committee of the 2017 International Conference on Identification, Information and Knowledge in the Internet of Things (IIKI2017).

*Keywords:*Recurrent Neural Network, long short-term memory, guided filter, hyperspectral image classification;

1. Introduction

In the past few decades, hyperspectral image has been widely used in various applications, such as land cover, environmental protection, agriculture, and so on. As a common technique in the above applications, HSI classification has attracted increasing attention and become a hot topic in the remote sensing community. However,

* Corresponding author.
E-mail address: 1335395@qq.com
The firt two authors contributed equally

Selection and peer-review under responsibility of the scientific committee of the 2017 International Conference on Identification, Information and Knowledge in the Internet of Things (IIKI2017).
10.1016/j.procs.2018.03.048

there are two critical problems in the HSI classification, curse of dimensionality and limited number of labeled training samples.

The task of HSI classification is to categorize the pixels into one of several classes by their spectral features. And a large number of pixel-wise classifiers have been proposed to deal with the HSI classification, including random forests [6], k-Nearest Neighbor [7], support vector machine(SVM)[8], and sparse representation[9]. However, the majority of these traditional methods suffer from the Hughes phenomenon [10]. To sovle the aforementioned problem, feature extraction and feature selection are adopted in these methods. Generally, principal component analysis (PCA) [11] and independent component analysis (ICA) [12] are common methods in feature extraction. And band selection [13] or subspace projection techniques [14] are actively employed in classification of spectral patterns. Although these methods have improved classification accuracy, both feature extraction and subspace projection can lead to information loss and can not make full use of hyperspectral features.

In recent years, deep learning has made promising achievements in many fields, due to its powerful ability of feature learning. Deep learning based methods also are adopted in HSI classification,.including the autoencoder [15,18], convolutional neural work (CNN) [16] and deep belief network (DBN) [4]. The paper [15] proposed a deep learning framework for HSI classification. Autoencoder learns to reconstruct the input vector and reduce the dimension of spectrum. Then a multi-class logistic regression was used to classify the HSI. CNN uses extensive parameter-sharing to tackle the curse of demensionality, and has been employed to classify hyperspectral data directly in the spectral domain. Hu et al. [3] proposed a five-layers network, which can achieve better classification performance. Chen et al. [1] proposed a regularized 3-D CNN-based feature extraction model to extract efficient spectral-spatial features for HSI classification. To improve the classification performance further, spectral-spatial classification was proposed by many researchers which combines spatial context with spectral information. However, this paper does not cover this aspect.

Autoencoder and CNN model obtain better classification accuracy in hyperspectral image, owing to their feature representation. Yet, there are a large number of parameters to be trained in the CNN model. For a hyperspectral image with only a small number of labeled samples, the advantages of CNN model can not be fully exploited. Moreover, all the aforementioned methods view the spectrum as a vector, and have a loss of information. What’s more, they think the spectrum of a pixel is independent of each other. A pixel is considered a point in an orderless feature space. However, hyperspectral data can be seen as a continuing spectra sequences in continuous spectrum bands. Recurrent neural network, as a typical deep learning model, is ideal for solving sequential problems, and have less parameters than CNN model. So, we make use of a recurrent neural network (RNN) to characterize the sequential property for classifying the HSI.

2. Related Methodology and Work

2.1. LSTM Neural Networks

An improved RNN model, named long short-term memory (LSTM) [5], is proposed for the above problem. The key to LSTM is the cell state, which takes the former state and current data as input. And LSTM consists of five parts, including input gate, output gate, forget gate, cell input and cell output. The calculation process is as follows.

First, we compute the values for the input gate i_t, and the candidate value $\widetilde{C_t}$ for the states of the memory cells at time t:

$$i_t = \sigma(W_i x_t + U_i h_{t-1} + b_i) \tag{1}$$

$$\widetilde{C_t} = \tanh(W_{ic} x_t + U_c h_{t-1} + b_c) \tag{2}$$

Then, we compute the forget gate activation f_t at time t:

$$f_t = \sigma(W_f x_t + U_f h_{t-1} + b_f) \tag{3}$$

Given the value of the input gate activation i_t, the forget gate activation f_t and the candiate state value $\widetilde{C_t}$, we can compute C_t the memory cells’ new state:

$$C_t = i_t * \widetilde{C_t} + f_t * C_{t-1} \tag{4}$$

Finally, we can compute the value of their output gates and their outputs:

$$o_t = \sigma(W_o x_t + U_o h_{t-1} + V_o C_t + b_o) \tag{5}$$

$$h_t = o_t * \tanh(C_t) \quad (6)$$

Where x_t is the input to the memory cell layer at time t, W terms, U terms and V terms denote the weight matrices, b terms are bias vectors, σ is the activation function.

2.2. Guided filter

Given a guidance I and an input image p, we can obtain an output image q by guided filter. Generally, q is a linear transform of I in a window ω_k centered at the pixel k. If the radius of k is r, the size of local window$\omega_k$is $(2r+1)\times(2r+1)$.

$$q_i = a_k I_i + b_k, \forall i \epsilon \omega_k \quad (7)$$

Where a_k is linear coefficient and b_k is a bias. From the model, it is obvious that $\nabla q = a\nabla I$, which means that the filtering output q will have similar edge with guidance image I.

To obtain the coefficient and bias, a minimum cost function in the window ω_k is applied as follows:

$$E(a_k, b_k) = \sum_{i\epsilon\omega_k}((a_k I_i + b_k - p_i)^2 + \epsilon a_k^2) \quad (8)$$

Here, ϵ is a regularization parameter which could affect the blurring for the guided filter. According to the literal [2], formula (10) leads to a solution as follows.

$$a_k = \frac{\frac{1}{\omega}\sum_{i\epsilon\omega_k} I_i p_i - \mu_k \overline{p}_k}{\sigma_k^2 + \epsilon} \quad (9)$$

$$b_k = \overline{p}_k - a_k \mu_k \quad (10)$$

Where μ_k and σ_k^2 are the mean and variance of I in ω_k, $|\omega|$ is the number of pixels in the local window, and $\overline{p}_k$ is the mean of p in the window. After obtaining the coefficienta_k,b_k, we can compute the filtering outputq_i. Through the above process, we can get a linear transform image q.

3. The analysis and modeling of HSI classification

3.1. Filerting the HSI with guided filter

From the figure 1 (especially class-1), we can see that the regularity with spectral data is obvious. However, there are still a lot of noise, inconsistent with the overall trend. To solve this problem, we introduce guided filter to smooth the noise. Guided filter is a edge-preseving filter, which can be used for edge-aware smoothing. In this paper, the spectral data in the figure 1 were converted into the data in the figure 2 by guided filter.

First, we need to obtain a guidance image by Principal Component Analysis (PCA). We take the first three principal components as a color guidance image. Given a data set $D = \{d_1, d_2, \cdots, d_S\}$, we adopt PCA to obtain the following result. Here d_i is the information of the ith band, and S denotes the number of bands.

$$[p_1, p_2, \cdots, p_S] = PCA(D) \quad (11)$$

So, the guidance image is $G = [p_1, p_2, p_3]$. Then,the based on formula (3,4) using input image$d_1$and guidance image G, we can get the filtering output u_1. By the same way, we can yield all the $\mathbf{d}_i$, and construct a new hyperspectral image $U = \{u_1, u_2, \cdots, u_S\}$.

3.2. RNN model for HSI classification

The effect of machine learning algorithms is influenced by many factors, such as normalization, modeling, and optimization function etc. We investigated the influence of different normalization methods on the HSIclassification accuracy. We adopt min-max normalization to normalize the raw data to [0,1]. The normalization formula is $X_{norm} = (X - X_{min})/(X_{max} - X_{min})$, where X denotes the raw data, X_{min} and X_{max} are minimum and maximum values, respectively.Experimental results show that the normalizationin the spectral data of a pixel is better than that of the spectral band. We also found that the Adam is faster and better than stochastic gradient descent (SGD) optimization method.

Moreover, data modeling plays an important role in the HSI classification. We improved the LSTM method (named GF-LSTM) and replaced the 1-dimension input with 5-dimension inputs. 1-dimension modeling method can get more details of the spectral sequence. However, due to the complexity of spectral features, it is difficult to achieve stable state. We analyze the influence of different input dimensions on stability and result, and choose 5-dimension input as data modeling method, which can achieve better classification accuracy, and easier to achieve stable state.

4. Experiments and results

4.1. Experimental Setup

- Data Sets

Two hyperspectral data, including Indian Pines and Kennedy Space Center (KSC), are employed to evaluate the effectiveness of the proposed method.

- Evaluation metrics

OA shows the number of hyperspectral pixels that are classified correctly, divided by the number of all test samples. AA is the mean of the classification accuracies of all classes. KA is a statistical measurement of agreement, based on the confusion matrix of different classes.

- Parameter settings

In our experiment, the parameters of guided filter and LSTM model are need to be set. There are two key factors to influence the result of guided filter, such as radius r and regularization parameter . Radius r is used to express the range of smooth. And is used to control the degree of smooth. We set r=3 and =0.001 in this paper.

For LSTM model, we set three layers to train, input layer, hidden layer and output layer. The size of hidden layer is 200, and the input size is 5. For Indian Pines data set, the size of time step is 40. And the size of time step in the KSC data set is 35. That is, how many times it carries out the forecast.

4.2. Experimental results

The first experiment was conducted on the Indian Pines data set. We compared the performance of the methods using the quantitative index of OA, AA and KA. The detailed results in our experiments are shown in Table 1. It can be seen that the result of SVM outperform the result of Autoencoder and LSTM, with 10% samples. The value of AA (89.04%) in the SVM is higher than the value of OA (83.84%). That means that the class with few samples has high precision, and the SVM is more suitable for classification with small samples. On the contrary, the value of AA in other three methods is lower than OA. This denotes that deep learning methods have excellent performance in the class with a large number of samples. Although the results of LSTM are lower than those of SVM and autoencoder, the GF- LSTM is far superior to other methods. Compared with LSTM, GF-LSTM has improved by about 15%.

Table 1. Classification accuracy on the Indian Pines data set (10% samples for training)

	SVM [35]	Autoencoder[15]	LSTM	GF-LSTM
OA	83.84	81.65	76.31	90.93
AA	89.08	75.60	72.69	86.34
KA	81.64	79.06	73.73	89.54

The second experiment is performed on the KSC data set. The global and individual classification accuracies for different methods are shown in Table 2. Because the number of the KSC data set is much lager than Indian Pines data set, it is more suitable for deep learning methods. Therefore, the results of autoencoder, LSTM, GF-LSTM are better than that of SVM. GF-LSTM was very exciting, 30% higher than the SVM in all three indicators. Compared with LSTM, GF-LSTM also has a 10% improvement. All these prove that our proposed methods are very effective for HSI classification.

Comparing all the methods on the two data sets, we can observe that the two results of SVM are very different. This leads to the conclusion that SVM is more suitable for little amount of samples. However, GF-LSTM performs well on two data sets, which is suitable for complex problem. Since the parameters of GF-LSTM are less, and the correlation among the spectral information is taken into account, GF-LSTM has better performance than Autoencoder and CNN for HSI classification, which is a complex problem with a small number of labeled samples.

Table 2. Classification accuracy on the KSC data set (10% samples for training)

	SVM [35]	Autoencoder[15]	LSTM	GF-LSTM
OA	61.72	67.05	85.51	93.80
AA	58.49	59.55	80.87	91.81
KA	58.80	63.33	83.87	93.09

5. Conclusion

In this paper, we proposed a guided filter based RNN model for HSI classification, under the assumption that the spectral data can be regarded as an ordered sequence. Compared with SVM classifier and autoencoder model, the proposed model could achieve higher accuracy at all the experimental data sets, with a 10% samples to train.

Our work is an exploration of using RNN for HSI classification and has excellent performance. The modeling of HSI classification by RNN is a guidance and plays a useful reference.

Acknowledgements

This research is supported by Shaanxi Natural Science Foundation (No.2017JQ6074), Fundamental Research Funds for the Central Universities (GK201603091), National Natural Science Foundation of China(61741208), and Fund for Integration of Cloud Computing and Big Data (2017A07053).

References

[1]. Chen Y, Jiang H, Li C, et al. Deep feature extraction and classification of hyperspectral images based on convolutional neural networks[J]. IEEE Transactions on Geoscience and Remote Sensing, 2016, 54(10): 6232-6251..

[2]. Friedman J, Hastie T, Tibshirani R. The elements of statistical learning[M]. Springer, Berlin: Springer series in statistics, 2001..

[3]. Hu W, Huang Y, Wei L, et al. Deep convolutional neural networks for hyperspectral image classification[J]. Journal of Sensors, 2015, 2015.

[4]. Chen Y, Zhao X, Jia X. Spectral–spatial classification of hyperspectral data based on deep belief network[J]. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, 2015, 8(6): 2381-2392.

[5]. Graves A. Supervised sequence labelling with recurrent neural networks[M]. Heidelberg: Springer, 2012.

[6].Dalponte M, Orka H O, Gobakken T, et al. Tree species classification in boreal forests with hyperspectral data[J]. IEEE Transactions on Geoscience and Remote Sensing, 2013, 51(5): 2632-2645.

[7].Ma L, Crawford M M, Tian J. Local manifold learning-based $ k $-nearest-neighbor for hyperspectral image classification[J]. IEEE Transactions on Geoscience and Remote Sensing, 2010, 48(11): 4099-4109.

[8].Melgani F, Bruzzone L. Classification of hyperspectral remote sensing images with support vector machines[J]. IEEE Transactions on geoscience and remote sensing, 2004, 42(8): 1778-1790.

[9].Chen Y, Nasrabadi N M, Tran T D. Hyperspectral image classification via kernel sparse representation[J]. IEEE Transactions on Geoscience and Remote sensing, 2013, 51(1): 217-231.

[10].Hughes G. On the mean accuracy of statistical pattern recognizers[J]. IEEE transactions on information theory, 1968, 14(1): 55-63.

[11].Fauvel M, Chanussot J, Benediktsson J A. Kernel principal component analysis for the classification of hyperspectral remote sensing data over urban areas[J]. EURASIP Journal on Advances in Signal Processing, 2009, 2009: 11.

[12].Wang J, Chang C I. Independent component analysis-based dimensionality reduction with applications in hyperspectral image analysis[J]. IEEE transactions on geoscience and remote sensing, 2006, 44(6): 1586-1600.

[13].Feng J, Jiao L C, Zhang X, et al. Hyperspectral band selection based on trivariate mutual information and clonal selection[J]. IEEE Transactions on Geoscience and Remote Sensing, 2014, 52(7): 4092-4105.

[14].Bioucas-Dias J M, Nascimento J M P. Hyperspectral subspace identification[J]. IEEE Transactions on Geoscience and Remote Sensing, 2008, 46(8): 2435-2445.

[15].Chen Y, Lin Z, Zhao X, et al. Deep learning-based classification of hyperspectral data[J]. IEEE Journal of Selected topics in applied earth observations and remote sensing, 2014, 7(6): 2094-2107.

[16].Chen Y, Jiang H, Li C, et al. Deep feature extraction and classification of hyperspectral images based on convolutional neural networks[J]. IEEE Transactions on Geoscience and Remote Sensing, 2016, 54(10): 6232-6251.