J Inf Process Syst,
http://dx.doi.org/10.3745/JIPS

ISSN 1976-913X (Print)
ISSN 2092-805X (Electronic)

JOURNAL OF INFORMATION PROCESSING SYSTEMS JIPS

Automated Classification of Spectra based on Enhanced Multi-scale Coded Convolutional Neural Network

Bin Jiang*, Haozhan Sun*, Yunpeng Zhong*, Qingwei Li*, Shuhua Cao*,
Donglai Wei*, Jiyu Wei*, Weihong Wang*, Hao Huang*

Abstract

With the advent of astronomical big data era, spectral data classification is of great significance for astronomical data processing. Feature extraction is an important part in the process of spectra automatic classification. The best classification network has the ability to extract most of the common spectral features without noises and individual features. Such a network has better generalization capabilities and can extract sufficient features for classification.

This paper aims to use a variety of classification networks of 1D and 2D to classify spectral data in two dimensions. One-dimensionalization is performed on the current advanced 2D classification network, and folding processing is performed on the 1D spectral data. The aim is to explore the feasibility of using one-dimensionalized 2D network to classify 1D data and the feasibility of carrying on classification on the 2D data converted from 1D with 2D network.

The experimental results show that the fully connected neural network cannot extract enough features. Although the convolutional neural network (CNN) with strong feature extraction capability (e.g. VGGNet and ResNet) can quickly achieve satisfactory results on the training set, we found that CNN is vulnerable to over-fitting, which cannot be solved during our experiments. Spectral data with different signal-to-noise ratios (SNR) also have an effect on the performance of classification of the same network. To solve the problems above, the Enhanced Multi-scale Coded Convolutional Neural Network (EMCCNN) is designed, which can perform denoising on spectral data, and feature extraction of denoised spectral data at different scales. The EMCCNN network also combines features in different scales to improve the performance of feature extraction, which can maximize the ability to extract common features, and minimize noises and individual features. The thorough experimental results demonstrate that EMCCNN outperforms other well-known standard classification models such as ResNet or VGG in terms of classification performance.

Keywords

Multi-scale, Coder, Spectral Classification, Convolutional Neural Network

1. Introduction

The Sloan Digital Sky Survey (SDSS) [1] began in year 2000 has created the most detailed three-

Manuscript received Month 06, 20XX; accepted Month 06, 20XX.
Corresponding Author : Hao Huang (huanghao_sdu@mail.sdu.edu.cn)
*School of Mechanical, Electrical & Information Engineering in Shandong University, Weihai, China (jdxy@sdu.edu.cn)

dimensional maps of the universe with multi-color images of one third of the sky. SDSS has produced massive spectra for more than three million astronomical objects. The current survey of SDSS is SDSS-IV. APOGEE-2 (The APO Galactic Evolution Experiment) is a stellar spectroscopic survey of the Milky Way with two major components: a northern survey using the bright time at APO (Apache Point Observatory), and a southern survey using the 2.5m du Pont Telescope at Las Campanas.

SDSS provides a wealth of valuable data for spectroscopy research. Accurate and efficient classification of acquired spectral images is an important task in astronomical data processing. Higher requirements are put forward for machine learning methods and theories. In this paper, a novel and robust method is proposed and implemented based on EMCCNN. Massive M type dwarf stellar spectra from SDSS-DR14[2] is classified by the method.

Spectral data has different SNRs, and data with different SNRs often leads to different results with the same method. This paper explores the influence of SNR on classification accuracy under various methods and SNR on classification accuracy under different dimensions of the same method. It also implemented different classification network structures under high and low SNR conditions, which greatly improves the classification accuracy. Finally, the paper gives an in-depth discussion on the degree of feature extraction and overfitting of spectral classification, and proposes corresponding solutions.

2. The related work

In 2014, Pavel firstly introduced a multi-layer convolutional neural network based on the learnable kernel into the field of spectral classification [3]. He tried to fold the 1D spectral image, but did not achieve satisfactory results. Wen-Yu W et al. attempted to use the 1D convolution network to automatically identify the white dwarf main sequence double star (WDMS) [4], which proved that the 1D convolution network plays a significant role in the spectral classification field.

Kurtz proposed the cross correlation and principal component analysis (PCA) method, applied the MK classification method to the low-resolution spectrum (1.4nm), and the classification accuracy of the spectrum of b0~m2 was 2.2 spectral subtypes [5]. LaSala used the nearest neighbor method to classify the spectra of 350 B-type stars, with an average error of 1.14 spectral subtypes [6]. Hippel used two-layer BP (backpropagation) network to train 575 b3~m4 type star samples, and the RMS (root mean squared-RMS) error of classification was 1.7 spectral subtypes [7]. Gulati et al. used multilayer feedback network (mb-pn) to divide the spectra of 158 stars into 55 types. The classification accuracy was 2 spectral subtypes, and the accuracy of luminosity type was 64% [8]. XUE uses SOFM (self-organizing feature map, SOFM) neural network to classify 7 types of stars and their subtypes [9], which is an unsupervised nonlinear learning method and both computation and speed need to be optimized.

In the LAMOST pilot survey data, Bin Jing discovered two unrecognized cataclysmic variable stars (CVs) by data mining [10], and Peng Wei discovered a batch of stars with spectral anomalies by using the outlier method [11]. Yinbi Li applied machine learning methods in LAMOST DR4 released data and found 2,651 carbon star test data. Yinbi Li divides these carbon stars into five subtypes, C-H, C-R, C-J, C-N and Ba, and further divides C-J carbon stars into three types [12]. Nicholas et al. made a systematic summary and induction of data mining and machine learning in astronomical research [13].

With the continuous development of deep learning technology, CNN is giving excellent performance for feature extraction and combination. There are many outstanding models in the field of image classification. As a classic deep convolutional network, VGGNet [14] is widely used in image classification tasks. ResNet [15] is the network structure proposed by Kaiming He in 2015. Its innovative use of residual learning allows deep networks to extract more effective features. DenseNet [16] firstly proposed that the output of each layer is the input of the next layer, allowing the network to extract features of more dimensions. For the spectral classification task, it is actually extracting different types of spectral features and this paper verifies the availability of the networks above for spectral classification tasks in two dimensions.

3. The experimental data and preprocessing

The experimental spectra with the total number of 3,136,427 are selected from SDSS-DR14 with different SNR distribution in datasets preparation. Table 1 shows the description of the original SDSS experimental data with preliminary classification provided by the pipeline of SDSS.

Table.1.Description of DR14

Classification	Number
Galaxies	2,401,952
Quasars	477,161
Stars	851,968
Sky	341,140
Unknown	200,490

The wavelength of the spectra is normalized to 3,800Å~9,000Å(Angstrom) and the flux of each original spectra is normalized as follows in data preprocessing:

$$y = \frac{(x - x_{i+1}) \cdot (x - x_{i+2})}{(x_i - x_{i+1}) \cdot (x_i - x_{i+2})} \cdot y_i + \frac{(x - x_i) \cdot (x - x_{i+2})}{(x_{i+1} - x_i) \cdot (x_{i+1} - x_{i+2})} \cdot y_{i+1} + \frac{(x - x_i) \cdot (x - x_i)}{(x_{i+2} - x_i) \cdot (x_{i+2} - x_{i+1})} \cdot y_{i+2} \quad (1)$$

Using the method above, we increase the 3,522 dimensional data to 5,000 dimensions, and fold it into a 50*100 spectral matrix as shown in figure 1.

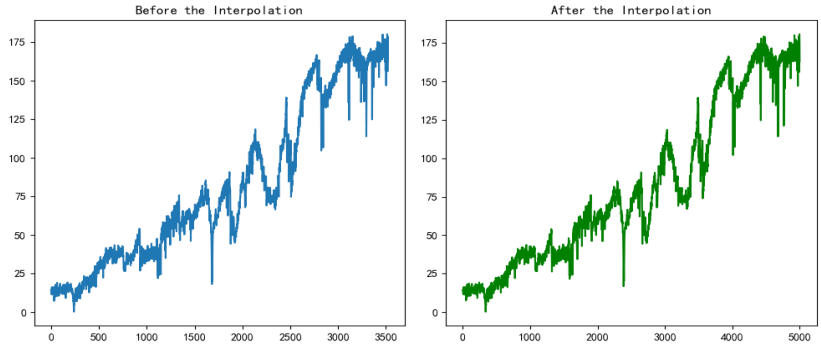


Fig.1. Comparison of spectral data before and after interpolation

SNR and spectral type of spectra are two important influence factors on the classification and are analyzed and discussed in our experiment. The spectral distribution of spectral type of the experimental spectra are shown in figure 2. The standard error of the temperature is shown to describe the distribution of the stellar type in detail.

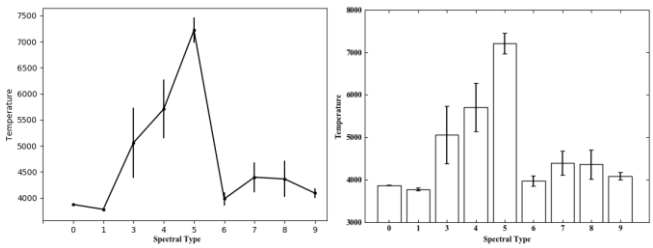


Fig.2. Spectral type and distribution

The data is classified according to the SNR and the type of M dwarfs. The SNR is divided into 5-10, 10-15, and above 15. The type of M dwarfs is divided into five subclasses from M0~M4. The specific data is shown in figure 3.

Fig.3. Distribution of the M dwarf dataset

4. Method

The goal of spectral classification is to maximize common features and minimize noises and spectral individual features. Therefore, the effect of spectral classification *P* should satisfy the following formula:

$$P_{bset} = \bigcup_{i=1}^{n} MAX(C_i) + MIN(S_i + N_i) \tag{2}$$

Where *P* is the final effect of spectral classification and *C* stands for common feature. *S* and *N* represent individual features of spectrum and noises.

4.1 Convolutional Neural Network (CNN)

We use CNN to classify M dwarfs, because CNN performs well in image classification, and it can also handle 1D classification work. The feature extraction capabilities of CNN can help us to improve the accuracy of traditional classification methods. 1D CNN is adopted in the experiment since M dwarfs spectra in the SDSS dataset are stored in 1D vectors. Considering the good performance of the CNN for image, quadratic interpolation is performed on the data to facilitate the folding of the spectral data to form a 2D spectral matrix.

As a special form of CNN, 1D CNN has certain applications in the field of signal processing. We adopted the network structure shown in figure 4.

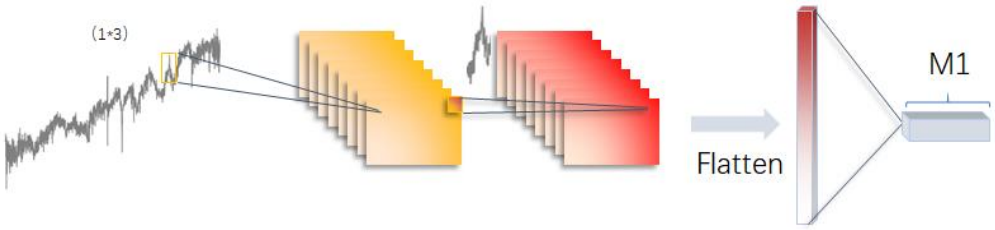


Fig.4. The network structure of CNN

In the specific network construction, the backpropagation algorithm is adopted. The 1D matrix information composed of spectral data is continuously trained to fit the parameters in each layer by gradient descent. Our loss function can be defined as the mean square error：

$$E = \sum \frac{1}{2}(\text{target} - \text{output})^2 \tag{3}$$

Using the total errors to calculate the partial derivative of the parameter, the magnitude of the influence of a parameter on the overall errors can be obtained, which is used to correct the parameters in the back propagation. Since the construction of the network uses a linear arrangement, the total error calculated by the final output layer is used to perform the partial derivative calculation of the parameters in all layers.

$$\frac{\partial E}{\partial w_{xj}} = \frac{\partial E}{\partial Out_n} * \frac{\partial Out_n}{\partial Net_n} * \dots\dots * \frac{\partial Out_x}{\partial Net_x} * \frac{\partial Net_x}{\partial w_{xj}} \tag{4}$$

In a CNN, each neuron is only connected to the neurons of the previous layer, so the calculation of the local gradient requires a forward recursive calculation of the gradient of each subsequent layer of neurons. After defining the linear output of each layer and the parameters in the structure, the output of the activation function is: Out=φ(v), then the recursive formula for each gradient can be derived as:

$$\delta_{ij}^{l-1} = \frac{\partial E}{\partial v_{ij}^{l-1}} = \frac{\partial E}{\partial Out_{ij}^{l-1}} * \frac{\partial Out_{ij}^{l-1}}{\partial v_{ij}^{l-1}} = \frac{\partial E}{\partial Out_{ij}^{l-1}} * \varphi'\left(v_{ij}^{l-1}\right) \tag{5}$$

In the successive calculation of the formula above, the gradient calculated by each parameter under the total errors is used as the basis to achieve the goal of correcting the parameters in the direction of the minimum loss function.

4.2 VGGNet and ResNet

VGGNet is widely used in 2D data. VGGNet's generalization ability for different datasets is very prominent, so we aim to make the VGG network one-dimensional, and also try to fold the spectral data. Considering that our spectral dimensions are not particularly large, in order to prevent overfitting on the training set, the shallow VGG16 network is used in VGGNets in our experiments.

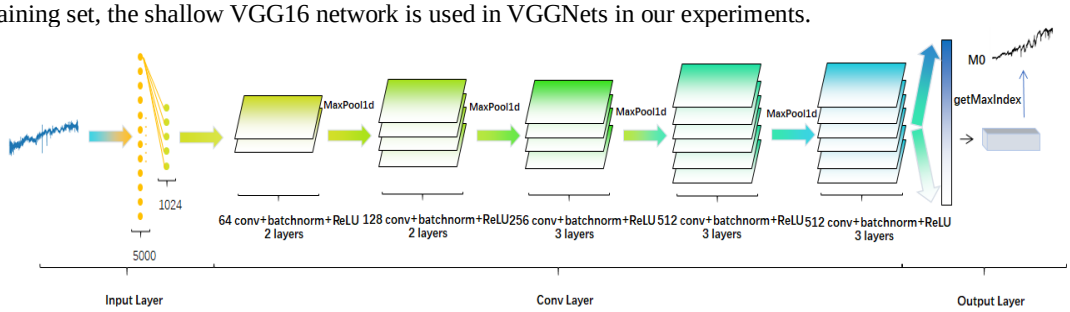


Fig.5. The network structure of VGG16

The greatest improvement of VGGNet is to convert large and short convolutional layers into small and deep neural networks by reducing the scale of convolution kernels. In the convolutional layer, the number of extracted features for each layer output can be calculated using the following formula:

$$n_{out} = \left(\frac{n_{in} + 2p - k}{s}\right) + 1 \tag{6}$$

With the continuous development of the classification network, the network is constantly deepening for better extraction and feature combination, but the deep network makes it very difficult to train and has a potential of losing information. ResNet proposes residual learning to solve these problems, allowing information to be transmitted over the layer to preserve information integrity, while learning only the residuals of the previous network output:

$$F(x) \coloneqq H(x) - x \tag{7}$$

This allows us to increase the number of convolutional layers, which is a good way to train a very deep network.

The most prominent characteristic of the ResNet is that in addition to the result of the conventional convolution calculation in the final output, the initial input value is also added. Therefore, the result of network fitting will be the difference between the two, thereby we can obtain the calculation formula of each layer of ResNet is as follows:

$$Out^l = \varphi(z^l + \mathrm{Out}^{l-1}) = \varphi(w^l \mathrm{Out}^{l-1} + b^l + \mathrm{Out}^{l-1}) = \varphi[(w^l + 1)\mathrm{Out}^{l-1} + b^l] \tag{8}$$

Then the gradient calculation formula of the neural network is changed on the basis of conventional ones:

$$\delta_{ij}^{l-1} = \frac{\partial E}{\partial Out_{ij}^{l-1}} * \left[1 + \varphi'\left(v_{ij}^{l-1}\right)\right] \tag{9}$$

Compared with the traditional network, the extra value "1" makes the calculated gradient value difficult to disappear, which means the gradient calculated from the last layer can be transmitted back in the reverse direction, and the effective transmission of the gradient makes spectral features more efficient in the training of neural networks. However, considering that our spectral features are limited and high noises interference, the efficient feature extraction of the network may lead to significant overfitting, which makes the trained network generalization capability poor.

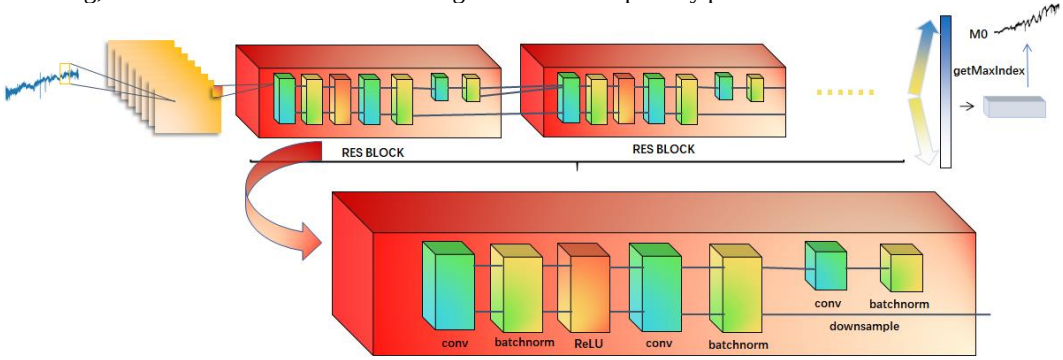


Fig.6. The network structure of ResNet

We previously suppose that the residual network is the leading classification network in the field of image classification. It is worth expecting the effect of spectral classification after one-dimensionalization.

4.3 EMCCNN

The Enhanced Multi-Scale Coding Convolutional Neural Network (EMCCNN) is a network designed for the characteristics of spectral data, and has achieved good results on different SNR data. We find that for spectral data, a deep convolutional layer can lead to significant overfitting of the data and a very poor generalization capability of the model. Directly convolution of spectral data will extract a lot of noises and individual features, which is not an ideal way to classify spectral data. Unsupervised denoising methods tend to remove some spectral feature peaks in the data, which is also not an ideal method to maximize the common features of the spectral type.

Therefore, we try to add a supervised denoising network before the convolutional neural network,

which can help the identification of features and noises. It allows the network to extract spectral features instead of noises. On the other hand, the spectral feature peak types are not consistent. For different types of feature peaks, the different convolution kernel sizes may be able to extract different quality features. The extraction of some features may be better for larger convolution kernels. Others may be more friendly to small-scale convolution kernels. We decide to let convolution kernels of different scales learn features simultaneously, then combine these features, and obtain different weights for features through the fully connected layer. A better feature extraction network EMCCNN can be obtained.

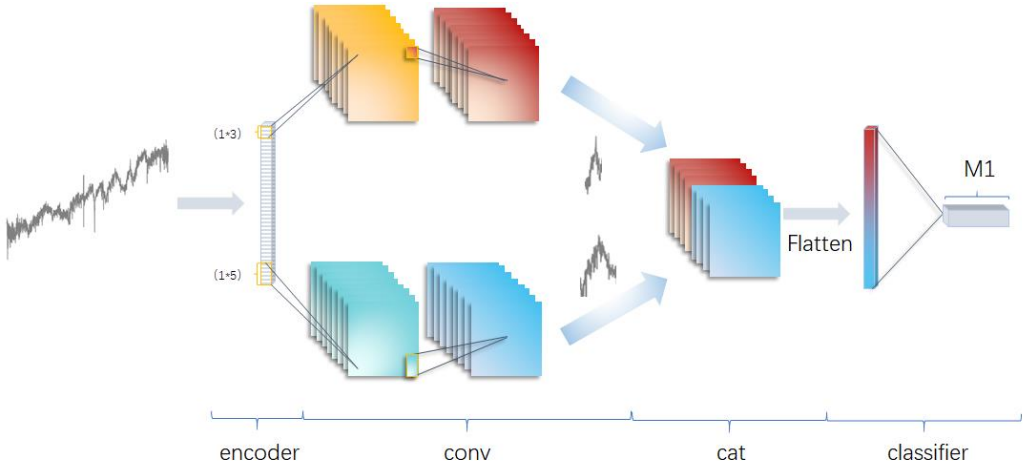


Fig.7. The structure of Enhanced Multi-scale Coded Convolutional Neural Network

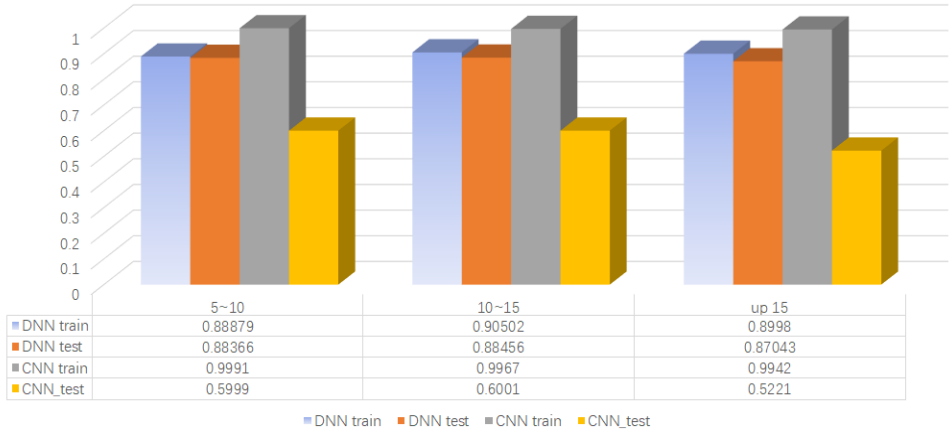
5. Experiment

5.1 Experimental process and results

5.1.1 1D convolution experiment and results

We firstly try to compare the spectral classification of the four hidden layers of DNN and 1D CNN. The optimizer used was a random gradient descent, the learning rate was set to 1e-4, and the activation function chosen is ReLU. We find that the CNN network can quickly extract the corresponding features, but it will also overfit quickly, which has an accuracy of 60% in the test set, and the generalization capability is extremely poor.

CNN has strong feature extraction capabilities, but it is not suitable for spectral data because it also extracts quite a lot of individual features and noises. The problem becomes even worse with the increme nt of SNR.



	5~10	10~15	up 15
DNN train	0.88879	0.90502	0.8998
DNN test	0.88366	0.88456	0.87043
CNN train	0.9991	0.9967	0.9942
CNN_test	0.5999	0.6001	0.5221

Fig.8. Comparison of classification accuracy between DNN and CNN

Considering that VGG16 is composed of several convolutional layers, the direct convolution of the spectral data will be the same as the direct use of CNN, leading to the overfitting of the training data. Hence we try to add the encoder to the VGG16 network to denoise. We found that after denoising, VGG16 is not as bad as CNN for data within the scope of $5 \leq sn \leq 15$, but it is also not satisfactory for data with $sn > 15$. It is considered that the data features of high SNR are more obvious. The spectral individual features and noises will have a strong interference effect on a very deep network. On the other hand, the feedback process of the denoising encoder is too long, which is not easy to identify the difference of individual features, noises and common features.

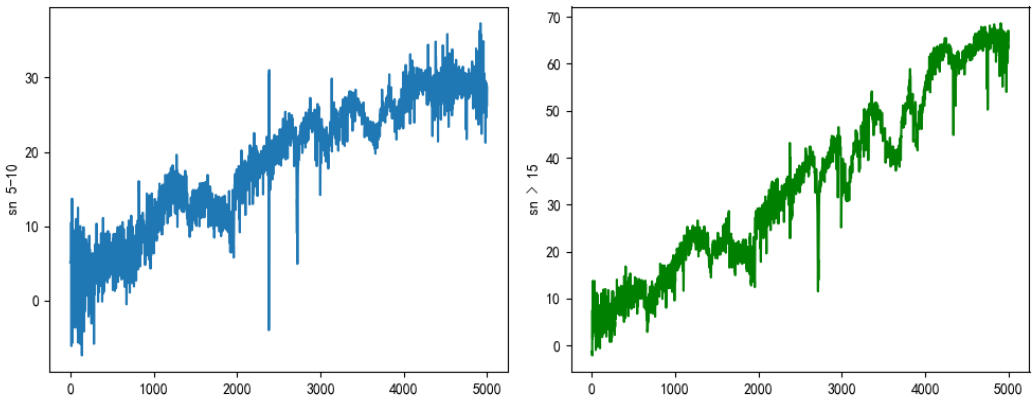


Fig.9. Spectral contrast of different signal-to-noise ratios (SNR)

ResNet's interlayer transfer mechanism might help us solve the problem of overfitting, but the effect is not very satisfactory. ResNet and CNN also perform well on the training data, and reached more than 99% accuracy on the three SNRs. It does not perform well in the test set, although not as serious as CNN. We try to add the dropout layer to help us solve the problem of overfitting, but this makes our training very slow and the final result is not satisfactory as well.

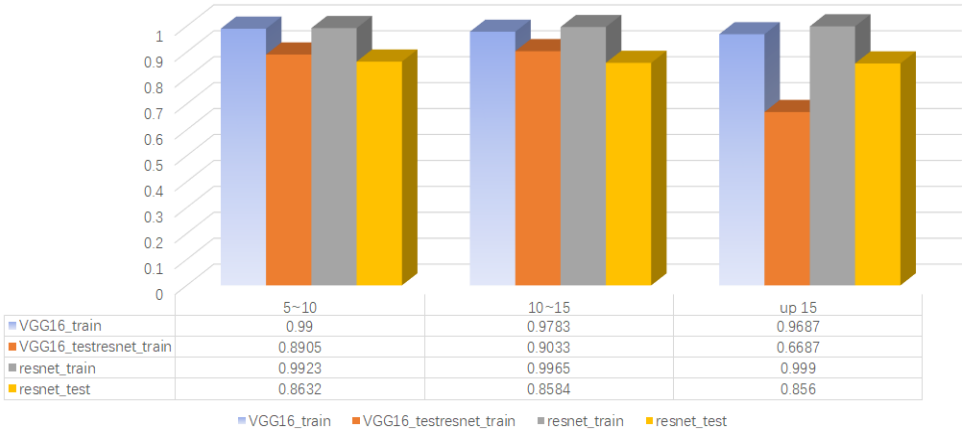


Fig.10. Comparison of classification accuracy between VGG16 and ResNet

Through experiments, it is noticed that the network with deep structure is not suitable for the classification of spectral data, which will make the individualized features of noises and spectrum fully extracted, and the denoising of the encoder before these networks will make the feedback time too long, leading to a bad denoising result. Based on the conclusions above, we propose EMCCNN, which is not too deep, and is beneficial to the training of denoising capability of the encoder.
We find that EMCCNN is very robust to SNRs, which proves the correctness of our ideas and is consistent with our expected experimental results.

Table.2. Classification accuracy of different network models

sn / model	5~10		10~15		up15	
	train	test	train	test	train	test
DNN	0.8888	0.8837	0.9050	0.8846	0.8998	0.8704
CNN	0.9991	0.5999	0.9967	0.6001	0.9942	0.5221
VGG16	0.9900	0.8905	0.9783	0.9033	0.9687	0.6687
ResNet	0.9923	0.8632	0.9965	0.8584	0.9990	0.8560
EMCCNN	0.9914	0.9217	0.9942	0.9277	0.9659	0.8916

Finally, we used SVM to directly process spectral data for classification and compared with the results of EMCCNN. The comparison demonstrates that SVM can quickly fit the training set of the data, but tends to overfit seriously.

Table.3. Classification accuracy of SVM and EMCCNN

sn / model	5~10		10~15		up15	
	train	test	train	test	train	test
SVM	1	0.6802	1	0.7213	1	0.741
EMCCNN	0.9914	0.9217	0.9942	0.9277	0.9659	0.8916

5.1.2 2D convolution experiment and results

We apply the classification network that is currently applied to 2D data in the field of spectral classification to explore the classification feasibility of the 2D folding of the 1D data. We fold the data into a 50x100 matrix and put it into a 2D classification network for experiments. The feature peaks of spectral data become inconspicuous after folding. The spectral data is only related on the same line after folding. The data correlation for each column is not obvious, but from the picture we can clearly see that each type of spectrum is different after folding, which brings the possibility of applying the 2D

classification network. We aim to fully apply the classification network so that satisfactory performance can be achieved.

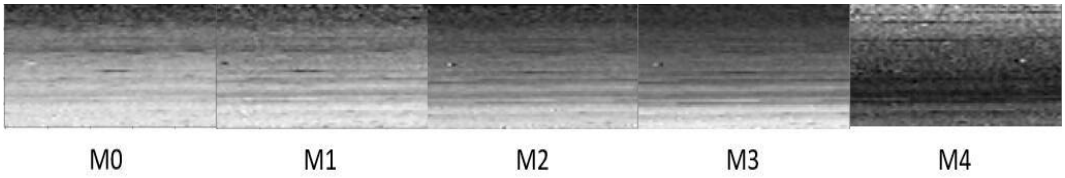


Fig.11. Spectral data after 2D folding processing

We conducted three network comparison experiments (CNN, VGG16 and Res18 respectively). We find that the 2D network and the 1D network each has its own merits on extracting spectral features and combination features. Among them, CNN overfits heavily on 1D network. Thus in comparison, the performance on 2D data has a significant improvement over 1D data.

Among them, VGG16 performs quite well in the range of 10<sn<15 SNR, and ResNet also performs better than 1D data in the range of 5<sn<15. This discovery is inspiring, because it proves that although 1D data is not directly related to the top and bottom after folding, but the 2D classification network can still achieve satisfactory results, which proves that the 2D classification of 1D data is feasible, even for some data can produce results better than 1D classification.

Table.4.2D-classification accuracy of different network models

sn / model	5~10		10~15		up15	
	train	test	train	test	train	test
CNN_1d	0.9991	0.5999	0.9967	0.6001	0.9942	0.5221
CNN_2d	0.9971	0.8288	0.9977	0.8271	0.9980	0.7310
VGG16_1d	0.9900	0.8905	0.9783	0.9033	0.9687	0.6687
VGG16_2d	0.9846	0.9103	0.9722	0.9320	0.9900	0.8504
ResNet_1d	0.9923	0.8632	0.9965	0.8584	0.9990	0.8560
ResNet_2d	0.9949	0.8738	0.9977	0.8801	0.9938	0.8132

6. Conclusions

There is a great need for accurate and automatic classification methods of specified objects in massive spectra. The goal that we propose spectral classification is to maximize the extraction of common features and to minimize the extraction of noises and spectral individual features, which enhances the generalization capabilities of the network. We also find that the simple DNN network cannot extract the features of spectra well, but the CNN with strong feature extraction capability can lead to significant overfitting. It also comes out that the deep network does not suit for the denoising training of the encoder, because its feedback process could be very long. This makes the encoder difficult to train and achieve good denoising result.

EMCCNN is not a deep network, which is ideal for the encoder to achieve good denoising result. On the other hand, convolution kernels of different scales can extract spectral peaks of different quality, which provides sufficient options for the classifier. By weighting the quality of features, the network can select high-quality spectral type features. This design makes EMCCNN achieve the best results in the three SNRs of 1D data.

Through the classification of 1D data with 2D folding, we can find that the 1D spectral data has relevance in 1D scale. Although it is not vertically related after folding, the 2D classification network can still achieve good results. This proves that the 2D classification of 1D data is feasible, especially for the case that the 1D classification extremely tends to overfit. In this case, folding the data into 2D can effectively prevent the tendency of overfitting.

Acknowledgement

This paper is supported by the National Natural Science Foundation of China (11473019), Shandong Provincial Natural Science Foundation, China (ZR2017MA046, ZR2018MA032).

References

[1] Abolfathi Bela, Aguado D S, AguilarGabriela, et al. The Fourteenth Data Release of the Sloan Digital Sky Survey: First Spectroscopic Data from the Extended Baryon Oscillation Spectroscopic Survey and from the Second Phase of the Apache Point Observatory Galactic Evolution Experiment. The Astrophysical Journal Supplement Series[J], Volume 235, Issue 2, article id. 42, 19 pp. (2018).

[2] Albareti Franco D, Allende Prieto Carlos, Almeida Andres,etal. The 13th Data Release of the Sloan Digital Sky Survey: First Spectroscopic Data from the SDSS-IV Survey Mapping Nearby Galaxies at Apache Point Observatory.The Astrophysical Journal Supplement Series[J], Volume 233, Issue 2, article id. 25, 25 pp. (2017).

[3] Pavel Hála. Spectral classification using convolutional neural networks[J]. Computer Science, 2014.

[4] Wen-Yu W , Ge-Lin G , Chun-Yu M A , et al. Extracting Convolutional Features of WDMS Spectra with Anti Bayesian Learning Paradigm[J]. Spectroscopy and Spectral Analysis, 2018.

[5] Kurtz MJ. Progress in Automation Techiques for MK Classification. In Garrison R F, Editor. Proceedings of the Workshop in Honour of Morgan WW, Keenan P C. David Dunlap Observatory, Toronto, Canada, 1984, 136.

[6] LasalaJ.In Chris Corbally, Gray RO, Garrison RFEditor. The MK Process at 50 Years. A Powerful Tool for Astrophysical Insight. Astronomical Society of the Pacific Conference Series, Proceedings of a Workshop of the Vatican Observatory, Tucson Arizona, USA, September, 2013, 312.

[7] Hippel T, Storrie-Lombardi L, Storrie-Lombardi MC, et al. Automated Classification of Stellar Spectra-Part One-Initial Results with Artificial Neural Networks. Monthly Notices of the Royal Astronomical Society, 2014, 269:97

[8] Gulati RK, Gupta R, Ran NK. Stellar spectral classification using automated schemes. Astronomy &Astrophysics, 2015, 322:933

[9] XUE Jianqiao, LI Qibin, ZHAOYongheng. Chinese Astronomy and Astrophysics, 2001, 25(1):120.

[10] Jiang Bin, Luo Ali, Zhao Yongheng, et al. Data mining for cataclysmic variables in the Large Sky Area Multi-Object Fibre Spectroscopic Telescope archive. Monthly Notices of the Royal Astronomical Society, 2013, 430, 986–995

[11] Peng Wei Ali Luo Yinbi Li Jingchang Pan Liangping Tu Bin Jiang Xiao Kong Zhixin Shi Zhenping Yi Fengfei Wang. Mining unusual and rare stellar spectra from large spectroscopic survey data sets using the outlier-detection method. Monthly Notices of the Royal Astronomical Society, 2013, 431, 1800–1811

[12] Yin-bi Li, A-Li Luo, Gang Zhao, et al.19 Low mass hyper-velocity star candidates from the first data release of LAMOST survey. Research in Astron. Astrophys, 2015,Vol. 15 No. 8, 1364–1377

[13] Nicholas M. Ball etc. Data Mining and Machine Learning In Astronomy, International Journal of Modern Physics, 2010,0:44

[14] SimonyanK , Zisserman A . Very Deep Convolutional Networks for Large-Scale Image Recognition[J]. Computer Science, 2014.

[15] He K , Zhang X , Ren S , et al. Deep Residual Learning for Image Recognition[J]. 2015.

[16] Huang G , Liu Z , Weinberger K Q . Densely Connected Convolutional Networks[J]. 2016.

Bin Jiang / https://orcid.org/0000-0002-2897-5745

He received D.S. degrees in University of Chinese Academy of Sciences. He started to work from May 2005 and now he is an associate professor in the Department of Computer Science in Shandong University, Weihai. His current research direction includes astronomical data mining and machine learning algorithms.

Haozhan Sun / https://orcid.org/0000-0001-6082-8166

He is receiving B.S (honors) degree from the Australian National University. His current research interests include natural language processing, network analysis, intelligent chatbot development and computer vision.

Yunpeng Zhong / https://orcid.org/0000-0002-4814-1692

He is a computer undergraduate in Shandong University (Weihai) . He is good at mathematics and has a strong interest in machine learning, physics and astronomy.

Qingwei Li / https://orcid.org/0000-0003-4197-3995

She is a junior undergraduate student majoring in Computer Science in Shandong University, Weihai. Her current research interests include machine learning algorithms and human-computer interaction techniques.

Shuhua Cao / https://orcid.org/0000-0001-5571-579X

She is a junior undergraduate student majoring in Computer Science in Shandong University, Weihai. Her current research interests include machine learning and human-computer interaction techniques.

Donglai Wei / https://orcid.org/0000-0002-5614-1633

He is a junior undergraduate student majoring in Computer Science in Shandong University, Weihai. His current research interests include deep learning and natural language processing.

Jiyu Wei / https://orcid.org/0000-0002-7837-3902
He is a freshman majoring in Software Engineering in Shandong University, Weihai. His current research interests include machine learning and data mining.

Weihong Wang / https://orcid.org/0000-0002-1067-360X
He is a sophomore undergraduate student majoring in Computer Science in Shandong university, Weihai. His current research interests include machine learning and data science.

Hao Huang / https://orcid.org/0000-0003-3826-5437
He is a junior undergraduate student majoring in Computer Science in Shandong University, Weihai. His current research interests include machine learning algorithms and computer vision.