Classification of Hyperspectral Remote Sensing Images Based on Three-Dimensional Convolutional Neural Network Model

Pan Zhao[1(✉)], Xiaoling Yin[1], and Shida Chen[2]

[1] Chizhou University, Chizhou 247000, China
zhaopan0827@126.com
[2] Shanghai Urban Construction Vocational College, Shanghai 200438, China

Abstract. In response to the problems of low accuracy and long time consumption in traditional hyperspectral remote sensing image classification methods, this paper proposes a hyperspectral remote sensing image classification method based on a three-dimensional convolutional neural network model. Firstly, the image data is preprocessed and normalized. Based on this, a three-dimensional convolutional neural network is introduced into the learning of image data. On this basis, by optimizing the overall connectivity parameters of the convolutional kernel function, hyperspectral remote sensing image classification based on the convolutional kernel function was achieved. Experiments have shown that the algorithm proposed in this article can accurately classify hyperspectral images and achieve good results.

Keywords: 3D Convolutional Neural Network Model · Normalization Processing · Hyperspectral Remote Sensing Images · Support Vector Machine

1 Introduction

Hyperspectral remote sensing is a means of analyzing and identifying the reflectance characteristics of the measured substance at different wavelengths. It has been widely used in environmental monitoring, geological exploration, agricultural production, and other fields. The acquisition, processing, and application of hyperspectral remote sensing images have been a research hotspot internationally in recent years. Hyperspectral remote sensing images are the process of synthesizing the information of ground objects in different bands such as visible light and infrared to form multi band images. Compared to conventional remote sensing images, hyperspectral images have higher spatiotemporal resolution and can provide more ground information. Hyperspectral remote sensing image classification is an important research topic in hyperspectral remote sensing images [1]. It is an important aspect of hyperspectral remote sensing image data, and its accuracy will directly affect the practical application of hyperspectral remote sensing images. Hyperspectral remote sensing image can provide abundant surface information due to its multi band, high spatial resolution, high Spectral resolution and other characteristics, and has important application value in land use, vegetation coverage, water

L. Yun et al. (Eds.): ADHIP 2023, LNICST 548, pp. 451–462, 2024.
https://doi.org/10.1007/978-3-031-50546-1_30

resources monitoring and other aspects. Hyperspectral remote sensing image classification refers to the classification of pixels in the image, including vegetation, water bodies, buildings, bare land, etc. The research results will provide scientific basis for land use, land cover, environmental monitoring, and resource management, as well as for agriculture, urban planning, and disaster risk assessment. However, currently hyperspectral remote sensing image classification still faces many challenges, such as high scene complexity and high image noise. Due to the similarity and confusion of a large number of pixels in the image, classification algorithms become more complex [2]. However, due to the frequent interference of clouds, shadows, and atmospheric disturbances in image data, the accuracy of image classification is low. In the context of massive hyperspectral data, achieving precise classification of remote sensing images remains a challenging issue. How to efficiently use hyperspectral data and classify it is crucial to improve.

In the classification of hyperspectral remote sensing images, researchers have used various methods to improve their classification accuracy. Reference [3] proposes a classification method for hyperspectral remote sensing images based on EMP and mixed kernel SVM. This method first effectively extracts spatial information through EMP, and then uses different kernel functions to process spatial and spectral information, ultimately completing hyperspectral image classification using mixed kernel SVM. This method has high classification accuracy for hyperspectral remote sensing images, but poor classification efficiency. Reference [4] proposes a hyperspectral remote sensing image classification algorithm based on multi-dimensional CNN. This algorithm integrates CNN from different dimensions and combines spatial and spectral information for hyperspectral remote sensing image classification. The comprehensive utilization of the abstract expression ability of three types of CNN for hyperspectral spatial spectral joint information has effectively promoted the application of CNN in the field of hyperspectral remote sensing image classification. This algorithm has good classification efficiency for hyperspectral remote sensing images, but poor classification accuracy.

In response to the problems of the above methods, this article intends to conduct research on hyperspectral remote sensing image classification based on convolutional neural networks. Compared with the traditional hyperspectral remote sensing image classification methods, this method introduces a three-dimensional Convolutional neural network model. Three dimensional Convolutional neural network can learn the spatial and spectral information of images at the same time, so as to capture image features more comprehensively. This method further improves classification accuracy by optimizing the overall connectivity parameters of the convolutional kernel function. The optimization of connectivity parameters can enhance the perceptual ability of convolutional kernels in hyperspectral remote sensing images, enabling them to better adapt to the feature distribution and classification requirements of images.

2 Hyperspectral Remote Sensing Image Preprocessing and Normalization Processing

The data preprocessing of hyperspectral remote sensing images is an important aspect that cannot be ignored in remote sensing data analysis. Firstly, hyperspectral images are composed of thousands of spectral bands, making data processing and storage difficult.

Moreover, hyperspectral image data often contains factors such as noise, atmospheric interference, and surface shadows, which can cause errors and affect the accuracy of the data. Therefore, it is necessary to preprocess hyperspectral remote sensing images to eliminate these effects and obtain more accurate and reliable information.

The preprocessing and normalization of hyperspectral remote sensing images are two important steps in hyperspectral data analysis. Preprocessing refers to the process of spatial correction, spectral correction, and denoising of hyperspectral images before analysis, in order to reduce the impact of errors and noise on the data and improve the quality and reliability of the data. Normalization processing is aimed at solving the problem of the difference in intensity values between different bands in hyperspectral data. A series of mathematical methods are used to convert hyperspectral image data into data at the same scale, making comparison and analysis between different bands more convenient and accurate.

2.1 Hyperspectral Image Data Preprocessing

For hyperspectral image classification algorithms, in order to achieve high classification accuracy while preserving the internal structure of the original data, it is necessary to find a low dimensional subspace composed of several end element spectral images, where each single pixel represents an actual feature. Using the dimensionality reduction method to construct linear graph M, ensuring that each adjacent pixel in the initial image maintains a relatively close projection distance and is always in the subspace. Thus, relevant information in local areas can be well preserved. For complex hyperspectral image classification structures, this can play a protective and constraining role, and effectively protect the diverse internal structure of the initial image [5]. Assuming the initial data of the training sample is:

Class $(x_{i,1}, x_{i,2}, ..., x_{i,j}..., x_{i,n})$, $x_{i,j} \in R^d$, is labeled as $y \in \{1, 2, ..., c\}$, n represents the total number of training samples, c represents the classification quantity value of training samples, and assuming n_l represents the l training sample number, there is $\sum_{l=1}^{c} n_l = n$. The intimate relationship between training sample x_i and training sample x_j is:

$$A_{i,j} = \exp\left(-\frac{\left\|x_i - x_j\right\|^2}{y_i - y_j}\right) \tag{1}$$

$$y_i = \left\|x_i - x_{i,m}\right\| \tag{2}$$

In the formula, y_i, y_j represents the number of samples, and $x_i x_j$ represents the m adjacent sample within local scale $x_{i,m}$.

On this basis, a new method that can effectively protect local information between pixels is proposed. It is necessary to use local dimensionality reduction methods to obtain a linear graph M through projection. Linear graph M not only effectively protects local information but also separates graph categories [6–8]. The formula for the scatter matrix between local information of adjacent pixels and the intra class scatter matrix is:

$$L_{lb} = \frac{1}{2}\sum_{i,j=1}^{n} W_{i,j}\left(x_i - x_j\right)\left(x_i - x_j\right)^T \tag{3}$$

$$L_{ho} = \frac{1}{2}\sum_{i,j=1}^{n} W_{i,j'}\left(x_i - x_j\right)\left(x_i - x_j\right)^T \tag{4}$$

In the formula, $W_{i,j}$ represents the close connection value between the training sample categories, and $W_{i,j'}$ represents the close connection value between the local categories of the training sample. Among them, the local inter category scatter matrix L_{ho} and the overall inter category scatter matrix L_{lb} are both $n \times n$-squared matrices.

According to formula (1), the close connection value between each adjacent pixel can be obtained [9], and the ratio of the maximum value obtained through local dispersion matrix is *Fisher*:

$$L_{lb}^T = \delta L_{ho}^T \tag{5}$$

In the equation, δ represents the diagonal characteristic value of the matrix, and the transformation matrix is:

$$T_{LFDA} = \arg\max_{T \in R^{d \times r}} tr\left[\left(T_{LFDA}^T L_{lb} T_{LFDA}\right)^{-1} T_{LFDA}^T L_{lb} T_{LFDA}\right] \tag{6}$$

In the equation, $tr(.)$ represents the motion trajectory function of the matrix.

This data preprocessing model has protective constraints and does not damage the diversified structure of the initial image. This model measures its constraint by the similarity between local adjacent pixels, utilizing the variable characteristics of matrix T_{LFDA} to make each individual adjacent pixel in the same category close to each other, while those in different categories are separated from each other. This not only preserves the local features of the initial data but also obtains a more compact distribution structure, while reducing Bayesian error, ultimately achieving the goal of improving classification accuracy.

2.2 Normalization Processing of Hyperspectral Remote Sensing Images

To achieve the desired effect, a basic preparation work needs to be carried out before classification, which is to perform standard normalization processing on the initial data. In classification models based on convolutional neural networks, using standardized methods to input data into different classification spaces can result in significant differences in classification results [10–12]. The spectral reflectance values of each individual pixel in the initial image may vary from several hundred to several thousand, and the numerical span is relatively large during the separation process, which increases the computational difficulty. However, using the standardized mean difference method can relatively improve. Firstly, each trajectory segment of the hyperspectral image is standardized. This operation can make the spectral curve fluctuation trajectory of each individual pixel more obvious and easier to determine, while increasing the variation of trajectory differences and relatively reducing the complexity [13–15]. Thus achieving an effect of improving classification training speed and accuracy. Assuming all pixels are taken as column vector x_i, the formula is as follows:

$$X_i = \frac{X_i - \mu}{\lambda} - 1 \tag{7}$$

In the equation, μ represents the average pixel value of the initial image, and λ represents the pixel standard deviation of the image at i curve band. After standardizing the initial data, it will lay a good foundation for subsequent image classification and greatly improve the classification effect [16, 17].

3 Image Classification Method Based on Three-Dimensional Convolutional Neural Network Model

3.1 Image Recognition

The CNN model is used to complete hyperspectral remote sensing image recognition, which includes an input layer, a hidden layer, and an output layer. The hidden layer serves as the neural structure layer in the model, including convolution, pooling, and a single-layer perceptron. This layer mainly realizes image recognition [5, 19].

Using processed hyperspectral remote sensing images as input samples for the CNN model, with a quantity of m, the sample set $\left\{\left(x^{(1)}, y^{(1)}\right), ..., \left(x^{(m)}, y^{(m)}\right)\right\}$ consists of n categories. Taking sample $x^{(i)}$ as a Reference, the corresponding category labels are represented by j. The basic cost function calculation formula for the network model is:

$$\begin{aligned} \varphi(W, b) = R(W, b) &= \left[\frac{1}{m}\sum_{i=1}^{m}\varphi\left(W, b; x^{(i)}, j\right)\right] \\ &= \left[\frac{1}{m}\sum_{i=1}^{m}\left(\frac{1}{2}\left\|h_{W,b}\left(x^{(i)}\right) - j\right\|^2\right)\right] \end{aligned} \tag{8}$$

In the formula, the weight value is represented by W, which is used to complete the connection of each layer in the model; The offset term is represented by b; $h_{W,b}\left(x^{(i)}\right)$ represents the output result and is completed at the last level of the model.

The training purpose of the model is to obtain the minimum value of $\varphi(W, b)$, using parameters W and b as Reference. The optimization objective function is completed through the gradient descent method, and the iterative equation is:

$$W_{ij}^{(l)} = W_{ij}^{(l)} - \alpha\frac{\partial}{\partial W_{ij}^{(l)}}\varphi(W, b) \tag{9}$$

$$b_{ij}^{(l)} = b_{ij}^{(l)} - \alpha\frac{\partial}{\partial b_{ij}^{(l)}}\varphi(W, b) \tag{10}$$

In the equation, the learning rate is represented by α. The BP algorithm is used to solve the partial derivative of formulas (9) and (10). The acquisition of $h_{W,b}\left(x^{(i)}\right)$ is completed through the forward propagation algorithm. The difference between this value and the actual value is represented by $\delta_i^{(nl)}$, and the solution of $\delta_i^{(nl)}$ and nl represents the model output layer; The residual error of each layer of the model is solved based on the residual error of 6, and the partial derivative of formula (9) and (10) is solved.

The residual solution formula for the last layer of the network is:

$$\delta_i^{(nl)} = \frac{\partial \varphi_1}{\partial Z_i^{(nl)}} = \frac{\partial}{\partial Z_i^{(nl)}} \frac{1}{2} \left\| h_{W,b}\left(x^{(i)}\right) - j \right\|^2 \tag{11}$$

In the formula, both $Z_i^{(l)}$ and $Z_i^{(nl)}$ represent weighted sums, and both belong to the i unit, with the former located in the l layer; The latter is located on the last layer of the model. Based on the above steps, complete hyperspectral remote sensing image recognition.

3.2 Model Optimization Image Classification

During the recognition process of the CNN model, the full connection process of the single-layer perceptron determines the recognition output of the model. Therefore, in order to improve the recognition effect and convergence efficiency of the model, double optimization is performed on it.

If k represents the number of convolutional layers and $l_{img} \times l_{img}$ represents the size, it belongs to the convolutional kernel; The matrix of $l_{img} \times l_{img}$ represents the input image size, while both n_{in} and n_{out} represent the number of images and correspond to the input and output respectively; Using iterative methods, the feature matrix S with the smallest objective function value is processed to obtain a matrix of Mat_1. The convolution kernel optimization is completed through convolution coefficients, and the convolution results are analyzed using the binary method; Establish a function expression and complete it based on the interpolation principle; μ represents the coefficient, which belongs to dynamic convolution, and its calculation formula is:

$$\mu = \frac{n_{in} n_{out} l_{img}^2}{2^k} + \theta_1 \tag{12}$$

In the formula, the correction Error term is represented by θ_1. The formula for solving the number of parameters is represented by formula (13), which corresponds to the convolutional kernel for both input and output data. The formula is:

$$f_{in} = n_{in} l_{\text{ker}}^2, f_{out} = n_{out} l_{\text{ker}}^2 \tag{13}$$

The optimized convolution kernel calculation formula (14) represents and is initialized:

$$M_{at_2} = 2\sqrt{\frac{\mu}{f_{in} + f_{out}}} \cdot M_{at_1} \tag{14}$$

In the equation, the convolutional kernel parameter matrix is represented by M_{at_2} and is optimized.

Let ρ represent the optimization coefficient, which is used to optimize the fully connected parameters. The formula is:

$$\rho = \frac{n_{cag}}{2}\left(\gamma - \varepsilon k^{\varepsilon - 1}\right) \tag{15}$$

In the equation, γ represents a factor that is related to the optimization coefficient. If θ_2 indicates the correction of Error term, then:

$$\gamma = \left[\lambda\left(\sqrt{n_f l_{img}^2 - n_{cag}}\right) - k^{\varepsilon}\right] + \theta_2 \tag{16}$$

$$\lambda = k + \sum_{i=0}^{\varepsilon-1} i \tag{17}$$

The optimized fully connected layer parameter formula is:

$$Mat_4 = 2\sqrt{\frac{\rho}{n_{cag} + n_f l_{img}^2}} \cdot Mat_3 \tag{18}$$

Based on the above optimization steps, the calculation formula for the optimization coefficient η of the model is:

$$\eta = \frac{\mu_0}{\mu_0 + \mu}\rho + \frac{\mu}{\mu_0 + \mu}\left(\gamma - k^{\varepsilon}\right) \tag{19}$$

Update and solve formula (18):

$$Mat_4 = 2\sqrt{\frac{\eta}{n_{cag} + n_f l_{img}^2}} \cdot Mat_3 \tag{20}$$

On this basis, an optimized 3D convolutional neural network model is used to introduce a support vector machine to construct an image classifier:

$$g\left(x_j\right) = aign\left(\sum_{i=1}^{Num} a_i y_i \sum_{m=1}^{M} \beta_m K_m\left(x_i, x_j\right) + b\right) \tag{21}$$

In the formula, $g\left(x_j\right)$ represents the j expected image label parameter in $MKL - SVM$, and Num represents the number of collected parameters. Input Num has different expressions in different situations.

a_i、y_i represents the optimized parameters and the number of samples collected during training, while $K_m\left(x_i, x_j\right)$ represents the kernel function in this formula. In the test set and training set, it is $N \times (n - m)$、$N \times m$ respectively. To obtain the results of image classification, the features of the fused test set images are input into an $MKL{-}SVM$ high-resolution image classifier to complete image classification.

4 Experimental Methods

4.1 Experimental Parameter Settings

Using hyperspectral remote sensing images as experimental samples. 15% of the labeled samples will be randomly selected as test samples, and the remaining 85% will be used as training samples. Firstly, the preprocessed initial data is transferred into the input

layer as raw data. Then, take 1×100 as the column vector of each pixel for preliminary data input, go through the second convolution layer, set the image feature to 30, and the convolution kernel to size 1×4, and use the actual effect activation function RELU for activation. At this time, take the column vector of 30 1×100 for each pixel as the initial data input into the convolution layer. Take the data output from the convolution layer as the input data of the next pooling layer. The window size of the pooling layer is 1×2. Each pixel passing through this layer will output 25 1×50 column vector. After this operation, good classification results can be achieved. Using Reference [3] and Reference [4] as experimental comparison methods to verify the effectiveness of the algorithm.

4.2 Experimental Results

Table 1. Comparison of Training Times and Coefficients of Three Methods

Index	Reference [3] Method	Reference [4] Method	Proposed method
training time/s	6.11	7.89	6.62
Test time/s	15.23	16.99	12.36
overall time/s	25.67	26.82	20.44
kappa coefficient	76.52	82.34	91.44

Table 1 shows the comparison of the time and *Kappa* coefficient required for the three classification algorithms in the classification experiment process. From the table, it can be seen that although the method in Reference [2] has slightly lower training time than the algorithm in this paper, the classification accuracy is the lowest among these methods. This is because in order to achieve ideal classification results during the training process, a large amount of training time is required. However, compared to other algorithms, this method has a relatively fast testing and training process, and has achieved the optimal classification experimental results. It can be said that there is a good balance between accuracy and time selection. The coefficient in this article *Kappa* is also the best among all methods, indicating that the optimal classification can be achieved in any aspect.

Classification of remote sensing images with different resolutions is an important research field in the application of remote sensing technology. Classification accuracy is the main indicator for measuring the quality of classification results. This project aims to study high-resolution and low-resolution data, and compare and analyze the image classification results under two different resolutions. The comparison results of classification accuracy obtained are shown in Figs. 1 and 2.

From Fig. 1, it can be seen that as the number of iterations increases, the three classification methods show an upward trend. However, compared to the other two methods, the overall classification accuracy of the classification method designed in this article is higher. When the number of experimental iterations reached 65000, the method

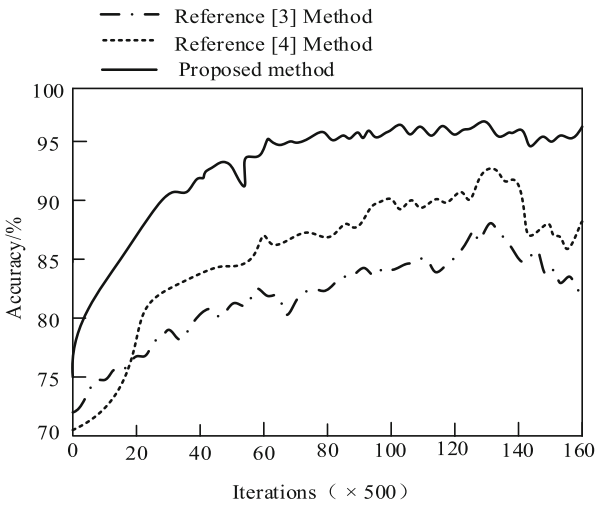

Fig. 1. Comparison results of classification accuracy for high-resolution remote sensing images

in this paper achieved the highest classification accuracy of 96.62% in the test dataset, and the classification effect was good. The reason for obtaining the experimental results is that the test set images selected in the experiment contain a large number of background regions, and the remote sensing image features are not obvious. The classification model construction module used in the literature method is relatively simple, so the accuracy of discriminative region positioning is lower, so the classification accuracy is lower. This article adopts a three-dimensional convolutional neural network model as the training model for hyperspectral remote sensing images. By optimizing the convolutional kernel and fully connected parameters, accurate classification of hyperspectral remote sensing images can be achieved.

Due to the blurry edge and target features of low resolution remote sensing images, as well as the partial loss of pixels in remote sensing images, it increases the difficulty of remote sensing image classification. From Fig. 2, it can be seen that as the number of samples increases, the overall classification accuracy of the classification method designed in this article is higher compared to the other two methods. When the sample size reached 90, the method in this paper achieved the highest classification accuracy of 95.5% in the test dataset, and the classification effect was good. The reason for obtaining the experimental results is that the method used in this paper preprocesses and normalizes the data of hyperspectral remote sensing images, eliminating the influence of noise and interference in hyperspectral image data, and thus obtaining more accurate and reliable information. This can prove that the proposed method can achieve better classification results for remote sensing images in both high-resolution and low-resolution scenarios.

Figure 3 shows the comparison of classification and recognition time between our method and traditional methods, with six datasets set in the experiment. From the figure, it can be seen that the classification and recognition time of our method is relatively short, while the Reference method has poor data convergence ability due to not removing redundant information, and cannot quickly achieve classification and recognition of

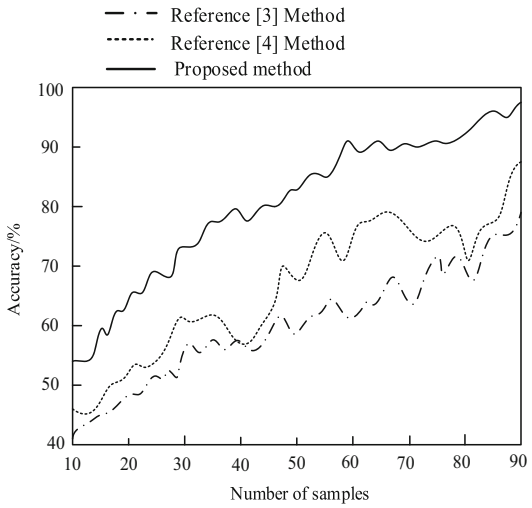

Fig. 2. Comparison Results of Classification Accuracy of Low Resolution Remote Sensing Images

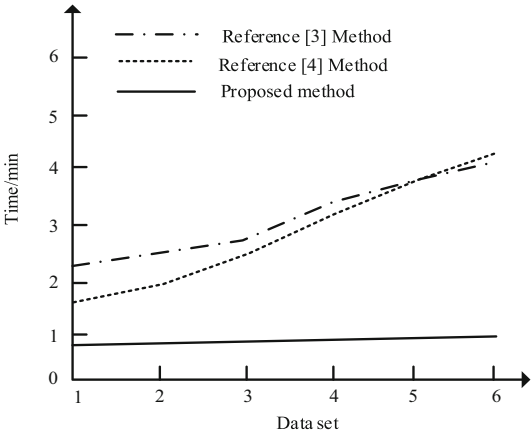

Fig. 3. Comparison of classification and recognition time for hyperspectral remote sensing images

hyperspectral remote sensing images. This also proves that the method proposed in this paper has good advantages and can adapt to precise image classification and recognition under various conditions. Three dimensional Convolutional neural network has good parallel computing ability, which can learn spatial and spectral information at the same time, effectively reducing processing time, thus accelerating the speed of classification.

On this basis, the recall rates of three methods for hyperspectral remote sensing image classification were tested, and the experimental comparison results are shown in Fig. 4.

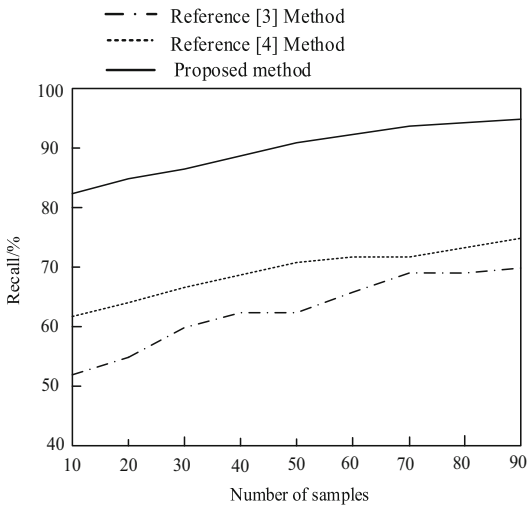

Fig. 4. Recall rate of hyperspectral remote sensing image classification

Analyzing Fig. 4, it can be seen that the recall rate of hyperspectral remote sensing image classification for the method in reference [3] is 68%, the recall rate of hyperspectral remote sensing image classification for the method in reference [4] is 73%, and the recommended English online teaching video resources for the method in this article take 94% of the time. Three dimensional Convolutional neural network model can automatically extract representative features from hyperspectral images by multiple convolution and pooling operations. These features typically have a higher level of abstraction and can better distinguish different categories. By feature extraction and dimensionality reduction, the dimensionality of input data can be reduced, thereby improving the classification recall rate of hyperspectral remote sensing images.

5 Conclusion

This article establishes a new classification method for hyperspectral remote sensing images. On this basis, further optimize the structure and parameters of the neural network, improve its learning and classification efficiency and stability, and enhance its ability to extract complex pixel distributions and semantics from remote sensing images. Experiments have shown that this algorithm can accurately classify hyperspectral images, better capture image feature information, improve classification accuracy, and have a relatively fast classification speed, effectively solving the classification problem of hyperspectral images.

In future research, it is planned to further improve and improve the classification algorithm of hyperspectral remote sensing images, improve their classification accuracy and robustness through in-depth research on convolutional neural network models based on previous work. At the same time, this project will also use convolutional neural networks to interpret hyperspectral remote sensing data and explore deep information within it.

Aknowledgement. 2020 Anhui Province University Excellent and Top notch Talent Cultivation (gxgnfx2020112)

References

1. Sadovnychiy, S.: Gabor features extraction and land-cover classification of urban hyperspectral images for remote sensing applications. Remote Sensing **13**(15), 2914 (2021)
2. Jijon-Palma, M.E., Kern, J., Amisse, C., et al.: Improving stacked-autoencoders with 1D convolutional-nets for hyperspectral image land-cover classification. J. Appl. Remote. Sens. **2**, 15 (2021)
3. Cao, H., Han, X., Li, J., et al.: Hyperspectral remote sensing image classification based on EMP and SVM with composite kernel. Geospatial Inf. **19**(11), 14–18 (2021)
4. Liu, J., Ban, W., Chen, Y., et al.: Multi-dimensional CNN fused algorithm for hyperspectral remote sensing image classification. Chinese Journal of Lasers **48**(16), 153–163 (2021)
5. Peng, Y., Wang, X., Zhang, J., et al.: Pre-training of gated convolution neural network for remote sensing image super-resolution. IET Image Processing **15**(5), 11791188 (2021)
6. Anand, R., Veni, S., Aravinth, J.: Robust classification technique for hyperspectral images based on 3D-discrete wavelet transform. Remote Sensing **13**(7), 1255 (2021)
7. Gong, H., Li, Q., Li, C., et al.: Multiscale information fusion for hyperspectral image classification based on hybrid 2D–3D CNN. Remote Sensing **13**(12), 2268 (2021)
8. Xi, J., Ersoy, O.K., Fang, J., et al.: Wide sliding window and subsampling network for hyperspectral image classification. Remote Sensing **13**(7), 1290 (2021)
9. Liu, X., Chen, S., Song, L., et al.: Self-attention negative feedback network for real-time image super-resolution. J. King Saud University - Computer and Information Sci. **34**(8B), 6179–6186 (2022)
10. Liu, M., Tan, J., Tian, Y.: Decoding auditory attentional states by a 3D convolutional neural network model. Int. J. Psychophysiol. **168**, 134–135 (2021)
11. Alameddine, J., Chehdi, K., Cariou, C.: Hierarchical unsupervised partitioning of large size data and its application to hyperspectral images. Remote Sensing **13**(23), 4874 (2021)
12. Hawkesford, M.J.: A neural network method for classification of sunlit and shaded components of wheat canopies in the field using high-resolution hyperspectral imagery. Remote Sensing **13**(5), 898 (2021)
13. Deshmukh, R.R., Ghule, A.: Wavelength selection and classification of hyperspectral non-imagery data to discriminate healthy and unhealthy vegetable leaves. Curr. Sci. **120**(5), 932–936 (2021)
14. Macfarlane, F., Murray, P., Marshall, S., et al.: Investigating the effects of a combined spatial and spectral dimensionality reduction approach for aerial hyperspectral target detection applications. Remote Sensing **13**(9), 1647 (2021)
15. Rossello, J.C., Graham, P., Prakash, A., et al.: Airborne hyperspectral data acquisition and processing in the arctic: a pilot study using the hyspex imaging spectrometer for wetland mapping. Remote Sensing **13**(6), 1178 (2021)
16. Zhang, Y., Du, J., Pi, W., et al.: Deep learning classification of grassland desertification in china via low-altitude UAV hyperspectral remote sensing. Spectroscopy **37**(1), 28+3035 (2022)
17. Ping, W., Fu, J., Qiao, W., et al.: Decision support system for hyperspectral remote-sensing data of Yellow River Estuary, China. Scientific Programming **2121**(9), 1–17 (2021)
18. Li, K., Xu, J., Zhao, T., et al.: A fuzzy spectral clustering algorithm for hyperspectral image classification. IET Image Processing **15**(12), 2810–2817 (2021)
19. Li, J., Shen, H., Li, H., et al.: Radiometric quality improvement of hyperspectral remote sensing images: a technical tutorial on variational framework. J. Applied Remote Sensing **15**(3), 1–33 (2021)