International Journal of Research in Engineering and Science (IJRES)
ISSN (Online): 2320-9364, ISSN (Print): 2320-9356
www.ijres.org Volume 12 Issue 2 ǁ February 2024 ǁ PP. 42-51

Multi-head Self Attention Machanism and Reflection Padding Network for Hyperspectral Image Band Selection

Fuhao Yang[1], Zhongmin Jiang[1], Wenju Wang[1]

[1] *Department of Communication and Art Design, University of Shanghai for Science and Technology, Shanghai China*

Abstract

Hyperspectral images can record dozens or even hundreds of narrow bands. However, when the number of bands is too large, the phenomenon that the classification accuracy first increases and then decreases with the increase of the number of bands involved in the operation occurs. Band selection is an important method for dimensionality reduction of hyperspectral images, however, many existing band selection methods estimate the significance of each band individually, which cannot fully take into account the nonlinear and global interactions between spectral bands. In this paper, a hyperspectral image Band Selection network framework based on Multi head Self-Attention mechanism and Reflection Padding (MSARPBS) is proposed, which consists of a cascade of Band Multi head Self-Attention module (BMSA) and Reflection Padding module (RP) as well as Feature Extraction and Image Reconstruction module (FER).The BMSA can acquire the nonlinear and global interdependencies between bands.The RP is used for better acquisition of image edge features.FER extracts the band information based on the learned information and recovers it into the original hyperspectral cube.Our proposed MSARPBS method has been applied to Indian Pines dataset. Experimental results show that the mean spectral dispersion (MSD) of a subset of bands selected by our proposed framework can be stabilized at 28, which is better than many existing band selection methods for hyperspectral images.

Keywords: *Band selection, Hyperspectral image, Attention mechanism, Deep learning*

Date of Submission: 26-01-2024 Date of acceptance: 08-02-2024

I. INTRODUCTION

Methods for hyperspectral image band selection are broadly categorized into two types: traditional machine learning methods and methods of deep learning. Traditional machine learning methods for spectral band selection are mainly sparse representation, filter, clustering and other methods. These methods not only have low classification accuracy, but are also easily affected by noise.

Band selection based on conventional machine learning: Band selection based on sparsity can be performed using a self-representation learning framework with a sparse one-dimensional operational self-encoder[1] , the operational layer in the self-encoder, the improved neuron model can efficiently learn the nonlinear kernel transformation function, so it can provide a better performance of band selection. Band selection can also be described as a sparse self-representation problem[2] , where the entire frequency band can be represented by a set of bands with complementary information. This improves the classification accuracy and the computational vision is shorter. The filter-based approaches are mainly Gaussian filters[3] and noise filters[4] , using Gaussian filters to preprocess hyperspectral image data and then using a binary correction balanced optimizer for band selection can achieve a high classification accuracy. The use of noise filters to minimize the effect of noise on band selection and the introduction of clustering to reduce spatial redundancy and extract different patterns from the data can lead to an improvement in the computational performance of the particle swarm algorithm for each iteration. There are many clustering-based band selection methods, listed as the method of hypergraph regularization weighted low-rank subspace clustering[5] , which introduces weighted low-rank subspace clustering in hyperspectral image band selection, improves the contribution of the representation of the most important features, and introduces soft hypergraph Laplacian regularization into the framework of weighted low-rank representation, which solves the limitations of ordinary hypergraph (hypergraph); A joint deep learning and clustering approach[6] , specifically, this approach embeds the popular K-mean clustering loss into the recently developed transformer deep learning framework and solves the subsequent formulas as a way to improve the classification accuracy through the alternating direction method of multipliers; A clustering band selection method based on deep subspace clustering[7] , which uses the subspace clustering task as a self-expression layer and combines it into a convolutional selfencoder, enabling end-to-end training. The method selects a subset of bands with significant classification accuracy. There are also linear prediction methods that use matrix Schmidt orthogonalization to improve the efficiency of linear prediction algorithms[8] as well as methods that select a representative set of bands for each dataset and make a selection of the bands by quantifying the difference between the maximum and minimum of the pixels in each band[9].

These traditional methods are relatively simple and easy to use though. However, the feature extraction of traditional machine learning mainly relies on manual labor, and the learning ability is weak. In addition, traditional machine learning methods generally have low classification accuracy.

Deep learning based band selection: The most basic deep learning network framework is the convolutional neural network, which can significantly improve the performance of band selection. However, hyperspectral images contain rich spectral information, and a separate convolutional neural network cannot fully extract all the feature information of hyperspectral images, so a variety of improved convolutional neural networks have emerged.Y-Net[10] is one of them, Y-Net is a 3D-2D hybrid convolutional neural network model, which can take into account both spatial and spectral features of hyperspectral images, making the selected bands more representative and robust. The band selection framework based on the combination of convolutional neural network and genetic algorithm [11] is based on the use of embedded 3D convolutional layer as the fitness function in the genetic algorithm and the parent checkbox is designed to make the genetic operation more efficient, which effectively improves the accuracy. Measuring band representativeness by a convolutional self-encoder network[12] captures the inherent nonlinear relationships between bands well and ensures the accuracy of representativeness measurements by re-scaling the occluded data as input to the trained convolutional self-encoder. Training a compact convolutional neural network[13] to evaluate the performance of band selection eliminates a large number of redundant bands and reduces the search space. However, these methods are less adaptive and more computationally intensive because they do not use the attention mechanism.And BS-Net[14] consists of a Band Attention Module (BAM) and a Reconstruction Network (RecNet). The former is used to model the nonlinear interdependencies between spectral bands, and the latter is used to recover the original HSI cube from the learned band information, resulting in a flexible architecture, which allows for accurate selection of a subset of information bands. But the computational complexity is high. There are also deep learning frameworks based on the attention mechanism for wave selection[15] altered the activation function of the output of the attention mechanism, which allows the network to alternate between wave full-time and wave selection, which effectively models the relationship between the waves and removes irrelevant waves. In addition to convolutional networks, reinforcement learning is often used for band selection. A subset of spectral bands is selected for selection using deep reinforcement learning[16] , using which training the model learns a band selection strategy that enables the model to select bands sequentially by making full use of the hyperspectral image and the previously selected bands. However, this approach simply removes redundant bands and lacks a significance analysis of the selected bands. In contrast, the reinforcement learning-based feature selection framework framework[17] proposes two feature evaluation structures by introducing a pre-trained evaluation network and cross-validation techniques, respectively, through which unique and valuable spectral features can be effectively selected. Modifying the incentive mechanism is also a common way of selecting waves. For example, a new reward mechanism strategy is proposed and Deep Reinforcement Learning is utilized to invoke Double Deep Q-Network (DDQN) [18] in the wave selection process. This improves the stability of the network, avoids local optimization, and the selected bands are well distributed along the spectral dynamic range. However, this method only estimates the significance of each band individually and does not fully consider the nonlinear and global interactions between the spectral bands. The similarity-based dual-depth Q-network-based band selection method[19] , on the other hand, uses three similarity metric schemes, namely cosine distance, Euclidean distance, and linear prediction, as a reward scheme for band selection, which can achieve a high classification accuracy. In addition, there are deep learning architectures consisting of a constrained measurement learning network for band selection and a classification network[20] trained in a data-driven manner on the proposed joint deep learning architecture to optimize the classification loss when selecting bands. And band selection with an end-to-end deep learning pipeline[21] that incorporates a constrained measurement learning structure to select bands in a data-driven manner, an approach that selects bands that directly optimize the cost associated with the final classification task.

Although existing deep learning-based BS methods have achieved good performance, the accuracy of the selected bands has to be further improved. In order to further improve the band selection accuracy and extract comprehensive and rich features, we have done the following two aspects: in this paper, we propose a framework based on Multi head Self-Attention Mechanism and Reflection Padding (MSARP). It serves as a framework for hyperspectral image band selection, including the Band Multi head Self-Attention module (BMSA), Reflection Padding module (RP) and Feature Extraction and Image Reconstruction module (FER). The experimental results demonstrate the superiority of the changed framework in band selection; MSARPBS constructs a Band Multi head Self-Attention module (BMSA). This module is capable of extracting rich channel features as well as spatial features of hyperspectral images. In addition, it preserves the computational complexity at a low level while acquiring global attention.

II. OUR WORK

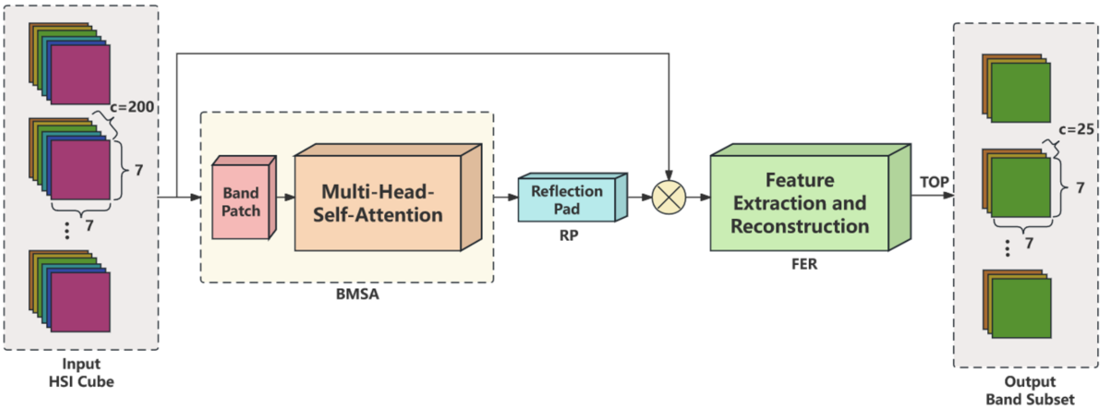

Figure 1: Multihead Self-Attention Reflection Padding Band Selection Network

As shown in Figure 1, the Multihead Self-Attention Reflection Filled Band Selection Network (MSARPBS) framework proposed in this paper is mainly divided into three parts: the Band Multihead Self-Attention (BMSA) module, the Reflection Filling (RP) module and the Feature Extraction and Image Reconstruction (FER) module. We use HSI cube as a training sample. For convenience, we denote the n training samples as $X \in \mathbb{R}^{n\times7\times7\times c}$ (n=10249,c=200),When X goes through the BMSA module to obtain the global band weights and through the RP module to obtain the boundary information, it becomes $Y \in \mathbb{R}^{n\times7\times7\times c}$. And before going into the FER module for feature extraction as well as image reconstruction, it is necessary to multiply X and Y to get the weighted spectral input $O \in \mathbb{R}^{n\times7\times7\times c}$,as shown in equation (1). The reconstructed hyperspectral image is the best subset of bands by directly selecting the TOP=25 band. The specific structure of the BMSA module is described in section 2.1, the specific structure of the RP module is described in section 2.2.

$$O = X \otimes Y \qquad (1)$$

2.1 Band Multi head Self Attention

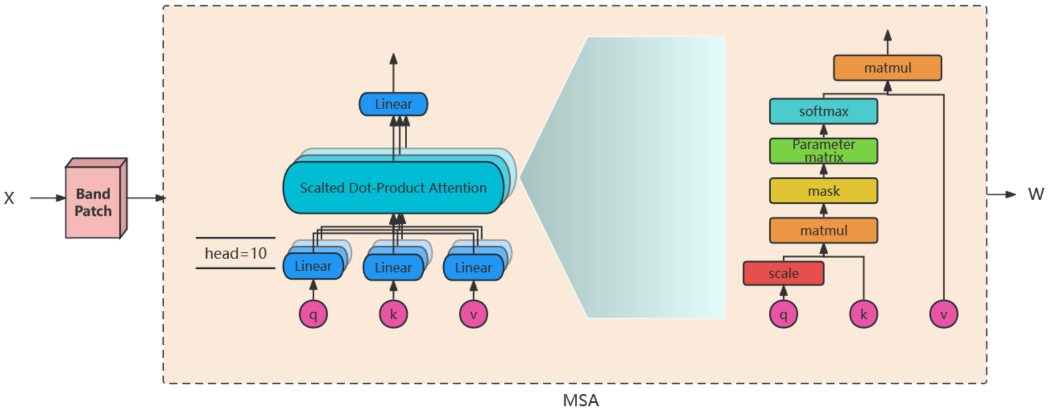

Figure 2: Band Multi-head Self-Attention Module

Our proposed Band Multi head Self-Attention Module (BMSA) is the part of MSARPBS used to obtain band weights. X is the training sample HIS cube and, W is the band weights. As shown in Figure 2, the BMSA mainly contains a band patch part as well as an MSA part.

2.1.1 Band patch division

Since the multi-head self-attention mechanism of 2.1.2 performs weight acquisition on the entire feature map, for this purpose attention operations need to be done on the pixels of the image, then the length of each sequence will be the number of image pixels. As a result, this will result in the length of the sequence being too long thus making the whole network training difficult. Therefore we do patch partitioning of the feature map before performing the attention operation. In particular, since the feature map for this task is a hyperspectral feature map containing 200 channels, Band Patch directly divides each channel into a patch instead of the traditional division of an image into multiple patches. In fact, this work is mainly performed by a convolutional layer, as detailed in Equation. (2)

$$L = Conv_{7\times7}\left(X\right), L \in ¡^{\,n\times1\times c} \tag{2}$$

2.1.2 Multi head Self-Attention acquiring channel weights

The Multi head Self-Attention section introduces the idea of Transformer [22] to obtain the channel weights of the patch sequence L in Equation (2) through the global sensing field, as a way to achieve the re-assignment of different attentions to X , which in turn optimizes the extraction of the channel correlations .The process can be divided into the following 6 steps.

① **Channel dimensionality downscaling**

The multi-head attention mechanism reduces the dimensionality of each vector of each $head$ and enhances the fitting performance. Its workflow, that is, with the total number of parameters kept constant. Change the channel dimension of the sequence L from C (C=200) to C' (C'=C/head) and add a new dimension whose size is head (head=10). This operation is accomplished by the reshape function and the sequence L is thus changed to L' . The process can be characterized as Equation (3):

$$L' = reshape\left(L\right), L \in ¡^{\,n\times1\times c}, L' \in ¡^{\,n\times1\times\frac{c}{\text{head}}\times head} \tag{3}$$

where HEAD is the number of heads.

② **head dimension refinement**

L' can be discretized into $L_1, L_2, ...\mathrm{L}_i ..., L_{head}$ along the HEAD dimension, and each L_i can be discretized into Q 、K and V matrices. Since the operations are the same for all L_i 's Q 、K and V divisions, as well as before going on to the linear merger, one of the L_i 's is used as an example to illustrate the detailed operation steps below.

The moving window sequence L_i requires the Q_i 、K_i and V_i matrices generated by matrix multiplication in order to do the attention operation. This is due to the fact that the use of three trainable parameter matrices enhances the fit of the model. As shown in Figure3, L_i is multiplied with $W_i^{\,q}$ 、$W_i^{\,k}$ and $W_i^{\,v}$ to obtain the matrices Q_i 、K_i and V_i respectively. where $W_i^{\,q}$ 、$W_i^{\,k}$ and $W_i^{\,v}$ are three randomly initialized parameter matrices whose values are updated throughout the backpropagation process. The process is described in Equations (4), (5), and (6).

$$Q_i = L_i W_i^q \tag{4}$$

$$K_i = L_i W_i^k \tag{5}$$

$$V_i = L_i W_i^v \tag{6}$$

③ **Obtaining the weighting matrix**

The weight matrix Z is obtained by multiplying Q and K^T (transpose of K) after scale operation. When the vectors in Q and K^T are long, the value obtained by direct multiplication of Q and K^T will be larger (not distinguishing between positive and negative, the farther away from 0 the larger). After this value goes to $\mathrm{softmax}$, the larger value will be closer to 1, and the smaller value will be closer to 0. That is to say, the bifurcation is more serious, so that the gradient is calculated to be smaller, and the accuracy of the trained model is difficult to be improved. Therefore the Q matrix needs to be divided by one $\sqrt{head\,\dim}$ □ to mitigate the polarization before it can be multiplied with the K^T matrix to obtain the weight matrix Z , see

Equation (7). Where $head\,\mathrm{dim}$ denotes the number of dimensions of each head in the multi-head attention.

$$Z = scale(\mathrm{Q})\mathrm{g}K^{T} = \frac{Q\mathrm{g}K^{T}}{\sqrt{head\,\mathrm{dim}}} \tag{7}$$

④ **Softmax assigns weights**

The weight matrix Z through $\mathrm{softmax}$ then converts the output values of the multichannel into a probability distribution in the range [0, 1] and for 1. The process of $\mathrm{softmax}$ is shown in Equation (8).

$$atten_{i} = \mathrm{softmax}(\mathrm{Z}_{i}) = \frac{e^{Z_i}}{\sum_{j=1}^{C} e^{Z_j}} \tag{8}$$

Where $atten_i$ is the weight distribution, $Z_i\,'$ is the output value of the ith node, and C is the number of channels, i.e., the number of categories for classification

The reason why $\mathrm{softmax}$ is chosen to make a weight distribution on the weight matrix $\mathrm{Z'}$ is that e^{x} in $\mathrm{softmax}$ is the simplest, monotone, smooth primitive function that can map any real number to a nonnegative real number.

⑤ **Weighted sum**

The weight distribution $atten_i$ obtained from Equation (8) is matrix multiplied with the matrix V_i to obtain the weighted matrix $V_{atten-i}$ V_i , which is equivalent to adding attention weights to the V matrix. See Equation (9) for the weighted summation process.

$$V_{atten-i} = atten\mathrm{g}V_{i} \tag{9}$$

⑥ **Linear splicing**

L becomes $V_{atten-i}, i \in [1, head]$ after steps ② to ⑤. In order to obtain the output L' corresponding to the L dimension, $V_{atten-1}, V_{atten-2}, \ldots V_{atten-i} \ldots, V_{atten-head}$ should be made reshape, stitching them together along the channel dimension C. The Equation is as follows:

$$L' = reshape\left(\sum_{i=1}^{i=head} V_{atten-i}\right), L' \in ¡^{\,n\times1\times c}, V_{atten} \in ¡^{\,n\times1\times\frac{C}{\mathrm{head}}} \tag{10}$$

In addition, to facilitate the subsequent part of the extraction of channel correlations, the weighting matrix L' needs to be recovered into the shape of the input feature map X. The process can be characterized as Equation (11).

$$W = \exp\mathrm{and}\left(L'\right), W \in ¡^{\,n\times7\times7\times c} \tag{11}$$

2.2 Reflective Padding

Conventional convolution kernels perform convolution operations around the center pixel, and the center of the convolution kernel does not reach the pixels at the edges. Therefore edge pixels are generally not efficiently convolved. Reflective Padding is pixel pad at the edges of the feature map, which is mirrored [27]. This enables the convolution kernel to process to the edge pixels, effectively acquiring the features of the edge pixels and obtaining the output Y . This procedure is shown in (12) to (14).

$$Y' = \mathrm{ReflectionPad\,2\,d}\left(W\right) \tag{12}$$

$$H_{Y'} = H_{W} + padding\_top + padding\_bottom \tag{13}$$

$$W_{Y'} = W_{W} + padding\_\mathrm{left} + padding\_right \tag{14}$$

Where H_W and W_W are the height and width of W ; $H_{Y'}$ and $W_{Y'}$ are the height and width of Y' ; $padding\_top = padding\_bottom = padding\_left = padding\_right = padding$; $padding$ is the number of pixels for padding and its calculation Equation is shown in (15).

When the feature map will increase the feature map size after reflection padding, in order to keep the feature map size consistent, a convolution is also needed to restore to the original size. The specific Equation is shown in (16).

$$Y = Conv_{3\times3}\left(Y'\right), Y' \in \mathbb{R}^{n\times c\times H_{Y'}\times W_{Y'}}, \mathrm{Y} \in \mathbb{R}^{n\times c\times H_Y\times W_Y} \tag{15}$$

$$padding \ = \ int\Big(dilation \ * \ \big(kernel\_size \ - \ 1\big) \ / \ 2\Big) \tag{16}$$

Where $kernel\_size = 3$; $dilation = 1$; int is a shaping function that will force the number in () to be converted to an integer; H_Y and W_Y are the height and width of Y .

III. EXPERIMENTAL RESULTS AND ANALYSIS

3.1 Environment Configuration

3.1.1 Training environment

The hardware environment used for the experiment is GPU is NVIDA GeForce RTX2080 SUPER, CPU is Intel Core i9-10900K, and 32GB of RAM. The operating system is Windows 10. The software environment is pytorch1.10.0, python3.8. Various parameters are set as follows: number of bands in the subset of bands k = 25, number of multiple heads as head = 10, input and output channels = 200, batchsize = 8, initial learning rate = 2e-3. In addition, the other four band-selective network models DARecNet-BS[23]、BSNet-Conv[14]、PCA[24]、SpaBS[25] , which were used as comparison experiments, were also used in the same experimental environment as this one.

3.1.2 Dataset Settings

The dataset chosen for this paper is the Indian Pines dataset. This dataset is the earliest test data for hyperspectral image classification. A piece of Indian Pines in Indiana, USA, was imaged by the Airborne Visual Infrared Imaging Spectrometer (AVIRIS) with 1992, and then intercepted at a size of 145×145 to be labeled for hyperspectral image classification test purposes.

The AVIRIS imaging spectrometer has an imaging wavelength range of 0.4-2.5 μm and is continuously imaging features in a continuous 220 band. However, since the 104th-108th, 150th-163rd and 220th bands cannot be reflected by water. Therefore, this paper uses the 200 bands left after removing these 20 bands for training.

3.2 Evaluation indicators

The Structural Similarity Index Measure SSIM[26] can reflect the attributes of the structure of objects in an image from the perspective of image composition, and the use of SSIM as an objective function for training the network can better preserve the high-frequency details of the image. Specifically, SSIM uses mean, standard deviation, and covariance as measures of brightness, contrast, and structural similarity, respectively, as shown in equation (17).

$$SSIM\left(x_i, y_i\right) = \frac{\left(2\mu x_i \mu y_i + c_1\right)\left(2\sigma x_i y_i + c_2\right)}{\left(\mu^2 x_i + \mu^2 y_i + c_1\right)\left(\sigma^2 x_i + \sigma^2 y_i + c_2\right)} \tag{17}$$

Where x_i and y_i denote the ith pixel value of the trained hyperspectral image and the original hyperspectral image, respectively. In addition, μx_i is the mean of x_i ; μy_i is the mean of y_i ; $\sigma^2 x_i$ is the variance of x_i ; $\sigma^2 y_i$ is the variance of y_i ; $\sigma x_i y_i$ is the covariance of x_i and y_i . And $c_1 = \left(k_1 L\right)^2$, $c_2 = \left(k_2 L\right)^2$ is a constant, used to maintain stability. L is the dynamic range of the pixel value. $k_1 = 0.01$, $k_2 = 0.03$.

The mean absolute error L1 Loss[14] is the average of the distance between the model's predicted value X' and the true value X . When using L1 Loss as the objective function for training the network, there will be a stable gradient no matter for what kind of input values, and there will be no problem of gradient explosion, making the network robust. Specifically, for a dataset with sample number n , the Equation for L1 Loss is shown in (20).

$$L1\_loss = \frac{1}{n}\sum_{i}^{n}\left|X_i - X'_i\right| \tag{18}$$

The mean spectral dispersion MSD[14] is an average measurement of a subset of selected bands.MSD can evaluate the inter-segment redundancy of the selected bands, and the larger the value of MSD, the less inter-segment redundancy of the selected bands. Its Equation is shown in (19), (20).

$$D_{SKL}\left(B_i \mathrm{P} B_j\right) = D_{KL}\left(B_i \mathrm{P} B_j\right) + D_{KL}\left(B_j \mathrm{P} B_i\right) \tag{19}$$

$$MSD = \frac{2}{k\left(k-1\right)}\sum_{i=1}^{k}\sum_{j=1}^{k} D_{SKL}\left(B_i \mathrm{P} B_j\right) \tag{20}$$

where B_i denotes the i th band in the selected subset of bands. D_{KL} is the Kullback-Leibler scatter, which can be computed from the grayscale histogram information. D_{SKL} is the symmetric Kullback-Leibler scattering for measuring the dissimilarity between B_i and B_j. k is the size of the selected subset of bands.

3.3 Experimental analysis

3.3.1 Results Performance Demonstration

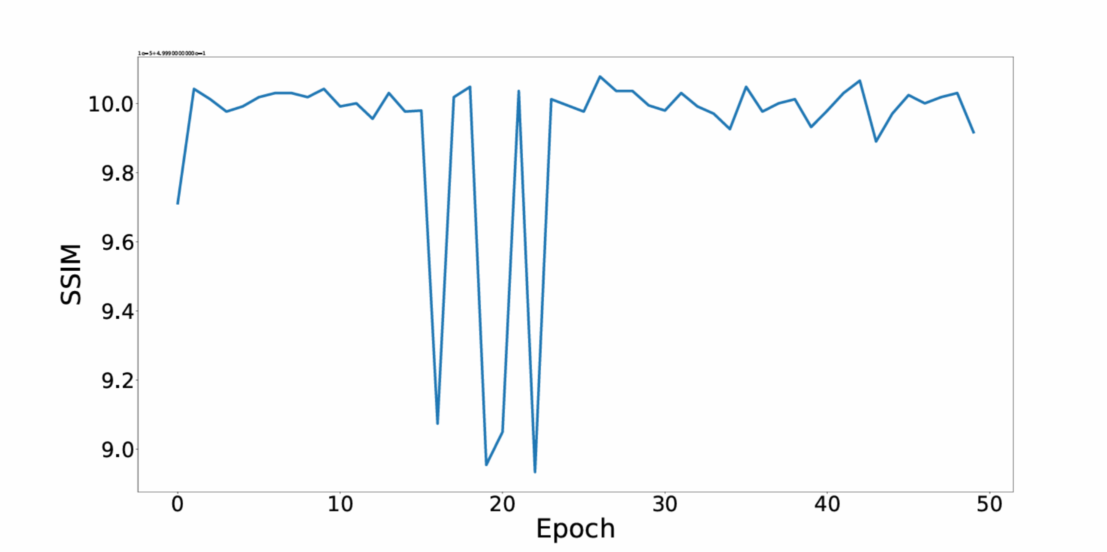

Figure 3: SSIM values obtained from 50 training sessions of our network MSARPBS

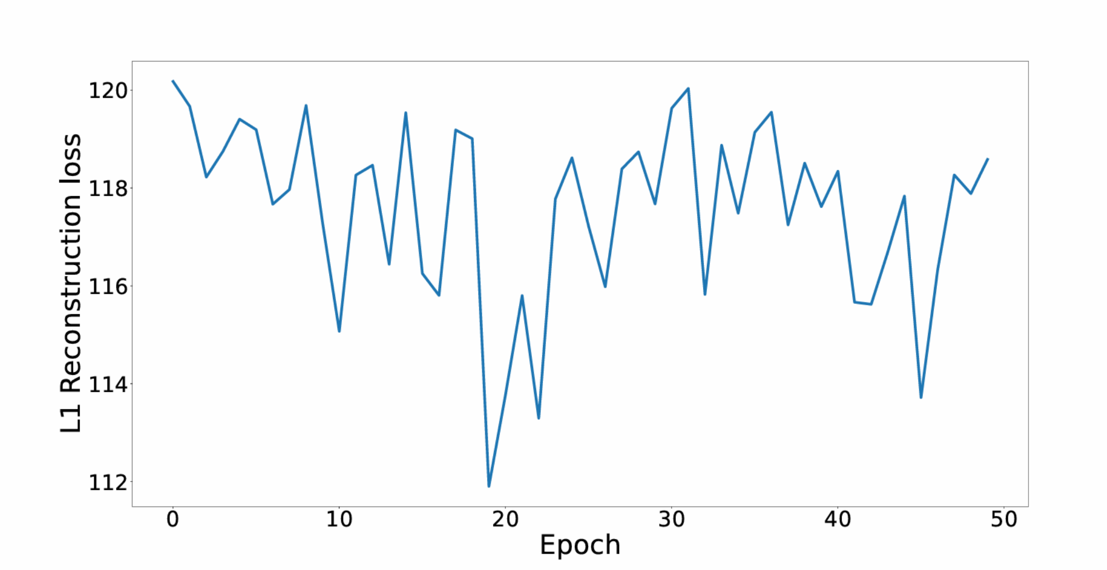

Figure4: L1 Loss value obtained from 50 training sessions of our work

The SSIM values and L1 Loss values obtained from 50 iterations of sub-training on our network using the Indian Pines dataset are shown in Figures 3 and 4 respectively. From these metrics, it can be seen that our network is able to obtain more stable results in the later stages of training, and thus the final band selection results in a subset of the bands selected for the last iteration of training. In addition, it can also be seen that the overall values are better. Proof of the effectiveness of MSARPBS

3.3.2 Comparative analysis with similar networks

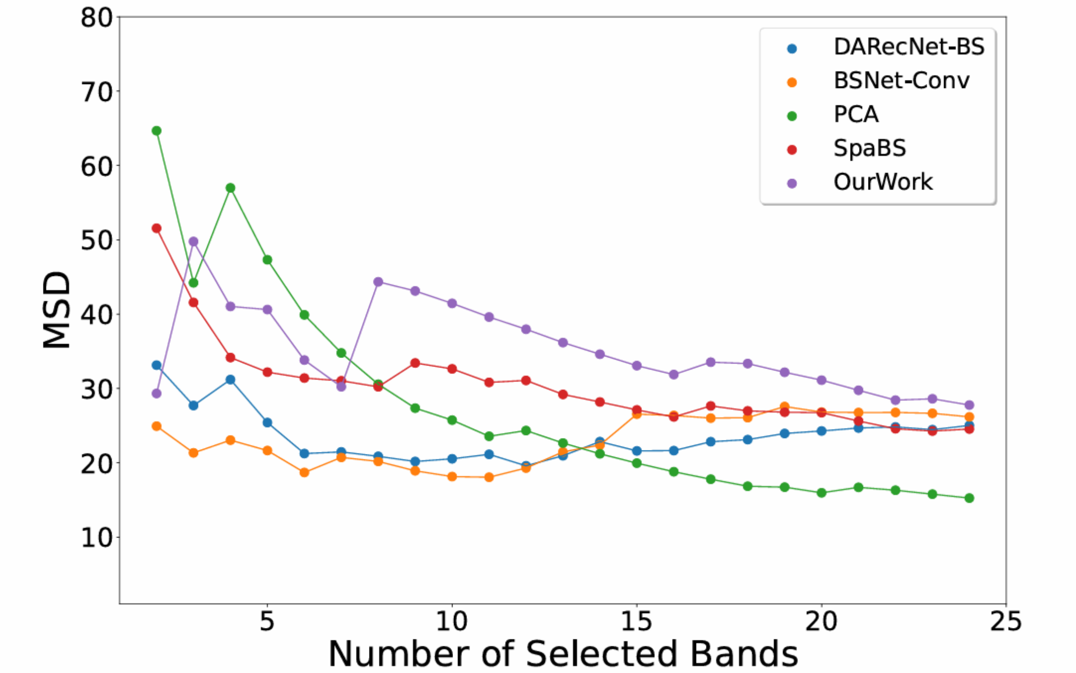

Figure 5: Comparison of MSD values of selected subset of bands for each network

As described in 3.3.1 we take out the subset of bands obtained from the last training of MSARPBS as the final band selection result and calculate the MSD value of that subset of so bands. Similarly, we also calculate the MSD values of four wave selection networks, DARecNet-BS[23]、BSNet-Conv[14]、PCA[24]、SpaBS[25] , with the same wave subset size to compare with our results. All the MSD values are displayed in a single plot with the same coordinate system, as shown in Figure 5. The MSD values obtained in the initial few bands fluctuate greatly, and the MSD values obtained in the subsequent bands stabilize. It can be clearly seen that the subset of bands selected by our network yields higher MSDs than all other networks on most of the bands. This proves the superiority of MSARPBS.

Table 1: Quantitative comparison of MSD values for the last three bands in the subset of bands selected by each algorithm

Method	last but two	last but one	Last
DARecNet-BS[23]	24.8181	24.4502	25.0194
BSNet-Conv[14]	26.7654	26.6518	26.1704
PCA[24]	16.3033	15.7755	15.2471
SpaBS[25]	24.5749	24.2607	24.5354
MSARPBS （Our work）	**28.4215**	**28.6040**	**27.7628**

From Figure 5, it can be seen that the MSD values obtained from the subset of bands selected by all networks tend to be more stable the closer they are to the last band, so the later bands can be considered more representative. We have evaluated and quantified the MSD values obtained from the last three bands of the subset of bands selected by each network, and the results are shown in Table 1. Compared to the other algorithms, it can be seen that our MSARPBS performs the best in the quantitative evaluation. The MSD values in the last band are 25.0194 for DARecNet-BS, 26.1704 for BSNet-Conv, 15.2471 for PCA, 24.5354 for SpaBS, and 27.7628 for MSARPBS.The MSARPBS is 10.97% higher than DARecNet-BS, 10.97% higher than

DARecNet-BS, and 27.7628 higher than MSARPBS. 10.97% compared to DARecNet-BS, 6.08% compared to BSNet-Conv, 82.09% compared to PCA, and 13.15% compared to SpaBS. From the results of quantitative comparison of this paper's algorithm with DARecNet-BS, BSNet-Conv, PCA and SpaBS algorithms, it can be seen that the framework model designed in this paper has a higher band selection accuracy compared to other methods. This is due to the fact that our proposed MSARPBS utilizes the band Multi head Self-Attention mechanism to extract the channel weights, and uses convolution and deconvolution for feature extraction and image reconstruction as well as reflective padding to obtain edge features. It can extract all features of hyperspectral images well, and has achieved the purpose of selecting a subset of bands with high accuracy to reduce the redundancy of hyperspectral images.

IV. CONCLUSION

In this paper, we propose a Multi head Self-Attention Reflection Padding Band Selection Network (MSARPBS) for band selection of hyperspectral images. This approach for hyperspectral image band selection ensures end-to-end training by obtaining channel weights through BMSA, extracting image edge features through RP, extracting image features using FER, and recovering image dimensions. Compared to some state-of-the-art hyperspectral image band selection models, MSARPBS achieves the most accurate band subset selection on scoring via MSD. However, since our network is still a supervised approach, it can only rely on a small number of existing hyperspectral classification datasets for training, which can lead to insufficient model training. Therefore, in our future research we will unsupervised method for band selection of hyperspectral images to further improve MSARPBS and provide more accurate subset of bands for hyperspectral image dimensionality reduction.

REFERENCES

[1] AHISHALI M, KIRANYAZ S, AHMAD I, et al. SRL-SOA: SELF-REPRESENTATION LEARNING WITH SPARSE 1D-OPERATIONAL AUTOENCODER FOR HYPERSPECTRAL IMAGE BAND SELECTION; proceedings of the 29th IEEE International Conference on Image Processing, ICIP 2022, October 16, 2022 - October 19, 2022, Bordeaux, France, F, 2022 [C]. IEEE Computer Society.

[2] LIU K-H, CHEN Y-K, CHEN T-Y. A Band Subset Selection Approach Based on Sparse Self-Representation and Band Grouping for Hyperspectral Image Classification [J]. Remote Sensing, 2022, 14(22).

[3] MINOCHA S, SINGH B. Band selection technique based on binary modified equilibrium optimizer for hyperspectral image classification [J]. Journal of Applied Remote Sensing, 2022, 16(4).

[4] PAUL A, CHAKI N. Band selection using spectral and spatial information in particle swarm optimization for hyperspectral image classification [J]. Soft Computing, 2022, 26(6): 2819-34.

[5] XU J, YAN G, ZHAO X, et al. Soft hypergraph regularized weighted low rank subspace clustering for hyperspectral image band selection [J]. International Journal of Remote Sensing, 2022, 43(14): 5348-71.

[6] GOEL A, MAJUMDAR A. K-Means Embedded Deep Transform Learning for Hyperspectral Band Selection [J]. IEEE Geoscience and Remote Sensing Letters, 2022, 19.

[7] ZENG M, CAI Y, CAI Z, et al. Unsupervised Hyperspectral Image Band Selection Based on Deep Subspace Clustering [J]. IEEE Geoscience and Remote Sensing Letters, 2019, 16(12): 1889-93.

[8] JU H, LIU Q, GAO H, et al. Linear Prediction Band Selection Based on Schmidt Orthogonalization for Hyperspectral Image; proceedings of the 2022 International Conference on Signal Processing, Computer Networks, and Communications, SPCNC 2022, December 16, 2023 - December 17, 2023, Zhengzhou, China, F, 2023 [C]. SPIE.

[9] CHANG D, LEE J, JEONG B, et al. Band selection techniques using discrete ranges and maxpooling operations for hyperspectral image pixel classification; proceedings of the 37th ACM/SIGAPP Symposium on Applied Computing, SAC 2022, April 25, 2022 - April 29, 2022, Virtual, Online, F, 2022 [C]. Association for Computing Machinery.

[10] JIA Y, SHI Y, LUO J, et al. Y–Net: Identification of Typical Diseases of Corn Leaves Using a 3D–2D Hybrid CNN Model Combined with a Hyperspectral Image Band Selection Module [J]. Sensors, 2023, 23(3).

[11] ESMAEILI M, ABBASI-MOGHADAM D, SHARIFI A, et al. Hyperspectral Image Band Selection Based on CNN Embedded GA (CNNeGA) [J]. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, 2023, 16: 1927-50.

[12] LIU Y, LI X, HUA Z, et al. A Band Selection Method with Masked Convolutional Autoencoder for Hyperspectral Image [J]. IEEE Geoscience and Remote Sensing Letters, 2022, 19.

[13] LUNA G L M, SHEPPARD J, LOGAN R, et al. Hyperspectral band selection for multispectral image classification with convolutional networks [Z]. arXiv. 2021

[14] CAI Y, LIU X, CAI Z. BS-Nets: An end-to-end framework for band selection of Hyperspectral image [Z]. arXiv. 2019

[15] WANG J, ZHOU J, HUANG W, et al. Attention Networks for Band Weighting and Selection in Hyperspectral Remote Sensing Image Classification; proceedings of the 39th IEEE International Geoscience and Remote Sensing Symposium, IGARSS 2019, July 28, 2019 - August 2, 2019, Yokohama, Japan, F, 2019 [C]. Institute of Electrical and Electronics Engineers Inc.

[16] MOU L, SAHA S, HUA Y, et al. Deep reinforcement learning for band selection in hyperspectral image classification [Z]. arXiv. 2021

[17] ZHAO L, TAN K, WANG X, et al. Hyperspectral Feature Selection for SOM Prediction Using Deep Reinforcement Learning and Multiple Subset Evaluation Strategies [J]. Remote Sensing, 2023, 15(1).

[18] YANG H, CHEN M, WU G, et al. Double Deep Q-Network for Hyperspectral Image Band Selection in Land Cover Classification Applications [J]. Remote Sensing, 2023, 15(3).

[19] BAO D, TUXWORTH G, ZHOU J. Similarity-Based Hyperspectral Band Selection Using Deep Reinforcement Learning; proceedings of the 12th Workshop on Hyperspectral Imaging and Signal Processing: Evolution in Remote Sensing, WHISPERS 2022, September 13, 2022 - September 16, 2022, Rome, Italy, F, 2022 [C]. IEEE Computer Society.

[20] AYNA C O, MDRAFI R, DU Q, et al. Learning-Based Optimization of Hyperspectral Band Selection for Classification [J]. Remote Sensing, 2023, 15(18).

[21] MDRAFI R, GURBUZ A C. Data Driven Joint Hyperspectral Band Selection and Image Classification; proceedings of the 2022 IEEE International Geoscience and Remote Sensing Symposium, IGARSS 2022, July 17, 2022 - July 22, 2022, Kuala Lumpur, Malaysia, F, 2022 [C]. Institute of Electrical and Electronics Engineers Inc.

[22] VASWANI A, SHAZEER N, PARMAR N, et al. Attention Is All You Need [J]. arXiv, 2017.

[23] ROY S K, DAS S, SONG T, et al. DARecNet-BS: Unsupervised Dual-Attention Reconstruction Network for Hyperspectral Band Selection [J]. IEEE Geoscience and Remote Sensing Letters, 2021, 18(12): 2152-6.

[24] CHANG C-I, DU Q, SUN T-L, et al. Joint band prioritization and band-decorrelation approach to band selection for hyperspectral image classification [J]. IEEE Transactions on Geoscience and Remote Sensing, 1999, 37(6): 2631-41.

[25] SUN K, GENG X, JI L. A new sparsity-based band selection method for target detection of hyperspectral image [J]. IEEE Geoscience and Remote Sensing Letters, 2015, 12(2): 329-33.

[26] ZHOU W, BOVIK A C, SHEIKH H R, et al. Image quality assessment: from error visibility to structural similarity [J]. IEEE Transactions on Image Processing, 2004, 13(4): 600-12.

[27] [CP/OL].pytorch.org,[2024-01-02]. https://pytorch.org/docs/stable/generated/torch.nn.ReflectionPad2d.html#torch.nn.ReflectionPad2d.