ISSN 2616-5775 Vol. 4, Issue 7: 11-17, DOI: 10.25236/AJCIS.2021.040703

Deep Transformer Network for Hyperspectral Image Classification

Kuiliang Gao[a,*], Bing Liu[b], Zhixiang Xue[c], Xibing Zuo[d], Yifan Sun[e], Mofan Dai[f]

PLA Strategic Support Force Information Engineering University, Zhengzhou, China
[a]gokling1219@163.com, [b]liubing220524@126.com, [c] xuegeeker@163.com, [d]2812463301@qq.com, [e]sincere_sunyf@163.com, [f]daidai_1115@163.com

Abstract: *Different from the existing CNN-based models, a novel method based on the transformer model is proposed in this paper, to further improve the classification accuracy of hyperspectral image (HSI). Specifically, a deep network model is constructed with the Transformer-iN-Transformer (TNT) modules, to carry out end-to-end classification. The outer and inner transformer models in the TNT module can extract the patch-level and pixel-level features respectively, to make full use of the global and local information in the input cubes. Experimental results show that the proposed method can achieve better classification performance than the existing CNN-based models. In addition, using the transformer-baesd deep model without convolution to classify HSI provides a new idea for the related researches.*

Keywords: *Hyperspectral image classification, deep learning, transformer, TNT, self-attention*

1. Introduction

In the field of remote sensing, hyperspectral image (HSI) classification has always been one of the most attractive research hotspots. In the whole process of HSI processing and analysis, HSI classification is one of the most important links, and its accurate classification results can provide strong data support for the follow-up tasks, which has been widely used in fine agriculture, land-use planning and many other fields[1].

In the early research of HSI classification, machine learning methods such as support vector machine (SVM), principal component analysis (PCA) and extended morphological profile (EMP) have been widely used[2]. However, the above methods cannot make full use of the deep abstract features in HSI, so they cannot obtain satisfactory classification results. In contrast, deep learning methods can automatically learn the deep abstract features conducive to the target tasks layer by layer, and these features are often highly informative and robust. Deep learning models such as stacked autoencoder (SAE), recurrent neural network (RNN), deep belief networks (DBN) and convolutional neural networks (CNN) have been widely applied in HSI classification[3], and have achieved better classification performance than traditional classification methods with sufficient training samples. Compared with other deep learning models, 2D-CNN and 3D-CNN can make more full use of the spatial-spectral information utilizing unique convolution operation to directly process the HSI data with grid structure, so they can obtain higher classification accuracy[4]. For example, Lee et al designed a novel contextual deep CNN model by introducing multi-scale 2D convolutional filter bank and residual connection[5]. Liu et al. constructed a deep spatial-spectral network utilizing 3D convolution, and the obtained classification accuracy was obviously better than that of the traditional methods[6]. In recent years, novel network structures such as generative adversarial networks (GAN), capsule network (CN) and graph convolutional network (GCN) have been introduced into HSI classification, to further improve the classification performance[7]. In the above network models, CNN is always an indispensable and important module.

As is known to all, CNN remains dominant in HSI classification. However, CNN indeed possesses some defects: it is not good at modeling the long-distance dependencies and obtaining global context information[8]. By contrast, the transformer model can better utilize the global context information within a large range by treating the input image as the sequential patches[9]. In view of this, this paper presents a novel deep transformer network, to further improve the accuracy of HSI classification. Specifically, the deep network model is constructed with Transformer-iN-Transformer (TNT)[10] structure as the basic module. The inner transformer block and the outer transformer block in the TNT

ISSN 2616-5775 Vol. 4, Issue 7: 11-17, DOI: 10.25236/AJCIS.2021.040703

module can respectively model the pixel-level and patch-level features, to make full use of the local and global information in the input cubes. Experimental results on two public HSI data sets show that the proposed method can achieve better classification performance than the existing CNN models. Meanwhile, using transformer-based deep model without any convolution operation to classify HSI in this paper provides a new idea for HSI classification.

2. Proposed Method

2.1. Workflow

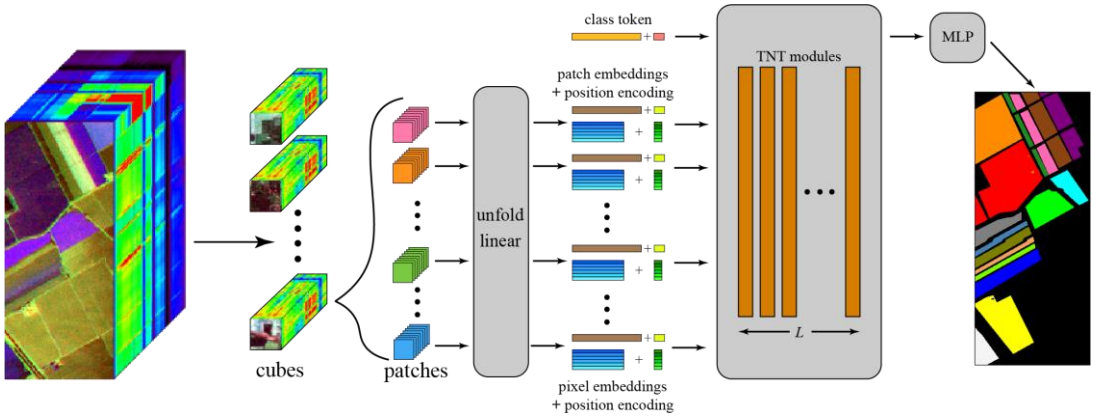

Figure 1: Workflow of the proposed method.

Figure. 1 shows the workflow of the proposed method. Firstly, the HSI is directly divided into a number of cubes without dimensionality reduction, to make full use of the spatial-spectral information. Then, each cube is sequentially divided into multiple patches referring to [9]. After unfold and linear transformation operations, each patch is converted into a patch embedding and several pixel embeddings. Finally, the patch embeddings and pixel embeddings are respectively added to their corresponding position encodings, and the resulting vectors are input together into the designed deep network model containing L TNT modules for classification. In the TNT modules, the outer transformer models can take full advantage of the global information, and the inner transformer models can take full advantage of the local information. The proposed method takes HSI as input and classification results as output, and possesses an end-to-end structure.

2.2. TNT Module

The TNT module is the core of the proposed method which can make full use of the global and local information in the input cubes. Before introducing the TNT module, we first review the vision transformer model that is widely used for the computer vision tasks.

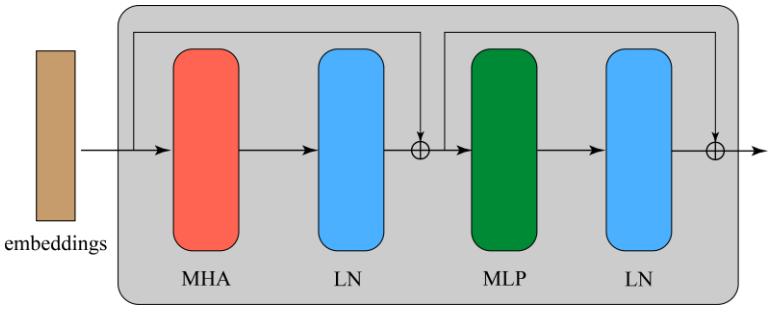

Figure 2: Illustration of a vision transformer model.

Figure 2 describes the network structure of a vision transformer model, which consists of three main parts: multi-head attention mechanism (MHA), multilayer perceptron and layer normalization (LN). MHA is the key part in the transformer model for feature learning, and MLP is introduced for feature transformation and nonlinearity. As a data normalization layer, LN can ensure the training stability and rapid convergence of the model. In addition, residual connections are introduced to take full advantage of the abstract features at different levels. Self-attention mechanism is the basic unit of

ISSN 2616-5775 Vol. 4, Issue 7: 11-17, DOI: 10.25236/AJCIS.2021.040703

MHA. In the self-attention mechanism, the input embeddings are firstly transformed to query matrix, key matrix and value matrix. The output actually is the weighted sum of the value vectors, and the weights assigned to each value vector are calculated from the query vectors and the corresponding key vectors.

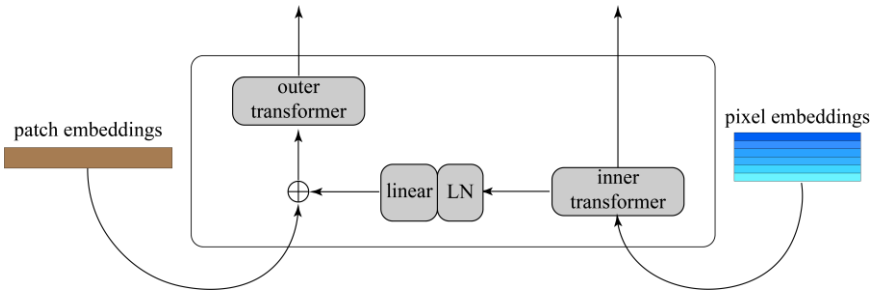

Figure 3: Illustration of a TNT module.

The visual transformer model views an input image as a sequence of patches, but ignores the intrinsic structure information inside each patch. To make full use of the global and local information within the input cubes, TNT module is introduced as the core of constructing the deep network model. For the input cube, we first split it into n patches $\mathcal{X}=\left[X^1, X^2, \ldots, X^n\right] \in \mathbb{R}^{n \times p \times p \times C}$, where (p, p) is the size of each patch. Then, each patch is further transformed into multiple (p', p') pixel embeddings with unfold operation and linear projection. Formally, the sequence of patch tensors is as follows:

$$\begin{aligned} \mathcal{Y} &= \left[Y_0^1, Y_0^2, \ldots, Y_0^n\right] \in \mathbb{R}^{n \times p' \times p' \times C}, \\ Y_0^i &= [y_0^{i,1}, y_0^{i,2}, \ldots, y_0^{i,m}], \end{aligned} \tag{1}$$

where each patch tensor Y_0^i is viewed as a sequence of pixel embeddings, and $m = p'^2$.

As shown in Figure 3, a TNT module contains two transformer models, one of which operates across the patch tensors and the other operates across the pixel embeddings. For the pixel embeddings, the inner transformer is used to explore the relation between pixels and extract the pixel-level features:

$$\begin{aligned} Y'^i_l &= Y^i_{l-1}, + LN(MSA(Y^i_{l-1})), \\ Y^i_l &= Y'^i_l, + LN(MLP(Y'^i_l)), \end{aligned} \tag{2}$$

where l = 1, 2,..., L index the layers and L is the total number of layers. This process can build the relationship among pixels by computing interactions between two pixel embeddings, to effectively utilize the local information within the input cubes. As for the patch level, the patch embedding memories are created to store the sequence of patch-level features: $\mathcal{Z}_0=\left[Z_{class}, Z_0^1, Z_0^2, \ldots, Z_0^n\right] \in \mathbb{R}^{(n+1) \times d}$, where Z_{class} represents the class token. In each TNT module, the patch tensors are transformed into the domain of patch embeddings by linear projection and added into the patch embeddings:

$$Z^i_{l-1} = Z^i_{l-1} + Vec(Y^i_{l-1})W_{l-1} + b_{l-1}, \tag{3}$$

where Vec represents the flatten operation, W and b are the learnable weights and bias respectively. Similarly, a transformer module (outer transformer) is used for feature learning:

$$\begin{aligned} \mathcal{Z}'^i_l &= \mathcal{Z}^i_{l-1} + LN(MSA(\mathcal{Z}^i_{l-1})), \\ \mathcal{Z}^i_l &= \mathcal{Z}'^i_l, + LN(MLP(\mathcal{Z}'^i_l)). \end{aligned} \tag{4}$$

This process can effectively learn the patch-level features by capturing the intrinsic information from the sequence of patches. In other words, the outer transformer can make full use of the global information in the input data.

As is mentioned above, the TNT module can process pixel-level and patch-level data simultaneously. This means that the deep network model built by stacking TNT modules can take full advantage of the global and local information in the cubes and learn richer and more robust features, to further improve the accuracy of HSI.

ISSN 2616-5775 Vol. 4, Issue 7: 11-17, DOI: 10.25236/AJCIS.2021.040703

3. Experimental Results and Analysis

3.1. Data Sets and Evaluation Criteria

To verify the effectiveness of the proposed method, two public HSI, Salinas (SA) and Houston 2013(HS) are selected for experiments. The SA data set is collected by AVIRIS sensor. It possesses 204 spectral bands and 16 labelled classes. The spatial size is 512 × 217 with 3.7 m/ pixel spatial resolution. The HS data set is collected by ITRES-CASI1500 sensor. It possesses 144 spectral bands and 15 labelled classes. The spatial size is 349 × 1905 with 2.5 m/ pixel spatial resolution. For each data set, 200 labeled samples per class are randomly selected as training samples, and the remaining samples are used as testing samples to evaluate the classification performance of the model. To quantitatively evaluate the classification results, the overall accuracy (OA), average accuracy (AA) and kappa coefficient are selected as the evaluation criteria.

3.2. Hyperparameters Settings

Firstly, we explore the influence of the depth of network on classification accuracy. Figure 4 describes the relationship between the depth of network and the classification accuracy. For the SA and HS data sets, the classification accuracy of the model generally increases first and then decreases with the increase in the number of TNT modules. This indicates that, appropriate network structure can achieve the best classification performance, while too many or too few network layers may lead to a decline in classification accuracy.

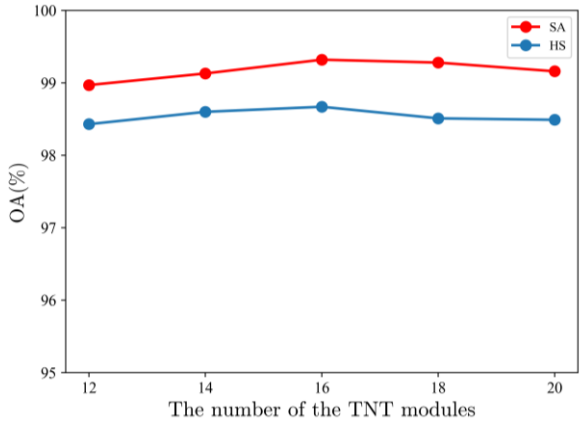

Figure 4: Relationship between the number of the TNT modules and classification accuracy on the SA and HS data sets.

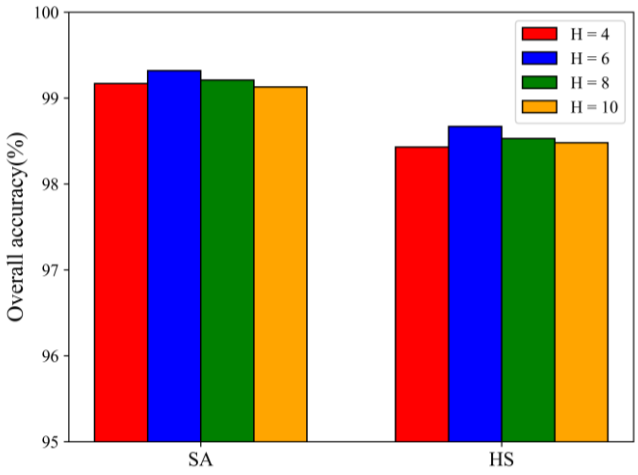

Figure 5: Relationship between the number of attention heads (H) and classification accuracy on the SA and HS data sets.

Secondly, the relationship between the number of attention heads and the classification accuracy is analysed. Theoretically, similar to the convolution kernels in the CNN models, an appropriate increase

ISSN 2616-5775 Vol. 4, Issue 7: 11-17, DOI: 10.25236/AJCIS.2021.040703

in the number of attention heads should enable the model to learn richer and more robust features. The results are shown in Fig. 5. In general, with the increase of H, the classification accuracy of the model increases gradually at first and then decreases slowly. When H is equal to 6, the classification accuracy reaches the maximum value.

In addition, a combination of large iteration times and small learning rate is adopted for training. Specifically, the number of training iterations is set as 500, the learning rate is set as 0.00001, and the Adam optimization algorithm is adopted to ensure that the designed model is fully trained. The batch size is set as 64. The input cube is first split into 16 patches in a spatial order, and each patch is further split into several pixel patches with a width of 2. In the TNT modules, the dimensions of patch embedding and pixel embedding are set to 128 and 64 respectively.

3.3. Comparison and Analysis

Different from the dominant CNN-based model for HSI classification, this paper proposes a novel deep transformer network, to further improve the accuracy of HSI classification. To verify the effectiveness of the proposed method, the classification results are compared with the classic machine learning method RBF-SVM and 5 advanced CNN-based models including CNN-PPF[11], CDCNN[5], RES-3D-CNN[6], DCCNN[12] and S-CNN[13]. In addition, to reduce the fluctuation of classification results caused by the randomness of sample selection, the average value of 10 experiments is used as the final result, further enhancing the persuasiveness of the experimental results.

Table 1: The classification results OF different methods on the SALINAS data set.

Class No.	SVM	CNN-PPF	CDCNN	RES-3D-CNN	S-CNN	Ours
OA	91.20	92.04	95.54	97.39	96.06	99.32
AA	95.46	95.03	97.32	98.99	98.19	99.54
kappa	90.20	91.17	95.04	97.09	95.48	99.25
1	99.20	100.00	99.50	100.00	100.00	100.00
2	99.62	99.76	100.00	100.00	100.00	100.00
3	99.70	97.26	98.21	100.00	99.10	100.00
4	99.50	97.20	99.43	99.36	99.86	99.43
5	96.75	98.61	99.96	99.78	99.93	99.87
6	99.42	99.62	99.95	100.00	100.00	99.98
7	99.36	99.97	99.71	99.97	99.78	100.00
8	84.75	88.15	94.58	91.15	92.80	99.42
9	99.10	98.83	99.98	99.94	99.97	99.90
10	93.29	85.42	99.78	99.15	97.80	99.13
11	97.85	91.21	91.96	99.07	98.40	100.00
12	99.84	99.23	99.90	100.00	100.00	98.17
13	98.58	98.49	100.00	100.00	100.00	99.63
14	95.79	96.56	99.17	99.91	100.00	100
15	65.74	73.75	81.13	95.55	83.51	97.16
16	98.89	96.49	93.89	100.00	99.83	100.00

Tables 1-2 lists the classification results of different methods on the SA and HS data sets. As we can see, the classification accuracy of SVM is obviously lower than that of the other 5 deep learning-based methods. Deep learning models can extract the deep abstract features that are more informative and robust, so they can obtain higher classification accuracy. Among the four CNN-based models, RES-3D-CNN using 3D convolution and metric learning-based S-CNN can achieve better classification performance. The proposed method can achieve better classification performance. Among all the listed methods, the proposed method can obtain the best classification results according to OA, AA and kappa. On the one hand, MHA enables the model to focus on a wealth of features conducive to the classification tasks, on the other hand, the TNT modules containing the inner and

ISSN 2616-5775 Vol. 4, Issue 7: 11-17, DOI: 10.25236/AJCIS.2021.040703

outer transformer can enable the model to make full use of the global and local information in the input cubes, to further improve the classification accuracy.

Table 2: The classification results OF different methods on the HOUSTON 2013 data set.

Class No.	SVM	CNN-PPF	CDCNN	RES-3D-CNN	S-CNN	Ours
OA	91.41	94.50	95.34	96.03	98.12	98.68
AA	91.95	94.76	95.84	96.56	98.41	98.77
kappa	90.71	94.06	94.97	95.71	97.96	98.57
1	95.73	96.71	84.49	87.14	99.66	99.76
2	97.94	98.71	97.42	97.97	93.67	97.73
3	100.00	99.86	99.71	99.85	99.71	99.86
4	99.83	99.27	99.32	96.55	98.78	97.13
5	96.39	97.85	97.69	99.92	99.84	99.76
6	99.69	100.00	94.20	95.03	100.00	98.78
7	84.71	94.35	97.36	97.86	97.51	95.77
8	95.19	96.67	96.65	95.10	100.00	99.51
9	82.19	88.23	92.31	95.58	97.66	97.98
10	86.03	92.26	94.02	92.50	95.64	99.11
11	86.68	88.70	98.01	99.50	99.19	99.76
12	83.59	88.42	94.48	94.05	97.44	98.95
13	73.57	83.96	97.61	97.40	99.12	99.58
14	97.71	96.61	98.16	100.00	99.77	98.39
15	100.00	99.85	96.21	100.00	98.21	99.55

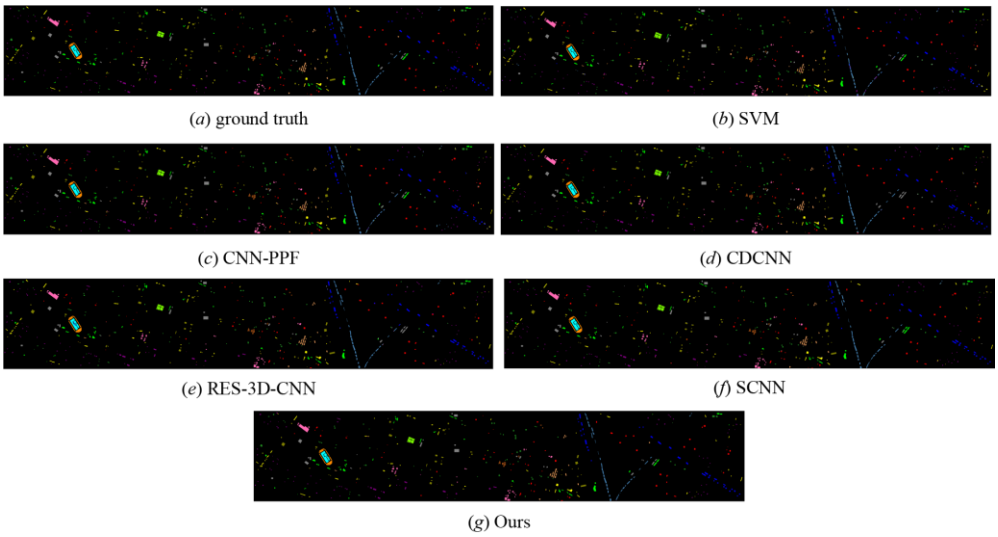

Figure 6: Classification maps resulting from the different methods on the HS data set.

Figure 6 shows the classification maps of the different classification methods on the HS data set. It can be seen that the proposed method can produce the classification map closest to the ground truth, which visually verifies the effectiveness of the proposed method.

4. Conclusions

Different the dominant CNN-based methods, this paper designs a novel deep transformer network by stacking the TNT modules, to further improve the accuracy of HSI classification. The inner and outer transformer block in the TNT modules can extract the pixel-level and patch-level features

ISSN 2616-5775 Vol. 4, Issue 7: 11-17, DOI: 10.25236/AJCIS.2021.040703

respectively, making full use of the global and local information in the input HSI cubes. Experimental results on two public HSI data sets show that the proposed method performs better than SVM and several existing CNN-based models.

Acknowledgments

We gratefully acknowledge the financial support by the National Natural Science Foundation of China (Grants Nr. 41801388).

References

[1] N. Audebert, B. L. Saux, and S. Lefevre, "Deep Learning for Classification of Hyperspectral Data: A Comparative Review," IEEE Geoscience and Remote Sensing Magazine, vol. 7, no. 2, pp. 159-173, 2019, doi: 10.1109/MGRS.2019.2912563.

[2] P. Ghamisi, J. Plaza, Y. Chen, J. Li, and A. J. Plaza, "Advanced Spectral Classifiers for Hyperspectral Images: A review," IEEE Geoscience and Remote Sensing Magazine, vol. 5, no. 1, pp. 8-32, 2017.

[3] M. J. Khan, H. S. Khan, A. Yousaf, K. Khurshid, and A. Abbas, "Modern Trends in Hyperspectral Image Analysis: A Review," IEEE Access, vol. 6, pp. 14118-14129, 2018.

[4] M. E. Paoletti, J. M. Haut, J. Plaza, and A. Plaza, "Deep learning classifiers for hyperspectral imaging: A review," ISPRS Journal of Photogrammetry and Remote Sensing, vol. 158, no. Dec., pp. 279-317, 2019.

[5] H. Lee and H. Kwon, "Going Deeper With Contextual CNN for Hyperspectral Image Classification," IEEE Trans Image Process, vol. 26, no. 10, pp. 4843-4855, 2017.

[6] Y. X. LIU Bing, ZHANG Pengqiang, TAN Xiong, "Deep 3D convolutional network combined with spatial-spectral features for hyperspectral image classification," Acta Geodaetica et Cartographica Sinica, vol. 48, no. 1, pp. 53-63, 2019-01-20 2019, doi: 10.11947/j.AGCS.2019.20170578.

[7] S. Shabbir and M. Ahmad, Hyperspectral Image Classification -- Traditional to Deep Models: A Survey for Future Prospects. 2021.

[8] A. Vaswani et al., "Attention Is All You Need," 06/12 2017.

[9] A. Dosovitskiy et al., An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. 2020.

[10] K. Han, A. Xiao, E. Wu, J. Guo, C. Xu, and Y. Wang, Transformer in Transformer. 2021.

[11] W. Li, G. Wu, F. Zhang, and Q. Du, "Hyperspectral Image Classification Using Deep Pixel-Pair Features," IEEE Transactions on Geoscience and Remote Sensing, vol. 55, no. 2, pp. 844-853, 2017, doi: 10.1109/TGRS.2016.2616355.

[12] H. Zhang, Y. Li, Y. Zhang, and Q. Shen, "Spectral-spatial classification of hyperspectral imagery using a dual-channel convolutional neural network," Remote Sensing Letters, vol. 8, no. 5, pp. 438-447, 2017/05/04 2017.

[13] B. Liu, X. Yu, P. Zhang, A. Yu, Q. Fu, and X. Wei, "Supervised Deep Feature Extraction for Hyperspectral Image Classification," IEEE Transactions on Geoscience and Remote Sensing, vol. 56, no. 4, pp. 1909-1921, 2018.