Expert Systems With Applications 250 (2024) 123939

Contents lists available at ScienceDirect

Expert Systems With Applications

journal homepage: www.elsevier.com/locate/eswa

RDTN: Residual Densely Transformer Network for hyperspectral image classification

Yan Li [a,1], Xiaofei Yang [b,1,*], Dong Tang [b], Zheng Zhou [b]

[a] Department of Artificial Intelligence, Shenzhen Polytechnic University, Shenzhen 518055, China
[b] School of Electronics and Communication Engineering, Guangzhou University, Guangzhou 510006, China

ARTICLE INFO

Keywords:
Hyperspectral image classification
Transformers
Convolution neural network

ABSTRACT

Transformer-based methods have achieved significant success in hyperspectral image (HSI) classification, which attribute to the strong capability of capturing the global dependencies from the input. However, the existing Transformer-based HSI classification methods are challenged in retrieving sufficient abound local information using the linear projection modules. Moreover, they do not fully use the hierarchical representations extracted from the original hyperspectral images. To overcome these challenges, this paper proposes a novel transformer model, called Residual Densely Transformer Network (RDTN) to comprehensively exploit the multi-hierarchical features and the local–global dependencies along the spatial–spectral dimensions from the hyperspectral images. Specially, the proposed RDTN is built with two modules: a Cross-Scale Convolution Attention (CSCA) module to extract abundant local spatial–spectral features using the multiscale convolution attention layers, and a Local Residual Transformer Block (LRTB) to respectively capture the abundant global dependencies along the spatial–spectral dimensions. Additionally, LRTB uses a residual connection operation to make full use of the hierarchical representations of all the transformer encoder layers. After acquiring dense global representations, we introduce a Global Residual Connection (GRC) to jointly fuse the local features obtained by CSCA, and then feed the fusion representations into the final avg-pooling layer and a classifier to predict the category. Finally, we conduct the extensive experiments based on four public benchmarks datasets, whose results are demonstrated that the proposed RDTN outperforms the state-of-the-art methods. The codes of this work are available at https://github.com/xiachangxue/DeepHyperX.

1. Introduction

With the rapid development of satellite sensors, it is easy and possible to acquire an increasing number of hyperspectral images (HSIs). Each pixel from hyperspectral images is composed of much more spectral information that are recorded as hundreds of bands. Thus, hyperspectral images could offer both significantly abundant spatial and spectral information to enable various fine-grained Earth observation missions, such as monitoring urban environments (Dupont et al., 2020), classifying land cover (Li et al., 2020; Navin & Agilandeeswari, 2020), and detecting urban change (Hou et al., 2021; Tang et al., 2024). A technique that is frequently used in these applications is to identify each pixel in a HSI. However, the high dimension and increasing spatial resolution of HSIs lead to new challenges within traditional HSIs classification tasks. Thus, accurate HSI classification has always been a focus of research in remote sensing.

There are two major factors that contribute to the inaccuracy of the HSI classification. On the one hand, HSIs exhibit high-dimensional properties, which inevitably results in the curse of dimensionality known as the Hughes effect (Hughes, 1968). Nevertheless, typical classification techniques perform well on small samples while poorly on large complicated hyperspectral data. For example, the Support Vector Machine (SVM) (Scholkopf & Smola, 2001), and K-nearest neighbor (KNN) (Ma et al., 2010) classifiers. To address this issue, many researcher used dimension reduction to process the data first and then classify the pixels. For instance, there are several established linear feature extraction techniques, such as Principal Component Analysis (PCA) (Wold et al., 1987), Independent Components Analysis (ICA) (Villa et al., 2011), and Nonparametric Weighted Feature Extraction (NFWE) (Kuo & Landgrebe, 2004). However, they eliminate specific spectrum information during the dimension reduction process, which results in the loss of hyperspectral properties. On the other hand, standard classification techniques identify pixels only based on their separate spectral curves, disregarding spatial information.

* Corresponding author.
E-mail addresses: liyan@szpu.edu.cn (Y. Li), xiaofeiyang@gzhu.edu.cn (X. Yang), tangdong@gzhu.edu.cn (D. Tang), zhouzheng@gzhu.edu.cn (Z. Zhou).
[1] Equal Contribution.

https://doi.org/10.1016/j.eswa.2024.123939
Received 25 February 2024; Received in revised form 1 April 2024; Accepted 6 April 2024
Available online 8 April 2024

In the past years, deep learning techniques have achieved significant success in natural image recognition (Hu et al., 2024). For example, Wu et al. (2023) proposed a lightweight image segmentation network for Dam Crack Width Measurement by integrating a multifeature fusion structure in ASPP, and improved the performance. Inspired by the powerful feature extraction ability of deep learning methods, numerous studies employed deep learning networks to HSI classification, including Recurrent Neural Networks (RNNs) (Schuster & Paliwal, 1997) and Convolution Neural Networks (CNNs) (Yan et al., 2015), and developed a variety of deep learning-based classification approaches. For example, Mou et al. (2017) considered the HSI classification from the spectral dimension and established a novel RNN-based method to classify hyperspectral image using a recurrent layer with 64 neuron units. Yang et al. (2018) employed CNNs for HSIs classification and built CNNs-based methods using 2D (or 3D) convolution layers, Batch Normalization (BN) (Ioffe & Szegedy, 2015) layers and Rectified Linear Units (ReLU) (Glorot et al., 2011) layers. These CNN-based methods have achieved the higher-precision classification results of HSI, which attributes to the powerful capability of exploring local spatial contextual information. However, they are unable to capture subtle spectral discrepancies across adjacent spectral bands, limiting their ability to improve classification accuracy further.

Recently, the transformer networks (Vaswani et al., 2017) are applied in the natural image classification task and have achieved impressive results. Different from the CNNs and RNNs, transformer networks can process the images from a sequence perspective and mine the long-term dependence of data. This is mainly because of the self-attention techniques. Vision image Transformer (ViT) network is the first study to introduce transformer architecture in the image recognition task by Dosovitskiy et al. (2020). Different researchers have changed ViT in a variety of ways. For example, Zhou et al. (2021) built a DeepViT classification network by using a learnable matrix to re-calculate the attention maps. Touvron et al. (2021) devised a novel transformer network for image classification, which is named CaiT. However, the previous transformer-based methods have a failure in capturing the local spatial contextual information. To solve this problem, some researchers improved it by inserting the convolution operations. For example, Graham et al. (2021) devised a novel transformer-based network, namely LeViT for image classification adopting convolution operations to project the input images. Heo et al. (2021) employed some beneficial modules to build a novel transformer-based network, which is named Robust Vision Transformer (RvT) for image classification.

Now, several transformer-based networks are used in solving HSI classification task and many transformer-based HSI classification algorithms have been presented. For instance, He et al. (2019) attempted to directly apply the Transformer architecture in HSIs classification and achieved significant classification results. However, existing transformer-based classification networks suffer from the following drawbacks:

1. the convolutional operation was frequently used to exploit local spatial contextual features in existing transformer-based networks. However, the classification accuracy cannot be enhanced, since a fixed receptive field convolution cannot capture more abundant 2D spatial information. On the other hand, it will result in a significant increase in processing cost and parameters due to the larger kernel convolution.
2. With increasing depth in network, the representation of each attention block becomes hierarchical and acquires specific features. They can provide a lot of information that can assist in item identification. However, the previous transformer networks (*e.g.*, ViT, Deep ViT, and CaiT) disregard to fully use the hierarchical representation.

To address these issues, we propose a novel transformer-based classification network called Residual Densely Transformer Network (RDTN), which makes extensive use of hierarchical representation and captures multiscale local spatial features to further improve HSI classification accuracy. More specifically, RDTN is composed of three modules: the Cross-Scale Convolutional Attention Projection (CSCA), the Local Residual Transformer Block (LRTB), and the Global Residual Connection (GRC). Using the large receptive fields, the CSCA module captures a wealth of local spatial–spectral information. The LRTB makes comprehensive use of the hierarchical representation. The GRC process would be employed to determine the categories by utilizing more abundant local spatial–spectral information. We summary the contributions of this paper, and list them as follows:

1. We establish a novel transformer method called Residual Densely Transformer Network (RDTN) to HSI classification task. This is first study in which the hierarchical representation is fully utilized in HSI classification field.
2. We develop a module called Cross-Scale Convolutional Attention Projection (CSCA) to collect more abundant local spatial information and thus improve classification accuracy.
3. We propose two key modules: a Local Residual Transformer Block (LRTB) to capture the hierarchical representation and a Global Residual Connection (GRC) to integrate the shallow spatial representation and the deep spatial representation.
4. Furthermore, we evaluate the proposed RDTN on four public benchmarks datasets, and the experimental results demonstrate the superiority and effectiveness of the proposed RDTN.

The remainder of this paper is organized as follows. Section 2 will introduce the related works, including the deep learning-based HSIs classification and transformer-based classification. Section 3 presents the detailed introduction of RDTN. Section 4 first gives an illustration of the four benchmarks HSIs datasets and experimental settings, and then presents the results and the analyses of the experiment. Finally, Section 5 conducts a conclusion and the future work.

2. Related works

2.1. Deep learning-based HSIs classification methods

Deep learning methods have achieved significant success in image recognition tasks (e.g., natural image classification), which is attributed to the powerful capability of exploring local contextual features. On this basis, many researchers have applied deep learning to HSI field and built a number of deep learning approaches to HSI classification field by inserting various modules (Boulch et al., 2017; Chen et al., 2014; Fan et al., 2018; Hamida et al., 2018; He et al., 2017; Li et al., 2019; Yang et al., 2020). For example, Mou et al. (2017) and Hang et al. (2019) employed RNN technique for modeling sequence information from HSI and built the RNNs-based HSI classification methods, respectively. However, these RNNs-based methods cannot retrieve the local contextual information in the spatial dimension.

Many researchers have applied CNNs to explore local spatial contextual features for HSI classification, resulting in many CNN-based hyperspectral image classification methods (Chen et al., 2016; Lee & Kwon, 2016; Li et al., 2017; Liu et al., 2017). The proposed CNNs-based HSI classification methods have been achieved great successes, attributing the powerful feature extraction ability. For example, Ran et al. (2016) adopted two CNN branches to separately capture the local contextual features and spectral information, respectively. Some researchers improve performance by inserting various novel techniques. For example, Lorenzo et al. (2020) inserted the attention mechanism for band selection and devised a CNN-based classification network. However, these CNN-based methods separately extracted the spatial and spectral information. To overcome this problem, some researchers utilized the 3D convolution operations to present CNN-based methods. For example, Yang et al. (2018) stacked 3D convolution layers and built 3D-CNNs classification networks (*i.e.*, 3D-CNN and recurrent 3D-CNN). Yang et al. (2020) built a novel hybrid CNNs-based

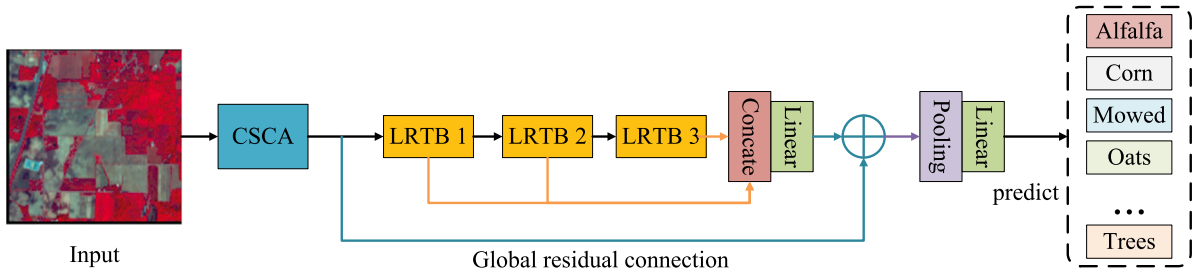


Fig. 1. The illustration of RDTN architecture. The input image are first extracted the local spatial–spectral features using the Cross-scale convolution attention projection (CSCA). Then the extracted features are encoded along the height, width of spatial dimension, and the spectral dimension by using three local residual transformer blocks (LRTB) and these hierarchical representation are concatenated. The extracted representation from CSCA will be added to the encoded representation using a global residual connection layer. Finally, a global average pooling layer and a fully connected layer are utilized to predict the input image.

method by using 2D-CNN for local spatial features extraction and 3D-CNN for local spatial–spectral fusion features extraction, which achieved a satisfactory classification accuracy using a small sample. Owing to the powerful capability of capturing the spatial–spectral fusion features, 3D-CNNs-based methods usually outperform those 2D-CNNs-based methods by using sufficient training samples.

However, current approaches to deep learning HSIs categorization have some limitations. For example, while RNNs excel at modeling sequential data, they may struggle to understand long-term dependencies, which is likely due to their heavy reliance on ordered input. Additionally, RNNs lack the ability to extract local spatial information. In contrast to RNNs, CNNs and their variants are capable of capturing local spatial information. However, they are unable to adequately collect ordered spectral information, owing to their inherent network architecture and excessive focus on extracting local spatial contextual features. All of these limitations limit their ability to enhance their classification accuracy further and contribute to the performance bottleneck in the HSI classification process.

2.2. Transformers-based image classification methods

In recent years, many researchers have applied the transformer network for solving natural image recognition tasks and presented many transformers-based image classification methods. For example, Dosovitskiy et al. (2020) first introduced transformer network into image classification and presented a vision transformer classification network called Vision Transformer (ViT) network, which is a classical transformer network. Specifically, the process of ViT can be divided into four steps: (1) image patch projection: this step first crops the input images into several patches, and then projects the patches to the representation using the linear operation; (2) dimension reduction: this step reduces the dimension of the representation using the linear norm layers; (3) transformer encoder: this is the key step of ViT, which encodes the representation using Multi-head self-attention (MHSA) modules; (4) final step: this step is to classify the image using average pooling layers and linear layers. The ViT performs a potential result, which attributes to the MHSA. However, with increasing depth, the attention maps from different MHSA layers would be similar that is named the attention collapse problem. To address this problem, Zhou et al. (2021) re-calculated the attention maps by inserting a learnable matrix and devised a DeepViT image classification network. Touvron et al. (2021) proposed a new and novel network for image classification that called CaiT, which introduced a learnable parameter in the residual structure. All these transformer-based classification networks have performed satisfactory classification results, however, they cannot explore the local spatial contextual and spectral information, which hinders the further improvement of their performance.

Some researchers inserted the convolution techniques into the transformer-based networks to extract the local spatial contextual information and proposed many transformers-based classification methods (Caron et al., 2021; Chen et al., 2021; Chu et al., 2021; Li et al., 2023; Su et al., 2021; Wu et al., 2021; Zhang et al., 2021). For example, Graham et al. (2021) adopted the convolution operations instead of the linear projection to exploit the local spatial contextual information, and devised a transformer-based classification network named LeViT. Heo et al. (2021) proposed a novel method called Robust Vision Transformer (RvT) network for image classification. The RvT consists two new and effective modules, a Position-aware Attention Scaling (PAAS) module to improve the self-attention mechanism by activating attention information with strong location correlation and a Patch-Wise Augmentation (PWA) module to make the training data rich in affinity and diversity. Inspired by the great success of transformer-based networks in natural image classification, many researchers attempted to design transformer-based hyperspectral image classification networks. For example, He et al. (2019) first introduced transformer networks into HSIs classification and presented transformer-based HSIs classification network by directly employing the transformer network. He et al. (2021) established a novel method called Spatial–Spectral Transformer Network (SSTN) for hyperspectral image classification, which was proposed to separately extract spatial information along the spectral dimension, respectively.

However, all these transformer-based networks have a limitation in capturing local spatial information because the receptive fields of convolution layers are fixed. Furthermore, they fail to fully use the hierarchical representations. To address these challenges, this paper proposes a new and novel transformer-based method called Residual Densely Transformer Network (RDTN) for hyperspectral image classification. More specifically, RDTN is composed of three key modules for addressing these challenges: a Cross-scale convolution attention projection (CSCA) module to extract abundant local spatial contextural information, a Local Residual Transformer Block (LRTB) module to integrate the hierarchical representations, and a Global Residual Connection (GRC) module to fuse the shallow local spatial representation to deep representation together.

3. Methodology

In this section, we will first give a detailed brief of the proposed RDTN. Secondly, we will introduce the key modules of RDTN, *i.e.* Cross-scale convolution attention projection, local residual transformer block, and global residual connection, respectively.

3.1. Overview of RDTN

As shown in Fig. 1, RDTN consists of four parts: Cross-scale convolution attention projection (CSCA), Local residual transformer block (LRTB), global residual connection, and finally the classification part. Supposing the (X, y) is the pair of the input image and its category. Specifically, we first use a Cross-scale convolution attention projection to extract local spatial features with a large receptive field. Then, the extracted feature R_0 could be calculated by

$$R_0 = F_{CSCA}(X), \tag{1}$$

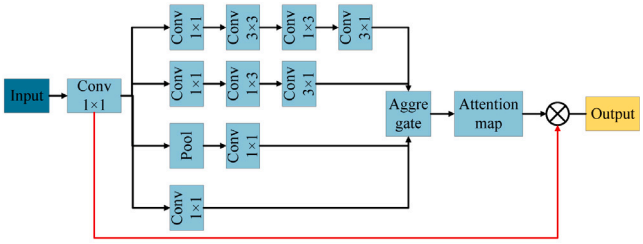


Fig. 2. An illustration of Cross-scale convolution attention projection.

where F_{CSCA} denotes the CSCA module. R_0 is then utilized for further global feature extraction and global residual connection.

Suppose that there are 3 local residual transformer blocks (LRTB) in the proposed RDTN. As a result, the output R_i of the ith LRTB could be obtained by:

$$R_i = F_{LRTB,i}(R_{i-1}) \tag{2}$$

$$= F_{LRTB,i}(F_{LRTB,i-1}(F_{LRTB,1}(R_0))), \tag{3}$$

where the $F_{LRTB,i}$ denotes the operation of the ith LRTB. LRTB is built by using various scale depth-wise convolution and linear operations. Thus, we can obtain hierarchical representation using these three LRTBs. We further make full use of the representation from all the LRTBs and use a linear operation to process the fusion representation. The final output R_4 can be obtained by

$$R_4 = F_{linear}(R_1, R_2, R_3), \tag{4}$$

where F_{linear} is the linear operation. We then add the R_0 into the R_4 by using a global residual connection.

After extracting local and global representation from the original HSI, we utilize two common operation layers (*e.g.*, an average pooling operation layer and a linear operation layer) to select the features for classifying pixels.

3.2. Cross-scale convolution attention projection

Now, we detailed introduce the proposed Cross-scale convolution attention projection (CSCA).

As shown in Fig. 2, the CSCA can be divided into three components: a convolution layer, a multiscale convolution layer (depth-wise convolution), and an aggregate layer. The first component is a channel convolutional operation layer, which is utilized to adjust the channel of the inputs. The second component is multiscale convolutional operation layer, which consists of four different convolutional layers with different receptive fields. The aggregate layer is designed to feature fusion and generate the multiscale attention map. Through the above components, the proposed CSCA can capture local spatial–spectral information with a large receptive field. Following the step of achieving the long-rang relationship, we could estimate the importance of each pixel and calculate an attention map. According to Fig. 2, the CSCA module can be formulated as:

$$X_0 = Conv(X), \tag{5}$$

$$Attention = Conv_{1\times1}(MS - Conv(X_0)), \tag{6}$$

$$Output = Attention \bigotimes X_0. \tag{7}$$

Where $X \in R^{C\times H\times W}$ denotes the inputs, and $MS - Conv$ denotes the multiscale convolution layer. $Attention \in R^{C\times H\times W}$ denotes the final attention map of the input. It is noted that the value in attention map demonstrates the importance of each pixel. Finally, $\bigotimes$ is the element-wise product.

Let $R_{0,i}$ be the output of the ith multiscale convolution branch. Thus, it can be obtained by

$$R_{0,1} = F_1(X_0), \tag{8}$$

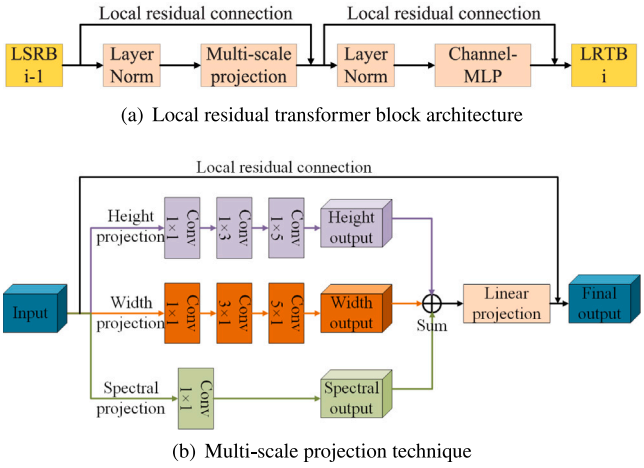


Fig. 3. Local residual transformer block (LRTB) architecture. The (a) is to illustrate of LRTB, and (b) is the multiscale projection that a key component of LRTB.

$$R_{0,2} = F_1(F_{pool}(X_0)), \tag{9}$$

$$R_{0,3} = F_{3,1}(F_{(1,3)}(F_1(X_0))), \tag{10}$$

$$R_{0,4} = F_{3,1}(F_{(1,3)}(F_{(3,3)}(F_1(X_0)))), \tag{11}$$

where F_1 denotes the 1×1 convolution layer, $F_{(1,3)}$ and $F_{(3,1)}$ are the 1×3 and 3×1 convolution layers, respectively. $F_{(3,3)}$ is the 3×3 convolution layers, and F_{pool} denotes the average pooling layer.

After extracting different scale features, we fuse the hierarchical features and re-weight them using a 1×1 convolution layer. The final output R_0 can be represented as

$$R_A = Concat([R_{0,1}, R_{0,2}, R_{0,3}, R_{0,4}]), \tag{12}$$

$$R_0 = F_1(R_A), \tag{13}$$

where the F_1 denotes the 1×1 convolution layer. The final output feature R_0 is the attention map. Thus, the proposed CSCA can not only capture the local spatial contextual information with the large receptive field, but also explore the global dependence of the spectral dimension. Furthermore, CSCA can achieve the adaptability in the spatial dimension and the spectral dimension, respectively.

3.3. Local residual transformer block

As shown in Fig. 3, the LRTB is divided into two stages: Multi-scale projection for extracting local spatial–spectral information and Channel-MLP for extracting the channel information. More specifically, the LRTB first adopt various scale depth-wise convolution layers to separately encode the representation along the height, width in spatial dimension and the spectral dimension, respectively. And then a Channel-MLP is utilized to encode the representation along the channel axis. Suppose the input embedding D tokens $R_{i-1} \in R^{H\times W\times M}$, the outputs of ith LRTB can be obtained by

$$R_{i,1} = MSP(LN(R_{i-1})) + R_{i-1}, \tag{14}$$

$$R_{i,2} = Channel - MLP(LN(R_{i,1})) + R_{i,1}, \tag{15}$$

where the LN is the LayerNorm, $R_{i,1}$ and $R_{i,2}$ denote the outputs of the Multi-scale projection and Channel-MLP, respectively. The MSP denotes the multiscale projection operation. In this paper, the Channel-MLP operation is built by using two fully connected layers followed by a Gaussian Error Linear Unit (GELU) (Hendrycks & Gimpel, 2016) activation function.

We further report the Multi-scale projection (MSP) in (b) of Fig. 3. It is clear that the MSP is a wider module building with multiscale depth-wise convolution operation. With the special design of MSP, the input representations will be separately encoded from the width, height in spatial dimension and spectral dimension. Supposing input representation $R \in R^{H\times W\times M}$, MSP encodes the representation R into three branches: height, and width dimensions of spatial, and spectral dimension. More specifically, we adopt multiscale depth-wise convolution layers to extract more abundant local spatial contextual information. We can achieve the height feature R_H, width feature R_W, and spectral feature R_S by using the MSP to project the inputs. Then, we fuse these three generated features using a hybrid module, including the element-wise addition operation and the fully connected operation. The output R_{out} can be obtained by

$$R_H = W_{1\times5}(W_{1\times3}(W_{(1\times1)}(R))), \tag{16}$$

$$R_W = W_{3\times1}(W_{3\times1}(W_{(1\times1)}(R))), \tag{17}$$

$$R_S = F_{1\times1}R, \tag{18}$$

$$R_{out} = F(R_H + R_W + R_S), \tag{19}$$

where the $W_{1\times5}$, $W_{1\times3}$, $W_{3\times1}$, $W_{5\times1}$, and $W_{1\times1}$ are the depth-wise convolution layer with different size filters. The $F_{1\times1}$ is represented as the point-wise convolution layer, and $F(\cdot)$ is the fully connected operation.

3.4. Hierarchical feature fusion

After encoding the local spatial–spectral representation with three LRTBs, we further a simple local fusion to exploit hierarchical features and a global residual connection operation to retrieve the local spatial information.

Hierarchical feature fusion aims to fusing all the features from all the LRTBs

$$R_{HF} = F_{LN}([R_1, R_2, R_3]), \tag{20}$$

where the $[R_1, R_2, R_3]$ is the concatenation of features produced by three LRTBs. F_{LN} denotes a linear layer.

3.5. Global residual connection

Residual connections allow input data to bypass some layers, adding directly to layer outputs. This enables models to learn differences between inputs and desired outputs, aiding training and performance. Inspired by this, we propose a Global Residual Connection (GRC) to enhance features adding the cross-layer connections. Specifically, we add the output R_{HF} of transformer-based layers to the output R_0 of our proposed Cross-layer Semantic Calibration Approach (CSCA), improving feature representations. Thus, GRC helps the proposed RDTN to learn residual information. It can be formulated as:

$$R_{output} = R_{HF} + R_0, \tag{21}$$

where R_0 is the shallow local spatial information.

4. Experiment results

Now, we will evaluate the performance of RDTN. We will first give a brief of the public HSI datasets and the evaluation metrics. Secondly, we will make a detailed introduction of the comparison methods and implementation setting. Thirdly, we conduct some ablation experiments to study the proposed methods. Finally, we report the experimental results of all the methods.

4.1. Datasets and metrics

In order to assess the performance of the proposed RDTN, we adopt four public benchmarks datasets. The detailed description of four datasets is reported as follows.

Indian Pines Dataset: It is a widely used set of hyperspectral image dataset, whose size is 145 × 145 × 220. It is noted that there are 200 bands in spectral dimension after removing the 20 noisy bands. Furthermore, it comprises 16 land cover categories, including Alfalfa, Corn, Oats, Wheat, etc.

Pavia University Dataset: This dataset is collected over Pavia University, Pavia, Italy with the Reflective Optics System Imaging Spectrometer (ROSIS) sensor. The size of Pavia University dataset is 610 × 340 × 103, where 610 and 340 are around in the height and width in the spatial, and 103 bands in the spectral. It consists of 9 categories of land covers, *i.e.*, Trees, Gravel, Meadows, etc.

Houston2013 Dataset: Houston2013 is the third public HSI dataset to evaluate the performance of the proposed RDTN. It is obtained over the University of Houston by using the ITRES CASI-1500 sensor. Its size is 349 × 1905 × 144, where 349 × 1905 pixels are along the spatial and 144 bands are along in spectral. There are totally 15 land covers, including Highway, Road, Tress, etc.

Xiongan Dataset: It is the fourth HSI public dataset, whose size is 3750 × 1580 × 250. It is obtained by using visible and near-infrared imaging spectrometer. 20 land covers are collected in this dataset, including ash, pear, grassland, house, etc.

Metrics: The classification results are evaluated with three widely used metrics, *i.e.*, Average Accuracy (AA), Overall Accuracy (OA) and Kappa Coefficient (κ). The OA refers to the proportion of correctly classified categories to the total number of categories. Its calculation formula is as follows:

$$\begin{aligned} Accuracy &= \frac{TP+TN}{TP+TN+FN+FP}, \\ OA &= (\frac{1}{m}\sum_{k} Accuracy_k). \end{aligned} \tag{22}$$

where TP is the number of positive instances correctly predicted as positive, TN refers the number of negative instances correctly predicted as negative. FP is the number of negative instances incorrectly predicted as positive, representing Type I error. And FN denotes the number of positive instances incorrectly predicted as negative, representing Type II error. The m is the number of categories.

κ is a coefficient used for assessing spatial consistency in image classification, representing the proportion of error reduction in classification compared to purely random classification. Its calculation formula is as follows:

$$\kappa = \frac{N\sum_{i=1}^{n} x_{ii} - \sum_{i=1}^{n}(x_{i+}\times x_{+i})}{N^2 - \sum_{i=1}^{n}(x_{i+}\times x_{+i})}. \tag{23}$$

where N is the total number of pixels, x_{i+} refers the sums of rows in the confusion matrix and x_{i+} is the sums of columns in the confusion matrix. x_{ii} represents the diagonal elements of the confusion matrix.

4.2. Experimental setup

Comparison with state-of-the-art backbone methods: We adopt three representative methods (such as, SVM and KNN) and four state-of-the-art methods (such as 2D-CNN and ViT) for the comparative evaluation, including traditional representative methods (*i.e.*, SVM and KNN), RNNs-based approaches (*i.e.*, Mou), classical CNNs-based approaches (*i.e.*, 2D-CNN, and He), and transformers-based approaches (*i.e.*, ViT, Deep ViT, CaiT, CvT). Comparative methods are reported as follows:

- For SVM, the libsvm toolbox in sklearn is adopted to handle the HSI classification.

- For KNN, the sklearn toolbox is selected for implementation of HSI classification. Since the number of nearest neighbors (K) is important to achieve a better classification result, it is set to 5.
- The Mou is a RNN-based method for HSI classification. It consists of one recurrent layer with 64 units and followed by the gated recurrent unit.
- The 2D-CNN is a state-of-the-art CNN-based method for HSI classification, which consists of three 2D convolution blocks to extract local spatial context information and a softmax layer to produce the class. It is noted that each 2D convolutional block consists of four layers: a 2D convolutional operation layer to extract local features, a BN operation layer to normalize features, a ReLU operation layer to activate the features and an avg-pooling operation layer to pool the features. Specifically, the 2D convolution layer of each 2D convolution block is set to $3 \times 3 \times 64$, $3 \times 3 \times 128$, and $3 \times 3 \times 256$, respectively.
- He is a 3D-CNN-based method for HSI classification, which is built with three 3D convolution blocks, a pooling layer and a softmax. Unlike 2D-CNN, each 3D convolutional block in He has different 3D convolution layers. More Specifically, the first convolution block has one 3D convolution layer with a receptive field $11 \times 3 \times 3 \times 16$, the secondly and thirdly convolution block have four different 3D convolution layers (*i.e.*, $1 \times 1 \times 1 \times 16$, $3 \times 1 \times 1 \times 16$, $5 \times 1 \times 1 \times 16$, and $11 \times 1 \times 1 \times 16$), and the final convolution block has one 3D convolution layer with $3 \times 2 \times 2 \times 16$. All the 3D convolution layers are used to extract the local spatial and spectral information. Then the pooling layer with a Max pool $3 \times 2 \times 2$ is utilized to reduce the dimension of features.
- ViT is the classical transformer network for image classification. We apply it for HSI classification following the ViT architecture setting, only including transformer encoders. In detail, five transformer encoder blocks are utilized in ViT for HSI classification.
- For Deep ViT, its architecture is similar to ViT. Different to ViT, it uses a linear layer to re-calculates the attention maps. We also employ the Deep ViT to HSI classification to further improve the classification results.
- CaiT is a deeper transformer-based approach for image classification. In the experiment, we follow the CaiT network architecture and apply it to recognize HSI pixel.
- CvT is a transformer-based approach for image classification, of which the characteristics of Convolution Neural Networks (CNN) (*i.e.*, shift, and scaling) are introduced into transformer architecture. The CvT takes the advantages of both the ability of exploring the global dependence of transformer networks and the ability of extracting local features of CNNs. In the experiment, we follow the CvT architecture included three stages, and apply it to HSI classification task.
- For RDTN, we first adopt a (CSCA) module to encode the input HSI cub to obtain the spatial–spectral information. Then, we utilize four transformer encoder stages to model the spatial–spectral fusion information. Specifically, hyperspectral image is divided into several patches. And then the patches are projected by using the CSCA to exploit local spatial–spectral fusion representation. The extracted representation is then fed into four cascaded local residual transformer encoder stages for extracting features. Each encoder stage consists of several multiscale depth-wise convolution layers, and an MLP with 256 hidden dimensions, and a GELU nonlinear activation layer (Hendrycks & Gimpel, 2016).

Implementation Details: The proposed RDTN and comparison methods are implemented on the server of the Windows operating system. The server is powered by an Intel Core 7 Duo CPU at 3.40 GHZ with 128 GB RAM, while the GPU processor is NVIDIA RTX A6000 with 24 GB ROM. We train all the methods from scratch using Adam optimizer with the weight decay of 0.0005, and with a batch size of 100. We initialize an equal learning rate for all trainable layers to 1e-2, which is manually decreased by a factor of 10 when the validation error stopped decreasing. Furthermore, we trained all the methods for 100 epochs.

4.3. Results and analysis

The experimental results on four public HSI datasets are listed on Tables 1, 3, 2, and 4, respectively.

In summary, the traditional classifiers, *i.e.*, SVM, and KNN, achieve the similar classification results on the four datasets, except the classification accuracies in terms of AA, OA, and κ with SVM on the Indian Pines and Xiongan datasets (which are far inferior to those using KNN). More specifically, SVM only achieves the results of AA (28.55%), OA (55.20%), κ (45.78%) on the Indian Pines dataset, and it could not recognize some categories, such as classes Alfalfa, Corn, and Grass-pasture. And it could not recognize the classes Acer negundo, elm, and ash in the Xiongan dataset. These results are likely to be related to the imbalanced classes and the complexity of the Xiongan and Indian Pines datasets. On the other hand, it demonstrates that the traditional classifiers could not better fit the complex data.

Interestingly, the RNN-based methods, *i.e.*, Mou, is observed to perform an unsatisfactory performance, such as the performance on Indian Pines (with OA is 61.88%), Pavia University (with OA is 92.12%), and Houston2013 (with OA is 89.46%). This observational finding may help us to understand that only feature extraction of spectral dimension would not perform a better performance in the HSI classification tasks. The most obvious finding to emerge from the analysis is that the deep learning architectures (*i.e.*, CNNs) observably outperform the traditional classifiers, such as OA 97.78% vs 89.23%. These observed results could be attributed to the powerful ability of feature extraction with deep learning and demonstrate the practicality of deep learning-based methods in HSI classification.

Another important finding is that the 3D-CNN method performs better than other CNNs (such as 2D-CNN), especially on the Houston2013 dataset. This is mainly because the 3D-CNN could explore the spatial–spectral fusion features, which also demonstrates that the spatial–spectral fusion information would improve the classification results. We can also find that the transformer-based methods (such as ViT, DeepViT) perform better than the classical deep learning methods (such as CNNs and RNNs), which attributes to the finer extraction of spectral representations. Finally, adding the convolution operation to exploit local spectral–spatial representation, transformer-based approaches, *i.e.*, Cait, CvT, could capture the local and global dependence and further improve the classification performance.

These transformer-based approaches could not only capture the global spatial–spectral dependence, but also explore the local spectral-spatial fusion features. However, they may have a failure in making full use of the hierarchical representations and capturing local spatial–spectral discrepancies. To overcome these challenges, we propose RDTN to capture abundant multiscale local spatial–spectral features using the CSCA module and fully utilize the hierarchical representations using the LRTB and GRC, resulting in improving the classification performance. In detail, RDTN performs better than other comparison methods (such as classical approaches and the state-of-the-art approaches), for instance, AA 95.46% vs 94.59% vs 94.72% on the Indian Pines dataset. This demonstrates that the joint consideration of the multiscale local spatial contextual information and the full use of the hierarchical representations would further improve the performance.

As shown in Figs. 4–7, we evaluate the proposed method in quantitative by visualizing classification maps with all methods. Roughly, we can observe that the traditional classifiers, *e.g.*, SVM and KNN, tend to produce salt and pepper noises in classification maps, especially of the Indian Pines and Xiongan datasets. This demonstrates that these traditional classifiers fail to fit the complex dataset. It is not surprising

Table 1
The Classification results of the Indian Pines dataset (70% percentage training sample).

Class no.	Traditional classifiers		RNNs	CNNs-based methods		Transformers-based methods				
	SVM	KNN	Mou	2D-CNN	3D-CNN	ViT	Deep ViT	CaiT	CvT	RDTN
1	0.00	66.62	84.73	100	100	93.64	98.52	96.53	100	100
2	4.00	66.25	45.09	99.29	97.65	96.98	98.80	95.67	99.18	99.77
3	0.00	62.36	49.99	85.08	85.97	82.9	85.22	81.55	85.81	93.19
4	0.00	49.54	23.99	93.48	92.54	89.73	92.1	91.06	93.15	97.87
5	25.00	89.08	61.23	89.21	90.57	86.33	88.21	87.57	88.69	95.68
6	8.00	90.95	84.73	100	100	98.38	99.75	98.61	99.95	99.32
7	0.00	91.54	59.99	100	100	94.91	98.89	92.27	98.42	100
8	90.80	96.84	96.65	94.54	92.94	94.66	94.31	94.04	94.80	100
9	0.00	60.73	39.42	100	100	89.52	93.76	91.03	100	100
10	1.60	73.80	36.96	95.05	94.10	92.92	94.7	92.44	94.47	97.37
11	58.70	77.90	45.51	96.69	96.24	94.90	96.34	94.08	96.77	97.02
12	0.00	61.08	41.60	94.68	92.63	92.30	93.94	91.70	94.31	96.81
13	84.24	91.32	97.61	100	100	99.36	99.84	98.87	100	100
14	83.80	93.36	90.31	98.49	98.00	97.82	98.31	97.53	98.29	99.87
15	9.80	53.58	65.64	71.00	67.05	67.42	71.27	68.10	70.45	83.74
16	90.90	91.87	82.81	98.04	96.43	97.36	97.46	98.36	99.10	100
AA (%)	28.55	76.05	62.89	94.72	94.01	91.82	93.84	91.84	94.59	**95.46**
OA (%)	55.20	77.25	61.88	90.96	90.49	89.24	90.67	88.84	90.86	**97.54**
κ (%)	45.78	73.95	56.88	89.80	89.25	87.84	89.46	87.39	89.69	**94.85**

Table 2
The Classification results of the Houston2013 dataset (70% percentage training sample).

Class no.	Traditional classifiers		RNNs	CNNs-based methods		Transformers-based methods				
	SVM	KNN	Mou	2D-CNN	3D-CNN	ViT	Deep ViT	CaiT	CvT	RDTN
1	92.43	98.88	96.94	96.20	99.60	99.45	99.73	98.73	99.71	100
2	94.19	98.90	98.01	98.74	99.33	99.17	99.43	98.92	99.51	100
3	99.06	99.62	100	99.84	100	99.95	100	99.83	100	100
4	95.88	99.53	98.42	96.89	100	99.67	99.77	99.04	99.93	100
5	94.76	98.19	99.17	98.49	98.51	98.49	98.69	97.89	98.53	100
6	90.71	99.33	99.64	94.09	98.98	98.21	99.85	98.86	100	100
7	74.45	95.04	82.17	94.93	98.94	98.62	99.34	98.75	99.42	100
8	70.56	95.15	81.50	92.11	99.19	98.61	99.12	97.81	99.14	99.73
9	70.07	88.06	79.28	92.96	98.53	98.93	99.22	97.51	99.09	99.60
10	65.81	91.05	83.35	91.47	99.59	99.47	99.86	97.46	99.96	99.86
11	66.60	90.37	84.66	95.05	98.09	98.60	98.85	97.38	98.68	99.47
12	61.26	89.06	77.10	93.87	99.73	99.20	99.85	97.39	99.97	99.46
13	12.69	64.09	75.50	95.68	99.29	98.39	99.32	94.33	99.89	98.93
14	86.85	98.69	99.09	99.28	100	99.69	99.96	99.54	100	100
15	99.04	99.47	99.32	98.79	100	99.87	100	99.40	100	100
AA (%)	78.29	93.70	90.28	95.89	99.32	99.09	99.53	98.19	99.59	**99.85**
OA (%)	79.78	94.35	89.46	95.16	98.85	98.65	99.07	97.72	99.02	**99.83**
κ (%)	78.11	93.89	88.60	94.77	98.76	98.54	99.00	97.53	98.94	**99.83**

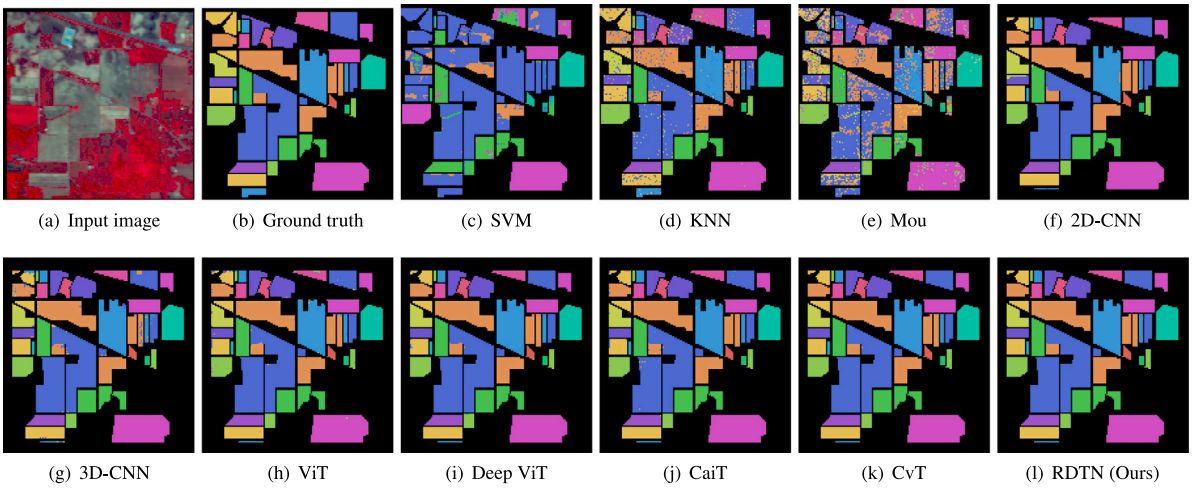
Table 3
Classification results of the Pavia University dataset (70% percentage training sample).

Class no.	Traditional classifiers		RNNs	CNNs-based methods		Transformers-based methods				
	SVM	KNN	Mou	2D-CNN	3D-CNN	ViT	Deep ViT	CaiT	CvT	RDTN
1	88.54	92.33	91.73	96.69	96.59	96.46	96.49	94.92	96.52	100
2	94.24	94.33	96.32	92.74	92.61	92.64	92.75	91.79	92.70	100
3	65.72	77.04	74.82	94.33	94.02	93.82	94.76	88.70	94.35	100
4	93.98	92.80	94.71	97.88	97.58	97.54	97.76	97.14	97.92	99.95
5	99.52	99.47	99.91	100	99.98	99.98	99.99	99.90	99.99	100
6	77.19	79.62	89.35	100	99.96	99.85	99.97	96.00	100	100
7	53.96	82.80	69.65	100	99.65	99.62	99.93	95.48	100	100
8	84.32	84.73	83.65	100	99.78	99.50	99.91	97.30	99.99	100
9	100	99.98	99.81	100	99.86	99.95	99.96	99.89	99.96	99.82
AA (%)	84.16	89.23	88.88	97.96	97.78	97.71	97.95	95.68	97.94	**99.99**
OA (%)	88.96	90.58	92.12	92.29	92.09	92.07	92.27	90.60	92.21	**99.97**
κ (%)	85.02	87.36	89.55	90.14	89.90	89.87	90.12	87.97	90.05	**99.99**

that the classical deep learning-based methods, *e.g.*, CNNs (such as 2D-CNN and 3D-CNN) produce relatively smooth classification maps, since they have powerful nonlinear data fitting ability. What is interesting about the data in these classification maps is that the RNNs (*i.e.*, Mou) also generate salt and pepper noises in classification maps. This result may be explained by the fact that the RNNs discard the ability of capturing the local spatial contextual information. The transformer-based approaches are capable of extract highly global dependence from HSIs, resulting in the comparable visualized results to the classical CNNs-based methods. By fully utilizing the hierarchical representations and enhancing multiscale spatial–spectral information, RDTN obtains highly desirable classification maps.

Table 4
The Classification results of the Xiongan dataset (70% percentage training sample).

Class no.	Traditional classifiers		RNNs	CNNs-based methods		Transformers-based methods				
	SVM	KNN	Mou	2D-CNN	3D-CNN	ViT	Deep ViT	CaiT	CvT	RDTN
1	0.00	74.69	82.40	97.09	92.21	98.73	98.08	47.31	99.08	99.12
2	6.00	73.90	89.55	99.11	96.75	99.80	99.40	57.00	99.96	99.98
3	0.00	76.98	84.45	97.23	97.03	99.71	99.09	52.93	99.91	99.89
4	90.00	97.91	98.91	99.36	99.25	99.49	99.46	94.41	99.51	99.51
5	13.00	69.87	84.93	98.77	95.50	99.51	98.87	54.54	99.92	99.96
6	0.00	72.26	90.70	99.06	95.51	99.26	98.80	48.96	99.70	99.71
7	0.00	70.90	94.59	99.73	99.70	99.95	99.83	26.62	99.99	100
8	83.00	94.97	97.51	99.47	99.35	99.82	99.70	89.09	99.85	99.86
9	81.00	96.33	97.99	99.60	99.62	99.95	99.82	93.14	99.97	100
10	83.00	95.69	98.37	99.88	99.70	99.98	99.94	89.34	99.98	100
11	0.00	6.64	35.20	91.95	76.07	97.77	95.57	0.00	99.46	99.91
12	0.00	59.96	73.31	95.20	89.54	98.97	97.91	27.13	99.53	99.75
13	59.00	77.64	88.37	98.31	95.21	99.32	98.83	74.21	99.73	99.77
14	0.00	56.91	53.18	91.69	85.27	98.60	97.78	4.54	99.56	98.41
15	0.00	62.78	76.94	94.87	87.74	98.88	98.35	27.78	98.94	99.47
16	0.00	36.77	47.00	84.23	68.96	97.32	95.80	21.69	99.35	97.23
17	0.00	2.51	17.20	53.26	39.98	91.94	90.10	0.07	98.92	94.10
18	49.00	74.07	83.85	97.81	93.87	99.25	98.66	62.46	99.49	99.64
19	0.00	56.79	73.29	97.80	91.73	99.24	98.58	54.14	99.90	99.89
20	51.00	90.55	92.28	96.75	96.92	99.01	98.58	83.57	99.19	99.34
AA (%)	25.75	67.41	78.00	94.56	90.00	98.83	98.16	50.45	**99.60**	99.28
OA (%)	53.00	78.82	88.40	98.08	95.23	99.15	98.70	68.31	99.44	**99.50**
κ (%)	41.00	75.32	86.52	97.77	94.46	99.01	98.50	62.90	99.36	**99.42**



(a) Input image (b) Ground truth (c) SVM (d) KNN (e) Mou (f) 2D-CNN

(g) 3D-CNN (h) ViT (i) Deep ViT (j) CaiT (k) CvT (l) RDTN (Ours)

Fig. 4. Classification maps obtained using different methods on the Indian Pines dataset.

Table 5
Complexity Analysis of the proposed RDTN and transformer-based comparison methods on IndianPines dataset.

Methods	F(G)	P (MB)	Training (s)	Testing (s)	TP	OA (%)
ViT	0.68	13.20	727.89	3.43	245	89.24
DeepViT	13.69	57.75	1497.97	6.82	47	90.67
CaiT	27.01	119.93	4808.70	20.74	24	88.84
CvT	9.08	17.83	2758.06	11.65	33	90.86
RDTN	7.45	56.93	9993.74	47.04	30	**97.54**

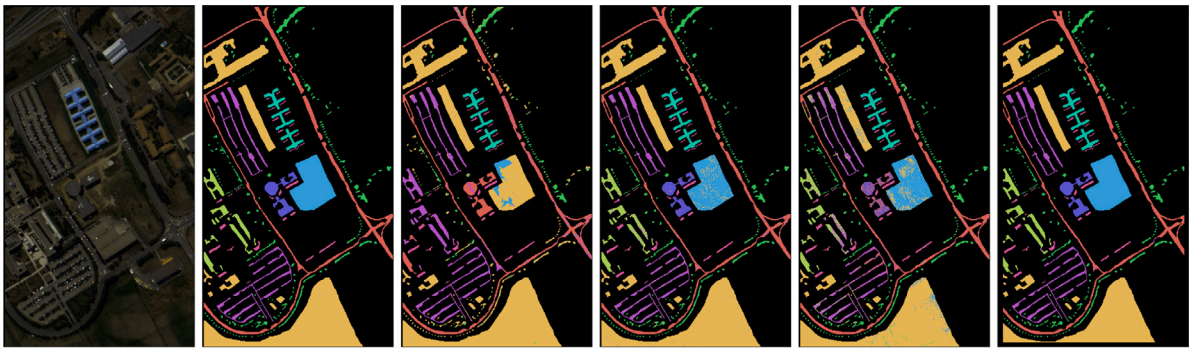
4.4. Complexity analysis

We analyze the complexity of the proposed RDTN and the transformer-based comparison methods on IndianPines dataset in terms of FLOPS, Parameters, Training and Testing times, and Throughput/second. The results are reported in Table 5. It is noted that "F", "P", and "TP" are short for "FLOPS", "Parameters", and "Throughput/second", respectively.

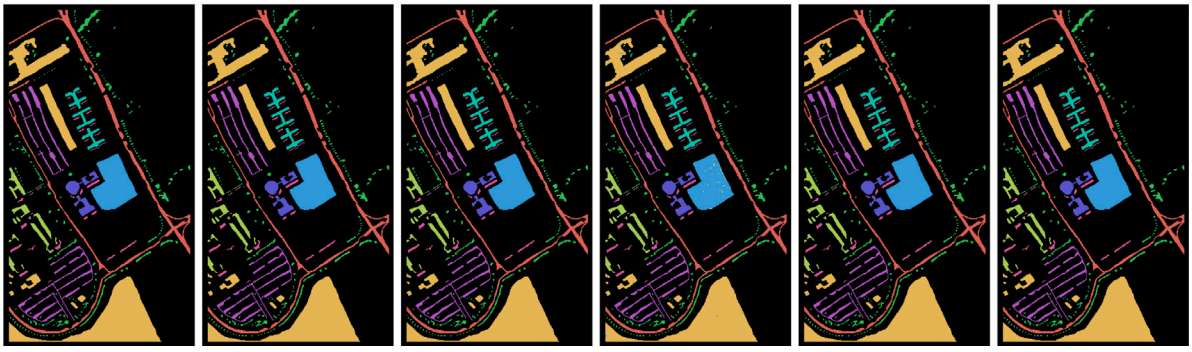
According to Table 5, we can find that the proposed RDTN achieves the best result in terms of OA, but requires significant time for training and testing. It is noted that the proposed RDTN has many more parameters, mainly due to the use of multiscale convolution operations. This demonstrates that joining multiscale convolution operations could improve performance, but also increases the parameters and requires much more time to train. This is a limitation of the proposed RDTN, and our future work will design a lightweight transformer-based method for hyperspectral image classification. Secondly, we can see that only the classic transformer (*e.g.*, ViT) has few parameters and can process many images (such as 245 images per second). However, it yields poorer classification results. Finally, we can find that ViT variants improve performance by integrating convolution operations (*e.g.*, CaiT, and CvT) or re-calculating the attention map (*e.g.*, Deep ViT), but they also increase the number of parameters, requiring more time and resources to train the model.

4.5. Ablation studies

(1) Ablation study of T and M: In the study of the melting, we studied the parameters of the RDTN: the number of LRTB (denote as T for short), and the max size of depth-wise convolution layer of LRTB (denote as M for short). In this study, the performance of ViT

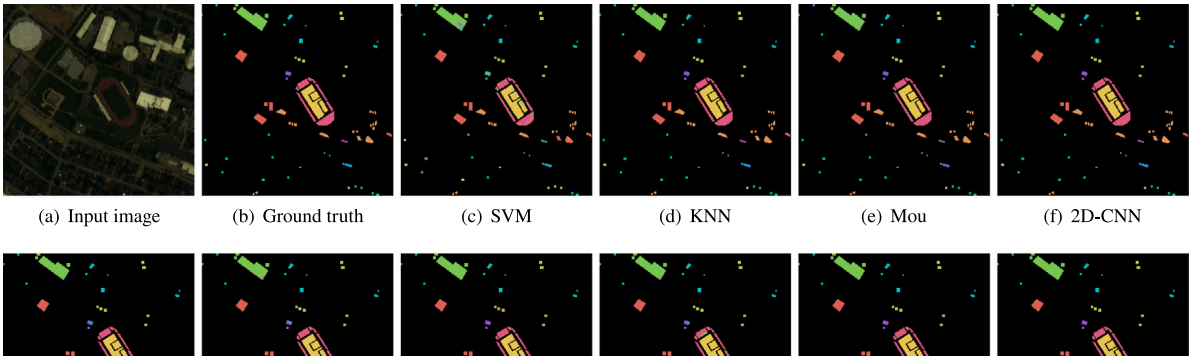


(a) Input image (b) Ground truth (c) SVM (d) KNN (e) Mou (f) 2D-CNN



(g) 3D-CNN (h) ViT (i) Deep ViT (j) CaiT (k) CvT (l) RDTN (Ours)

Fig. 5. Classification maps obtained by different methods on the Pavia University dataset.



(a) Input image (b) Ground truth (c) SVM (d) KNN (e) Mou (f) 2D-CNN

(g) 3D-CNN (h) ViT (i) Deep ViT (j) CaiT (k) CvT (l) RDTN (Ours)

Fig. 6. Classification maps obtained by different methods on some area of the Houston2013 dataset.

is selected as a reference. Fig. 8 presents the convergence analysis of RDTN with different values of T and M. From Fig. 8(a), we can see that larger T leads to higher performance. This is mainly because that RDTN network becomes deeper with larger T. From Fig. 8(b), we can observe that larger M produces a better performance. A possible explanation for this might be that the larger size of the kernel is, the more abundant local feature is retrieved of the deep learning-based HSI classification method. In addition, RDTN with smaller T or M may suffer from performance degradation, but it still outperforms ViT. Moreover, RDTN allows a deeper and wider transformer network, from which more hierarchical representations are encoded for achieving higher performance.

(2) Ablation study of RDTN: We conduct an ablation investigation on the effects of different modules, such as Cross-scale convolution attention projection (CSCA), Local residual transformer block (LRTB), and global residual connection (GRC), and the results are presented in Table 6. In this study, the ViT is chosen as the baseline, which is a classical transformer without using CSCA, LRTB, and GRC. We can find that it performs a very poor result (OA = 98.65%). This is caused by the

Table 6
Ablation investigation of Cross-Scale Convolution attention projection (CSCA), local residual transformer block (LRTB), and global residual connection (GRC).

Modules	Different combinations of CSCA, LRTB, and GRC							
CSCA	×	√	×	×	√	√	×	√
LRTB	×	×	√	×	√	×	√	√
GRC	×	×	×	√	×	√	√	√
OA	98.65	99.05	98.96	99.11	99.14	99.25	99.16	**99.59**

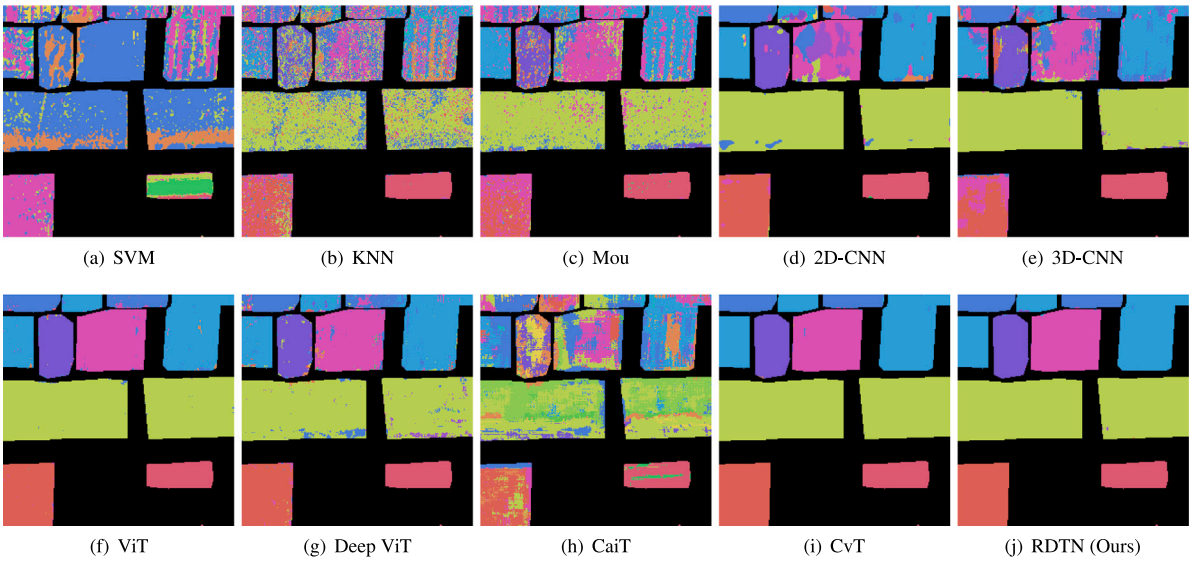
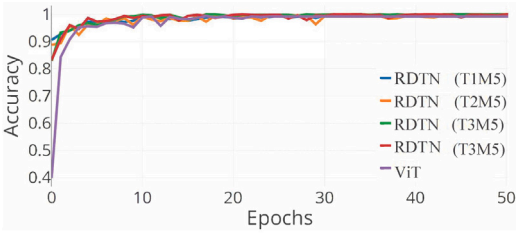
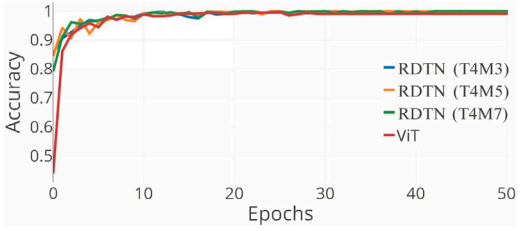


Fig. 7. Classification maps obtained by different methods on some area of the Xiongan dataset.



(a) Investigation of T



(b) Investigation of M

Fig. 8. Convergence analysis of RDTN with different values of T, and M. The curves for each combination are obtained on Houston2013 dataset with 50 epochs.

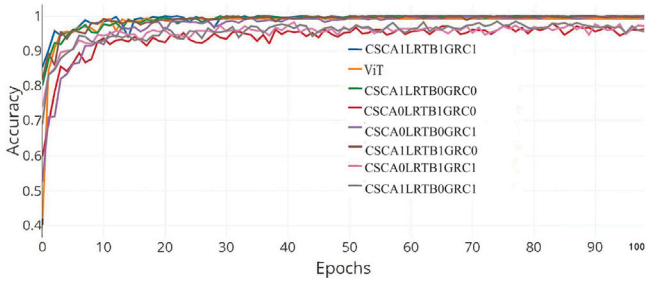


Fig. 9. Convergence analysis on CSCA, LRTB, and GRC. The curves for each combination are obtained on Houston2013 dataset in 100 epochs.

missing consideration of spectral dimension and without the local spatial features extraction, resulting in unsatisfactory performance. It also demonstrates that directly applying the classical transformers (*i.e.*, ViT) to HSI classification tasks would not perform the good performance.

We choose one module from the three modules, such as CSCA, LRTB, and GRC, and then add it to ViT. As a result, we could achieve three different results, which are remarked as CSCA1LRTB0GRC0, CSCA0LRTB1GRC0, and CSCA0LRTB0GRC1, respectively. It is noted that these three results are reported in Table 6 from 2nd to 4th. According to the results, we can see that all the modules can efficiently improve the performance, surpassing 0.3% than the ViT (with OA 98.96% vs 98.65). This is mainly because each module is capable of extracting the local spatial contextual and global information, and further enhancing its ability to capture global dependence.

We further randomly adopt two modules from the proposed three modules and add them to ViT, resulting in three results CSCA1LRTB 1GRC0, CSCA0LRTB1GRC1, and CSCA1LRTB0GRC1. The results are listed in Table 6 (from 5th to 7th). From the results, we can observe that the methods with two modules outperform those only with one module. This result may be explained by the fact that two modules would extract much more local feature. We finally add all three modules to ViT, and obtain the result remarked as RDTN_CSCA1LRTB1GRC1. It shows that RDTN simultaneously using three modules performs the best performance.

We also report the visualization of the convergence process of these eight methods in Fig. 9. It is noted that the studies are conducted on the Houston2013 dataset. We can observe convergence curves that fit the above analysis. These findings may help us to understand that CSCA, LRTB, and GRC can further stabilize the training process without obvious performance degradation. These quantitative and qualitative analyses demonstrate the superiority of the proposed RDTN and the effectiveness of the proposed CSCA, LRTB, and GRC.

(3) Ablation study of the percentage of training samples: In this ablation study, we investigate the effects of training samples. We conduct extensive experiments on different percentage of training samples. It is noted that the training samples are varied from 10% to 70% at intervals of 10% on three public benchmark HSI datasets, but 1% to 7% at intervals of 1% on the Xiongan dataset.

As shown in Fig. 10, with the increase of the percentage of training samples, the classification accuracy is gradually improved. On the other hand, RDTN achieves poorly with 10% training data, indicating that the RDTN requires substantially more training examples. Additionally, the OAs tend to be stable when the training samples varies between 40% and 60%, demonstrating the superiority of proposed RDTN.

(4) Ablation study of the percentage of training samples: In this ablation study, we investigate the effects of training samples. We conduct extensive experiments on different percentage of training

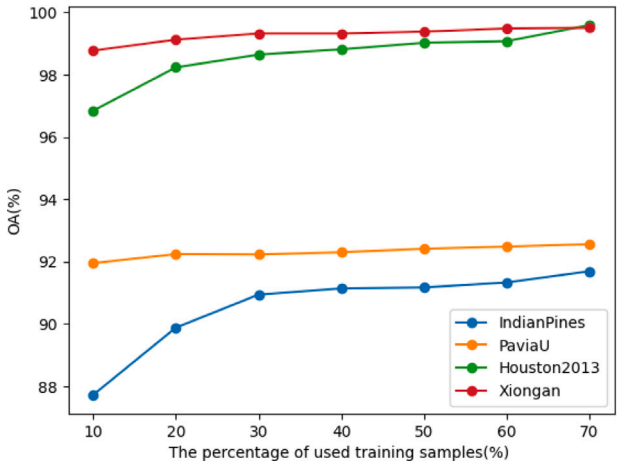


Fig. 10. Classification results (OA) achieved by RDTN using different numbers of training samples on four benchmarks datasets.

samples. It is noted that the training samples are varied from 10% to 70% at intervals of 10% on three public benchmark HSI datasets, but 1% to 7% at intervals of 1% on the Xiongan dataset.

5. Conclusion

In this paper, we presented a novel transformer-based method called RDTN to fully utilize the hierarchical representations and model the local spatial–spectral differences. The proposed RDTN is composed of three key modules: a Cross-scale Convolutional Attention (CSCA) module, a Local Residual Transformer Block (LRTB) module, and a Global Residual Connection (GRC) module. The CSCA is used to extract multiscale local spatial data by using multiscale convolutional layers. The LRTB is built with multiscale depth-wise convolution layers. It is not only utilized to collect multiscale local spatial–spectral characteristics, but also fully exploit the hierarchical representations. Additionally, the GRC is used to integrate the local spatial contextual information and the global representations. So that, the proposed RDTN would achieve dense spatial–spectral fusion features and deep supervision. Finally, we conduct extensive experiments on four public benchmark datasets: Indian Pines, Pavia University, Houston2013, and Xiongan datasets. The experimental results demonstrate the superiority of the proposed RDTN.

Future study will focus on improve transformer architecture, such as transfer learning, and mutual learning with various networks (CNNs and Transformers). Then a standardized and universal method will be established for HSI classification based on transformers.

Funding

This work was jointly supported by the following projects: Project of Shenzhen Polytechnic under Grant No. 6022310002K. National Natural Science Foundation of China (NSFC) Fund under Grant 62301174. Guangzhou basic and applied basic research topics under Grant 2024A 04J2081.

CRediT authorship contribution statement

Yan Li: Methodology, Conceptualization, Investigation, Writing – review & editing. **Xiaofei Yang:** Methodology, Conceptualization, Investigation, Writing – review & editing. **Dong Tang:** Investigation, Writing – review & editing. **Zheng Zhou:** Investigation, Writing – review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Acknowledgments

The authors would like to thank the reviewers for their insightful comments and useful suggestions.

References

Boulch, A., Audebert, N., & Dubucq, D. (2017). Autoencodeurs pour la visualisation d'images hyperspectrales. In *XXV colloque gretsi: juan-les-pins, France*.

Caron, M., Touvron, H., Misra, I., J'egou, H., Mairal, J., Bojanowski, P., & Joulin, A. (2021). Emerging properties in self-supervised vision transformers. In *2021 IEEE/CVF international conference on computer vision* (pp. 9630–9640).

Chen, C.-F., Fan, Q., & Panda, R. (2021). CrossViT: Cross-attention multi-scale vision transformer for image classification. In *2021 IEEE/CVF international conference on computer vision* (pp. 347–356).

Chen, Y., Jiang, H., Li, C., Jia, X., & Ghamisi, P. (2016). Deep feature extraction and classification of hyperspectral images based on convolutional neural networks. *IEEE Transactions on Geoscience and Remote Sensing*, *54*(10), 6232–6251.

Chen, Y., Lin, Z., Zhao, X., Wang, G., & Gu, Y. (2014). Deep learning-based classification of hyperspectral data. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, *7*(6), 2094–2107.

Chu, X., Tian, Z., Wang, Y., Zhang, B., Ren, H., Wei, X., Xia, H., & Shen, C. (2021). Twins: Revisiting the design of spatial attention in vision transformers. In *Neural information processing systems*.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., & Houlsby, N. (2020). An image is worth 16x16 words: Transformers for image recognition at scale. arXiv:2010.11929.

Dupont, M. F., Elbourne, A., Cozzolino, D., Chapman, J., Truong, V. K., Crawford, R. J., & Latham, K. (2020). Chemometrics for environmental monitoring: A review. *Analytical Methods*, *12*(38), 4597–4620.

Fan, J., Chen, T., & Lu, S. (2018). Superpixel guided deep-sparse-representation learning for hyperspectral image classification. *IEEE Transactions on Circuits and Systems for Video Technology*, *28*(11), 3163–3173.

Glorot, X., Bordes, A., & Bengio, Y. (2011). Deep sparse rectifier neural networks. In *Aistats*: *vol. 15*, (no. 106), (p. 275).

Graham, B., El-Nouby, A., Touvron, H., Stock, P., Joulin, A., J'egou, H., & Douze, M. (2021). LeViT: A vision transformer in ConvNet's clothing for faster inference. In *2021 IEEE/CVF international conference on computer vision* (pp. 12239–12249).

Hamida, A. B., Benoit, A., Lambert, P., & Amar, C. B. (2018). 3-D deep learning approach for remote sensing image classification. *IEEE Transactions on Geoscience and Remote Sensing*.

Hang, R., Liu, Q., Hong, D., & Ghamisi, P. (2019). Cascaded recurrent neural networks for hyperspectral image classification. *IEEE Transactions on Geoscience and Remote Sensing*, *57*(8), 5384–5394.

He, X., Chen, Y., & Lin, Z. (2021). Spatial-spectral transformer for hyperspectral image classification. *Remote Sensing*, *13*(3), 498.

He, M., Li, B., & Chen, H. (2017). Multi-scale 3D deep convolutional neural network for hyperspectral image classification. In *2017 IEEE international conference on image processing* (pp. 3904–3908). IEEE.

He, J., Zhao, L., Yang, H., Zhang, M., & Li, W. (2019). HSI-BERT: Hyperspectral image classification using the bidirectional encoder representation from transformers. *IEEE Transactions on Geoscience and Remote Sensing*, *58*(1), 165–178.

Hendrycks, D., & Gimpel, K. (2016). Gaussian error linear units (gelus). arXiv preprint arXiv:1606.08415.

Heo, B., Yun, S., Han, D., Chun, S., Choe, J., & Oh, S. J. (2021). Rethinking spatial dimensions of vision transformers. In *2021 IEEE/CVF international conference on computer vision* (pp. 11916–11925).

Hou, Z., Li, W., Tao, R., & Du, Q. (2021). Three-order tucker decomposition and reconstruction detector for unsupervised hyperspectral change detection. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, *14*, 6194–6205.

Hu, K., Chen, Z., Kang, H., & Tang, Y. (2024). 3D vision technologies for a self-developed structural external crack damage recognition robot. *Automation in Construction*.

Hughes, G. (1968). On the mean accuracy of statistical pattern recognizers. In *IEEE trans. inf. theory 1968* (pp. 55–63).

Ioffe, S., & Szegedy, C. (2015). Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International conference on machine learning* (pp. 448–456). PMLR.

Kuo, B.-C., & Landgrebe, D. A. (2004). Nonparametric weighted feature extraction for classification. *IEEE Transactions on Geoscience and Remote Sensing*, *42*(5), 1096–1105.

Lee, H., & Kwon, H. (2016). Contextual deep CNN based hyperspectral classification. In *IGARSS, 2016 IEEE international* (pp. 3322–3325). IEEE.

Li, Z., Guo, F., Li, Q., Ren, G., & Wang, L. (2020). An encoder–decoder convolution network with fine-grained spatial information for hyperspectral images classification. *IEEE Access*, *8*, 33600–33608.

Li, S., Song, W., Fang, L., Chen, Y., Ghamisi, P., & Benediktsson, J. A. (2019). Deep learning for hyperspectral image classification: An overview. *IEEE Transactions on Geoscience and Remote Sensing*, *57*(9), 6690–6709.

Li, Y., Zhang, K., Cao, J., Timofte, R., Magno, M., Benini, L., & Goo, L. V. (2023). LocalViT: Analyzing locality in vision transformers. In *2023 IEEE/RSJ international conference on intelligent robots and systems* (pp. 9598–9605).

Li, Y., Zhang, H., & Shen, Q. (2017). Spectral–spatial classification of hyperspectral imagery with 3D convolutional neural network. *Remote Sensing*, *9*(1), 67.

Liu, P., Zhang, H., & Eom, K. B. (2017). Active deep learning for classification of hyperspectral images. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, *10*(2), 712–724.

Lorenzo, P. R., Tulczyjew, L., Marcinkiewicz, M., & Nalepa, J. (2020). Hyperspectral band selection using attention-based convolutional neural networks. *IEEE Access*, *8*, 42384–42403.

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

Mou, L., Ghamisi, P., & Zhu, X. X. (2017). Deep recurrent neural networks for hyperspectral image classification. *IEEE Transactions on Geoscience and Remote Sensing*, *55*(7), 3639–3655.

Navin, M. S., & Agilandeeswari, L. (2020). Multispectral and hyperspectral images based land use/land cover change prediction analysis: An extensive review. *Multimedia Tools and Applications*, *79*(39), 29751–29774.

Ran, L., Zhang, Y., Wei, W., & Yang, T. (2016). Bands sensitive convolutional network for hyperspectral image classification. In *Proceedings of the international conference on internet multimedia computing and service* (pp. 268–272). ACM.

Scholkopf, B., & Smola, A. J. (2001). *Learning with kernels: Support vector machines, regularization, optimization, and beyond*. MIT Press.

Schuster, M., & Paliwal, K. K. (1997). Bidirectional recurrent neural networks. *IEEE Transactions on Signal Processing*, *45*(11), 2673–2681.

Su, J., Lu, Y., Pan, S., Wen, B., & Liu, Y. (2021). Roformer: Enhanced transformer with rotary position embedding. arXiv:2104.09864.

Tang, Y., Qi, S., Zhu, L., Zhuo, X., Zhang, Y., & Meng, F. (2024). Obstacle avoidance motion in mobile robotics. *Journal of System Simulation*, *36*(1), 1–26.

Touvron, H., Cord, M., Sablayrolles, A., Synnaeve, G., & J'egou, H. (2021). Going deeper with image transformers. In *2021 IEEE/CVF international conference on computer vision* (pp. 32–42).

Vaswani, A., Shazeer, N. M., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., & Polosukhin, I. (2017). Attention is all you need. In *Neural information processing systems*.

Villa, A., Benediktsson, J. A., Chanussot, J., & Jutten, C. (2011). Hyperspectral image classification with independent component discriminant analysis. *IEEE Transactions on Geoscience and Remote Sensing*, *49*(12), 4865–4876.

Wold, S., Esbensen, K., & Geladi, P. (1987). Principal component analysis. *Chemometrics and Intelligent Laboratory Systems*, *2*(1–3), 37–52.

Wu, Z., Tang, Y., Hong, B., Liang, B., & Liu, Y. (2023). Enhanced precision in dam crack width measurement: Leveraging advanced lightweight network identification for pixel-level accuracy. *International Journal of Intelligent Systems*, *2023*, 1–16.

Wu, H., Xiao, B., Codella, N. C. F., Liu, M., Dai, X., Yuan, L., & Zhang, L. (2021). Cvt: Introducing convolutions to vision transformers. In *2021 IEEE/CVF international conference on computer vision* (pp. 22–31).

Yan, L. C., Yoshua, B., & Geoffrey, H. (2015). Deep learning. *Nature*, *521*(7553), 436–444.

Yang, X., Ye, Y., Li, X., Lau, R. Y., Zhang, X., & Huang, X. (2018). Hyperspectral image classification with deep learning models. *IEEE Transactions on Geoscience and Remote Sensing*, *56*(9), 5408–5423.

Yang, X., Zhang, X., Ye, Y., Lau, R. Y., Lu, S., Li, X., & Huang, X. (2020). Synergistic 2D/3D convolutional neural network for hyperspectral image classification. *Remote Sensing*, *12*(12), 2033.

Zhang, Z., Zhang, H., Zhao, L., Chen, T., & Pfister, T. (2021). Aggregating nested transformers. arXiv:2105.12723.

Zhou, D., Kang, B., Jin, X., Yang, L., Lian, X., Hou, Q., & Feng, J. (2021). DeepViT: Towards deeper vision transformer. arXiv:2103.11886.