Hyper-LKCNet: Exploring the Utilization of Large Kernel Convolution for Hyperspectral Image Classification

Rong Liu, *Member, IEEE*, Zhilin Li, Jiaqi Yang, Jian Sun, and Quanwei Liu, *Student Member, IEEE*

***Abstract*—Recently, transformers have garnered significant attention due to their exceptional capability to capture long-range dependencies in data. A critical factor contributing to their superior performance is their ability to operate over large receptive fields. As such, a natural question arises as to how to expand the receptive fields in convolutional neural networks to achieve the superior performance comparable with that of transformers. Large kernel convolution provides the inspiration for the above issue. To explore the potential of large kernel convolution, we propose a hyperspectral image (HSI) classification algorithm in this article that utilizes a large kernel convolution module combined with multiscale coattention and an adaptive geometric feature (AGF) classifier, named Hyper-LKCNet. By integrating this feature enhancement module, our method effectively adjusts the contributions of various spectral and spatial features, ensuring that the network captures critical but easily overlooked information across both dimensions and improving the performance to classify HSI. The AGF classifier, derived by neural collapse theory, alleviates the sample imbalance problem and incorporates the label smoothing focal loss function to enhance generalization ability. Extensive experiments on four HSI datasets demonstrate that the proposed method outperforms the state-of-the-art approaches. In addition, our algorithm maintains a low parameter count and reduced floating point of operations.**

***Index Terms*—Attention mechanism, hyperspectral image (HSI) classification, imbalanced data, large kernel convolution.**

I. Introduction

REMOTE sensing technology has undergone significant theoretical and practical developments since the latter half of the 20th century. Among these advancements, hyperspectral imaging has emerged as a critical innovation in improving remote sensing technology. Hyperspectral imaging acquires unique, continuous spectra of land surfaces in a wide spectral range, often encompassing tens to hundreds of spectral bands. This process generates 3-D, cube-structured hyperspectral images (HSIs), which are rich in information and cover a broad spectrum, enabling detailed land-cover classification. Therefore, HSI plays an important role in real-world applications, such as environmental monitoring [1], [2], urban development [3], [4], and resource surveying [5].

However, compared with traditional image classification tasks [6], HSI classification faces several unique challenges, including information redundancy [7], the curse of dimensionality [8], and limited labeled samples. These challenges make HSI classification significantly more challenging [9], [10]. The traditional machine learning methods, such as the K-nearest neighbor classifier [11], the maximum likelihood classifier [12], and the distance classifier [13], primarily leveraged spectral information without fully exploiting the spatial data, whereas the redundant spectral information gives rise to a large number of misclassified cumulative. In response, researchers have explored spectral selection [14] and feature selection [15] methods, which aim to reduce data dimensionality and generate more effective feature representations. Moreover, HSI labeling is complicated by complex land-cover distribution, changing ambient light, and spectral heterogeneity. Under this circumstance, traditional feature extraction methods [16] heavily rely on expert knowledge and manual design, limiting their adaptability to different datasets and preventing the full utilization of HSIs information.

Deep learning has introduced a transformative solution to these limitations [21]. It automates feature extraction and reduces dependence on manual parameter settings, allowing models to adaptively learn features directly from data. Recent years have witnessed a boom in deep learning in image classification, target detection [17], and other fields. Various deep learning models, including stacked autoencoders [18], recurrent neural networks [19], convolutional neural networks (CNNs) [20], and transformers [21], [22], have been applied to HSI classification, yielding remarkable outcomes. CNNs have garnered significant attention due to their ability to automatically extract local features. However, most CNN-based methods primarily extract

Received 15 March 2025; revised 21 April 2025; accepted 11 May 2025. Date of publication 20 May 2025; date of current version 9 June 2025. This work was supported in part by the National Natural Science Foundation of China under Grant 62201622, in part by Guangdong Basic and Applied Basic Research Foundation under Grant 2024A1515010785, and in part by the Opening Project of Guangdong Province Key Laboratory of Computational Science at the Sun Yat-sen University under Grant 2024019. *(Corresponding author: Jiaqi Yang.)*

Rong Liu and Zhilin Li are with the School of Geography and Planning, Sun Yat-sen University, Guangzhou 510275, China (e-mail: liurong25@mail.sysu.edu.cn; lizhlin27@mail2.sysu.edu.cn).

Jiaqi Yang is with the State Key Laboratory of Information Engineering in Surveying Mapping, and Remote Sensing, Wuhan University, Wuhan 430079, China (e-mail: jqyang@whu.edu.cn).

Jian Sun is with the Enshi Urban Planning and Design Institute Company Ltd., Enshi 445099, China (e-mail: jiansun@whu.edu.cn).

Quanwei Liu is with the College of Science and Engineering, James Cook University, Cairns, QLD 4878, Australia (e-mail: quanwei.liu@my.jcu.edu.au).

The code will be available at https://github.com/liurongwhm.

Digital Object Identifier 10.1109/JSTARS.2025.3571954

features from several convolutional or pooling layers, which limits the receptive field. This restricted focus on local features leads to the exclusion of critical global information [20] and diminishes the ability to capture subtle spectral differences in neighboring spectral bands.

The advent of transformers makes it possible to mitigate this issue. Transformers have gained widespread attention due to their self-attention mechanisms, which enable the capture of long-term dependencies across different positions within a sequence [23]. When applied to HSI, transformers can effectively utilize the rich and continuous spectral information in HSI data, capturing complex spectral patterns and improving classification performance, including spectral attention transformer [24], deep hierarchical vision transformer [22], and spectral–spatial transformer network [25], [26]. Despite their advantages, transformers face challenges, such as high computational costs and insufficient local information extraction, making it difficult for them to entirely replace CNNs. Recent studies have explored hybrid models that combine CNN and vision transformer architectures for HSI classification [27], [28] to obtain both local details and global dependencies. Yet, these approaches often struggle to balance accuracy with computational efficiency and the inherent redundancy of stacked attention layers remains unaddressed.

CNN architectures, with their local connectivity and weight-sharing capabilities, remain computationally efficient and continue to attract research attention. To further improve accuracy without a large computational burden, several recent studies [29], [30] have attempted to use the large kernel convolution in natural images. Large kernel convolutions improve model performance and achieve higher accuracy at a lower computational cost than transformers by providing a larger receptive field [31], [32] to capture global features. However, HSI data have significant spectral–spatial heterogeneity, which makes it difficult for the model to fully adapt to the data features when large kernels are directly applied to HSI, thus affecting performance.

In the HSI classification task, the large kernel convolution design is significantly different from the design in computer vision tasks. In computer vision tasks, large kernel convolution is mainly used to expand the receptive field to capture long-range spatial dependencies, usually by stacking large kernel convolution layers or combining attention mechanisms. This design has achieved remarkable results in natural images, but it needs to be further adjusted in HSI classification to adapt to its unique data structure. The design of a large kernel convolution needs to take into account the extraction of spectral and spatial features. Since HSI has high-dimensional spectral information, each pixel contains dozens or even hundreds of continuous spectral bands. Directly applying a large kernel convolution may lead to a sharp increase in computational complexity. Therefore, the design needs to be combined with relevant convolution strategies to reduce computational costs and improve feature extraction efficiency. In addition, recent research has achieved efficient extraction of irregular shape features by combining deformable large kernel attention [33] with deformable convolution. This idea can be transferred to HSI processing to dynamically adjust the shape of the convolution kernel to adapt to the complexity of spectral features. In addition, depth-separable convolution can significantly reduce the number of parameters by decoupling spatial filtering and channel fusion. For example, the MobileNet series [34] has verified its lightweight advantages in natural images and can also be migrated to HSI classification for related applications. Furthermore, the large convolution kernel attention mechanism strengthens the feature expression of key areas through weight distribution. For example, in the glioma grading task, the dual-domain attention [35], [36] significantly improves the fusion effect of multimodal information, which is inspiring for the collaborative analysis of HSI multiband features.

Another critical challenge in HSI classification is the limited availability of labeled samples, which are often expensive and difficult to obtain. In recent years, unsupervised learning has also made significant progress in the field of HSI classification, and unsupervised methods based on self-supervised learning and contrastive learning frameworks have gradually become a research hotspot. The spatial–spectral masked autoencoder [37] captures the correlation between spatial and spectral features through a mask modeling strategy, solving the limitation of traditional masked autoencoders that only focus on a single modality; while the nearest neighbor-based contrastive learning [38] optimizes feature representation through local sample similarity, improving the model's utilization of unlabeled data. However, unsupervised learning still has the problem of data distribution sensitivity [39], which is easily affected by the uniformity of input data distribution and tends to be biased toward the dominant class in scenarios with imbalanced categories. There is a dependency on parameter tuning, which requires a lot of experimental verification. This results in insufficient generalization and limited application. Moreover, in some cases, the number of samples per class may vary significantly, yet most classifiers are designed under the assumption that sample sizes across classes are balanced. When confronted with imbalanced data, classifiers may produce biased results, with certain classes perceived as noise or distorted [40]. To address this problem, researchers have proposed various solutions, often categorized as either preprocessing techniques [41] or improved classifiers [42]. Preprocessing methods typically attempt to balance datasets through resampling, either by adding samples to underrepresented classes or by removing samples from overrepresented classes [43], [44]. However, in HSI, where labeled data are scarce, removing samples can result in information loss [45], while adding samples may introduce noise, diminishing the classifier's generalization ability. Consequently, traditional preprocessing techniques may not be well suited for HSI classification, making classifier improvements a more viable solution.

Improved classifiers often focus on enhancing generalization by enriching features or adjusting the loss function [46], [47]. The loss function, which defines the optimization objective, measures the classifier's accuracy during training and adjusts parameters to minimize classification errors. Regularization terms can also be added to the loss function [48], reducing overfitting

by penalizing overly large parameter values and improving the model's generalization ability [49].

In this article, we propose a novel HSI classification algorithm designed to overcome the challenges mentioned above. Our framework leverages large kernel convolution-based CNN architectures and explores the potential of large kernel convolution for HSI, providing a paradigm for the application of large kernel convolution in the field of HSI classification. By dynamically adjusting the weights of different spectral and spatial locations through multiscale coattention (MSCA), the network could focus more on critical features that are beneficial for classification. In addition, to address the problem of imbalanced samples in HSIs, we introduce an adaptive geometric feature (AGF) classifier, combined with a label smoothing focal (LSF) loss function to further enhance the model's generalization ability.

We validate the effectiveness of the proposed method through comprehensive experiments on four benchmark HSI datasets. Our primary contributions are summarized as follows.

1) A CNN architecture based on large kernel convolution is proposed to address the limitations of traditional CNNs in capturing long-range features. By introducing large kernel convolution, our architecture is capable of extracting both detailed features of small targets and the global distribution of large targets, enhancing the overall feature representation in HSIs.
2) Within the large kernel convolutional architecture, a spectral–spatial attention mechanism, called MSCA, is utilized to distinguish contributions with different characteristics. This mechanism enables the network to focus on the most critical spectral and spatial information by using convolutional kernels at different resolutions to extract features across multiple scales. This design enhances the complementarity between features and emphasizes important spectral bands and spatial pixels, assisting the network to better extract intrinsic features.
3) To alleviate the issue of hyperspectral sample imbalance, the AGF classifier is introduced. This classifier combines fully connected layers and an equiangular tight frames (ETF) classifier, where the fully connected layers optimize the feature inputs, and the ETF structure balances class differences by normalizing eigenvectors and maximizing the pairwise angles between them. The proposed approach improves classification stability and reliability across diverse datasets.
4) An LSF loss function is designed to be combined with the AGF classifier to achieve better model confidence calibration and improve the model's generalization ability. The designed loss function helps mitigate the overfitting phenomenon that may arise due to large kernel convolution, reduces the impact of feature normalization, and promotes better clustering of samples within each category. In addition, it is tailored to harmonize with the AGF classifier for optimal performance.

The rest of this article is organized as follows. Section II describes the proposed framework in detail. Section III reports the experimental results and analysis. Section IV discusses the important findings. Finally, Section V concludes this article.

II. Methodology

A. Overall Framework

The overall classification architecture is depicted in Fig. 1. First, the raw HSI is preprocessed by principal component analysis for spectral downscaling. The preprocessed data are then fed into a convolutional network, where multiple conv-3D layers are used to simultaneously extract detailed spectral and spatial features from the HSI data. These features are subsequently fed into an MSCA mechanism block, which focuses on the most important features.

To reduce both the computational complexity and the parametric quantities of the large kernel convolutional, we adopt depthwise-separable convolution [50] for global feature capture. Depthwise-separable convolution can be decomposed into depthwise convolution and pointwise convolution. To begin with, the depthwise convolution performs a large kernel operation independently on each channel, greatly reducing the amount of required calculation. Next, the pointwise convolution fuses the information across channels through a 1×1 convolution, retaining the global feature-capturing ability of the large kernel. The strategy of depthwise-separable convolutions provides a solution for the efficient use of large kernel convolutions and lays the basic framework of our model.

On that basis, joint AGF classifier and LSF loss function are used to generate feature vectors. LSF loss effectively alleviates the problem of class imbalance by smoothing the labels, avoiding the bias of traditional cross-entropy (CE) loss in the case of class imbalance, thereby improving the classifier's ability to identify minority categories. In addition, LSF loss enhances the model's learning ability by reducing excessive penalties between categories, making the classifier more stable and robust during feature learning. Finally, these features are fused and fed into a SoftMax layer to obtain the predicted labels of the samples.

B. Multiscale Coattention

Attention mechanisms are widely used in a range of tasks. In the context of visual attention, these mechanisms are generally categorized into three types [51]: spatial attention, channel attention, and joint spatial–spectral attention. The squeeze-and-excitation attention (SENet) [52], one of the most popular mechanisms in deep learning networks, computes channel attention by using 2-D global pooling to obtain significant performance gains. However, it only focuses on interchannel dependencies, overlooking spatial features. Convolutional block attention module (CBAM) [53] introduced convolutions to extract spatial features, but ignored long-range dependencies [54]. Coordinate attention [55] improved upon SENet and CBAM by incorporating direction-dependent position information, resulting in excellent performance across various detection and segmentation tasks.

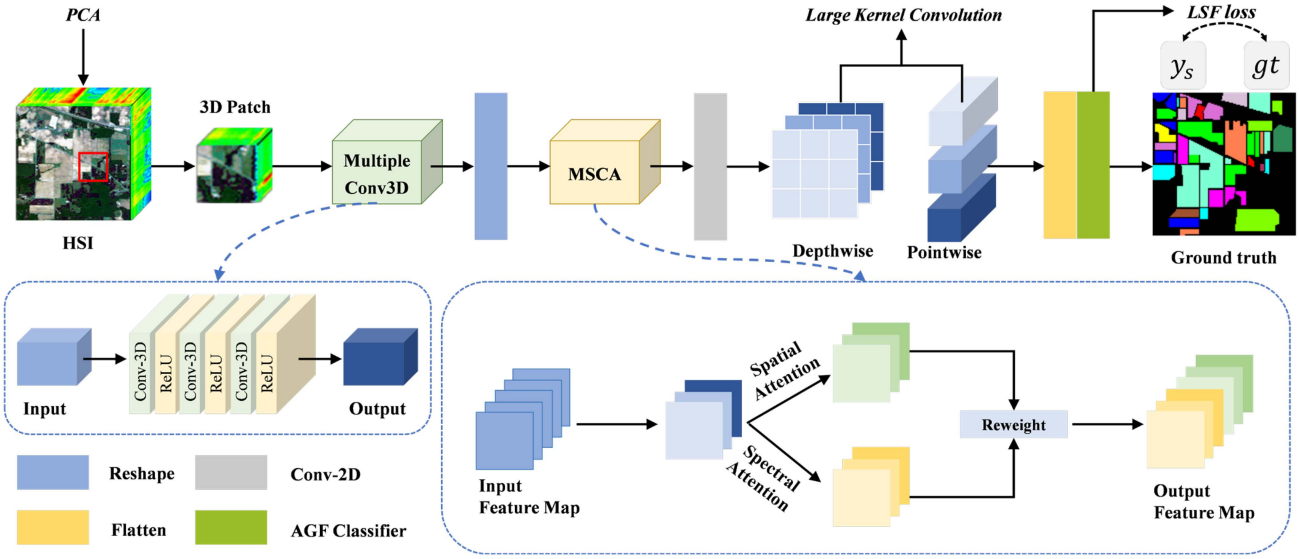


Fig. 1. Flowchart of the proposed framework. The specific structures of the MSCA mechanism and the AGF classifier are shown in Figs. 2 and 3, respectively.

In HSI classification, it is crucial to fully utilize spectral and spatial information. Therefore, drawing inspiration from the CA attention mechanism and considering the spectral–spatial multidomain characteristics of HSIs, we propose a more effective attention mechanism tailored to HSI classification.

Different from SENet, SENet uses a "squeeze–excitation" operation, global average pooling (GAP), and fully connected layers to explicitly learn channel weights, adopts a single-channel attention branch, and lacks adaptive adjustment to spatial dimension features. Suboptimal weight distribution may occur in hyperspectral scenes with complex spatial distribution. MSCA is aimed at multiscale feature collaborative optimization, aiming to solve the scale sensitivity problem caused by diverse target sizes and complex boundaries in HSIs. By connecting convolutional layers of different sizes in parallel, a multiscale receptive field is explicitly constructed, and the dynamic fusion of local details and global context is achieved by combining spatial–spectral dual-path attention. MSCA is not a simple variant of SENet, but a scale-adaptive attention framework designed for the characteristics of hyperspectral data. Unlike CA attention, which only utilizes single-scale spatial features, our method introduces a multiscale feature extraction module. This module captures spatial features at different scales through varying convolutional layer sizes and fuses them to capture the subtle differences in HSIs. For spectral features, each spectral band is processed independently using 1×1 convolution to capture local spectral features. These key features from both the spatial and spectral dimensions are combined to produce classification results with high accuracy. This method, which emphasizes multiscale convolution and coordinated attention across dimensions, is referred to as MSCA. Fig. 2 provides a schematic diagram of the proposed attention block.

In the spatial dimension, input features are aggregated along two directions, vertical and horizontal, using 1-D global pooling operations. The input feature map $\boldsymbol{X}\in\mathbb{R}^{H\times W\times C}$ is pooled, and

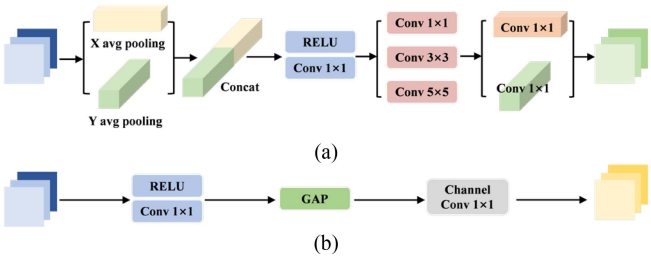


Fig. 2. Operations in the spectral and spatial branches of the MSCA block. (a) Spatial branch. (b) Spectral branch.

the outputs of the cth channel along the height (h) and width (w) dimensions are obtained as follows:

$$z_c^h(h) = \frac{1}{W}\sum_{i=0}^{W} x_c(h,i) \tag{1}$$

$$z_c^w(w) = \frac{1}{H}\sum_{j=0}^{H} x_c(j,w) \tag{2}$$

where $x_c(h,w)$ denotes the value of the cth channel at (h,w).

Next, the outputs along the height and width are concatenated and passed through a 1×1 convolution layer and a rectified linear unit (ReLU) activation function [56] to produce an intermediate feature map, reducing the dimension to C/r, with r being the reduction ratio, indicating the degree of reduction in the number of feature map channels (C), thereby reducing computational complexity and the number of parameters

$$\boldsymbol{f}_1 = \mathrm{ReLU}\left(\mathrm{Conv}_{1\times1}\left(\left[z^h, z^w\right]\right)\right). \tag{3}$$

Subsequently, multiscale features are extracted using convolutional kernels of various scales, and these features are fused

by element-by-element summation

$$\boldsymbol{F}_{k_i}\ (f_1) = \mathrm{Conv}\left(f_1, k_i\right), i \in \{1,3,5\} \tag{4}$$

$$\boldsymbol{f}_2 = \mathrm{ReLU}\left(\sum_{i=1}^{3} \boldsymbol{F}_{k_i}\right) \tag{5}$$

where $\mathrm{Conv}(\boldsymbol{f}_1, k_i)$ denotes a convolution operation using a convolution kernel of size i.

To construct the spatial attention map $\boldsymbol{g}$, the output $\boldsymbol{f}_2$ is divided into two tensors $\boldsymbol{f}^h$ and $\boldsymbol{f}^w$, which are processed through additional 1×1 convolution layer and the sigmoid function σ

$$\boldsymbol{g}^h = \sigma\left(\mathrm{Conv}^h_{1\times1}\left(\boldsymbol{f}^h\right)\right) \tag{6}$$

$$\boldsymbol{g}^w = \sigma\left(\mathrm{Conv}^w_{1\times1}\left(\boldsymbol{f}^w\right)\right). \tag{7}$$

With this operation, the feature maps are encoded into two separate attention maps: one along the height and the other along the width. These attention maps are applied complementarily to the input feature maps, capturing long-range dependencies along each spatial direction. This completes the construction of spatial-dimensional attention.

In the spectral dimension, a 1×1 convolution is applied to each position in the spectral axis to fuse information across bands, learning the interrelationships between different spectral bands

$$\boldsymbol{f}_3 = \mathrm{ReLU}\left(\mathrm{Conv}_{1\times1}\left(\boldsymbol{X}\right)\right). \tag{8}$$

GAP is then performed on $\boldsymbol{f}_3$, followed by a 1×1 convolution, which expands 1 channel to C channels. And sigmoid function is used to generate spectral attention weights

$$\boldsymbol{g}^c = \sigma(\mathrm{Conv}^c_{1\times1}\left(\mathrm{GAP}\left(\boldsymbol{f}_3\right)\right). \tag{9}$$

GAP computes global features for each channel, compressing each channel's values into a scalar. This step greatly reduces the spatial dimension of the feature map while preserving the global information of each channel.

Finally, the output of MSCA is computed by multiplying the input $\boldsymbol{X}$ with the combined spatial and spectral attention maps

$$\boldsymbol{Y}\ (i,j) = \boldsymbol{X}\ (i,j) \times (1 + \sigma(\boldsymbol{g}^h\ (i,j) \times \boldsymbol{g}^w\ (i,j) \times \boldsymbol{g}^c\ (i,j))) \tag{10}$$

where $\boldsymbol{Y}$ is the feature map, and 1 is added to retain the original features without losing the original key information. With this hybrid attention layer, the network's ability to capture both global and local information is significantly enhanced.

C. AGF Classifier

Neural collapse is a phenomenon observed during the final stage of training (when the training loss is close to zero), where the last-layer features from the same class converge to a single vertex [57]. When the training dataset is balanced, these class-specific vertices and their corresponding classifier vectors form a structure called the ETF, in which the vectors are normalized and the pairwise angles are maximized [58]. This predefined ETF structure is usually defined as follows:

$$\boldsymbol{V} = \sqrt{\frac{C}{C-1}}\,\boldsymbol{U}\left(\boldsymbol{I}_C - \frac{1}{C}1_C 1_C^T\right) \tag{11}$$

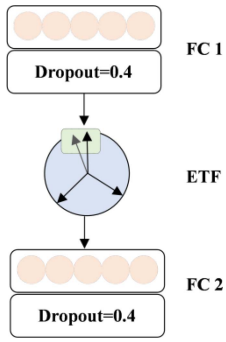


Fig. 3. Structure of the AGF classifier. The ETF classifier represents its feature distribution that achieves feature vector normalization with maximized angle separation in the feature space. FC means the fully connected layer.

where $\boldsymbol{V} = [\boldsymbol{v}_1, \ldots, \boldsymbol{v}_C] \in \mathbb{R}^{d\times C}$. $\boldsymbol{U} \in \mathbb{R}^{d\times C}$ is a rotation matrix, which satisfies $\boldsymbol{U}^T\boldsymbol{U} = \boldsymbol{I}_C$; $\boldsymbol{I}_C$ is a unit matrix and 1_C is an all-one vector. All vectors in the ETF have equal $\updownarrow_\in$ norms and the same diagonal angles, i.e.,

$$\boldsymbol{v}_i^T\ \boldsymbol{v}_j = \begin{cases} -\frac{1}{C-1}, i \neq j \\ 1,\ i = j \end{cases} \quad \forall i,j \in [C] \tag{12}$$

where the paired angle $-\frac{1}{C-1}$ is the maximal equiangular separation of the C-vectors in $\mathbb{R}^d$.

The parameters need to be manually adapted to the specific characteristics of each dataset during the training process to achieve optimal classification results. This requires extensive experiments and validation, relying heavily on the researcher's experience and intuition. Generally, it is difficult to find the globally optimal combination of parameters. Therefore, an automated, parameter-free tuning module is necessary.

The ETF framework defines the optimal geometric configuration for the classifier by balancing the minimization of within-class variation with the maximization of between-class variation and has demonstrated good results in handling imbalanced training in the recent study [59]. In this article, we propose the AGF classifier as a network classifier to enhance category differentiation and improve model stability, as shown in Fig. 3.

Unlike previous works [60], [61], which directly configure classifiers to the optimal ETF structure before network training, our AGF classifier combines fully connected layers with the ETF classifier to extract and optimize features learned by the network. A simple ETF classifier may lead to overstretching of the feature vector. Therefore, we introduced a fully connected layer to optimize the learning ability of the feature space. The fully connected layer generates higher level global features by weighted combination of input features, helps eliminate redundancy and noise, and screens out more representative and separable features. In addition, the trainable weight parameters of the fully connected layer help avoid overstretching of features and alleviate the oversmoothing phenomenon that may be caused by large kernel convolution, thereby improving the classifier's ability to recognize complex ground object categories.

After the previous network convolution operations and feature spreading, a feature map $\boldsymbol{X}^f \in \mathbb{R}^{N\times(H\times W\times C)}$ is obtained, where

N is the batch size. This feature map is then passed through a fully connected layer, with a dropout rate $p = 0.4$ applied to avoid overfitting

$$\boldsymbol{h}_1 = \text{ReLU}\left(\boldsymbol{X}^f \boldsymbol{W}_1 + \boldsymbol{b}_1\right) \tag{13}$$

$$\boldsymbol{h}_1^d = \text{Dropout}\left(\boldsymbol{h}_1, p = 0.4\right) \tag{14}$$

where $\boldsymbol{W}_1 \in \mathbb{R}^{(H\times W\times C)\times D_1}$ is the weight matrix, $\boldsymbol{b}_1 \in \mathbb{R}^{D_1}$ is the bias vector, and D_1 is the dimension of the hidden layer after the fully connected layer operation. The new features $\boldsymbol{h}_1^d$ are obtained to capture complex nonlinear relationships in the high-dimensional data, providing more representative and separable features.

Then, the classifier is aligned with ETF, and a simplex ETF is randomly formed, i.e., $\boldsymbol{U}$ in (15), and $\boldsymbol{V}$ is obtained by placing $\boldsymbol{U}$ into (15) for the ETF vector computation

$$\boldsymbol{V} = \boldsymbol{U} \cdot \boldsymbol{G} \tag{15}$$

Noted that $\boldsymbol{U} \in \mathbb{R}^{d\times c}$, and $\boldsymbol{U}^T\boldsymbol{U} = \frac{c}{d}\boldsymbol{I}_c - \frac{1}{d}\boldsymbol{I'}_{c\times c}$, where $\boldsymbol{d}$ denotes the number of feature dimensions of the ETF and c denotes the number of classes. $\boldsymbol{I}_c$ is the identity matrix with the shape of $c \times c$, and $\boldsymbol{I'}$ refers to a matrix entirely filled with ones with the shape of $c \times c$.

A projection layer is used to map the raw features $\boldsymbol{h}_1^d$ into the ETF feature space, where the predictive score $\boldsymbol{\mu}$ for each class is computed by multiplying the normalized feature vector with the ETF vectors

$$\hat{\boldsymbol{h}} = g\left(\boldsymbol{p}; \boldsymbol{h}_1^d\right) \tag{16}$$

$$\boldsymbol{h} = \hat{\boldsymbol{h}} \,/\, \|\hat{\boldsymbol{h}}\|_2 \tag{17}$$

$$\boldsymbol{\mu} = \boldsymbol{h} \times \boldsymbol{V} \tag{18}$$

where $\boldsymbol{p}$ denotes the parameters of the projection layer. The projection layer is essential in our AGF classifier design. Hyperspectral raw features are typically high dimensional, and such vectors are easier to orthogonalize but harder to collapse into angle-maximizing ETFs. Therefore, some mapping of the features through the projection layer is necessary. The projection can also be fine-tuned if needed.

Finally, the category scores are further converted into new hidden layer features to combine different input features

$$\boldsymbol{h}_2 = \text{ReLU}\left(\boldsymbol{\mu}\boldsymbol{W}_2 + \boldsymbol{b}_2\right) \tag{19}$$

$$\boldsymbol{h}_2^d = \text{Dropout}\left(\boldsymbol{h}_2, p = 0.4\right) \tag{20}$$

where $\boldsymbol{W}_2 \in \mathbb{R}^{C\times D_2}$ is the weight matrix, $\boldsymbol{b}_2 \in \mathbb{R}^{D_2}$ is the bias vector, and D_2 is the dimensionality of the hidden layer after the second fully connected layer operation.

Through the training process, the feature vector obtains the optimal geometric structure under the guidance of the neural collapse phenomenon. The feature vector normalization and angle-maximized feature distribution are maintained in the feature space, emphasizing the use of the ETF structure and the geometric regularization method to constrain the class prototype to an equiangular distribution on the unit hypersphere, enhance the interclass separability, and alleviate the dominant effect of the majority class samples, ensuring that the minority class samples form a discriminative feature submanifold. Combined with the fully connected layer, a purer and more representative feature vector is generated, the separability of the minority class features is enhanced, and a classifier model with a better generalization effect is obtained.

D. LSF Loss

HSI contains rich spectral and spatial information, but limited number of labeled samples and the imbalanced distribution of samples across categories pose significant challenges. The widely used CE loss function [62] focuses on correctly labeled predictions, ignoring model performance on incorrect categories. When dealing with class imbalance, CE loss often favors majority classes, leading to poor performance on minority categories. To alleviate this problem and enhance the generalization ability of the AGF classifier, we propose an LSF loss, which improves the network's ability to represent features.

Focal loss [63] is an extension of the CE loss, initially designed for binary classification in target detection tasks, and is used to address class imbalance by focusing on hard-to-classify examples. However, HSI classification is a multiclass problem, we generalize focus loss for multiclass classification by denoting the number of classes as n

$$L\ (p_i, y_i) = y_i \log p_i \tag{21}$$

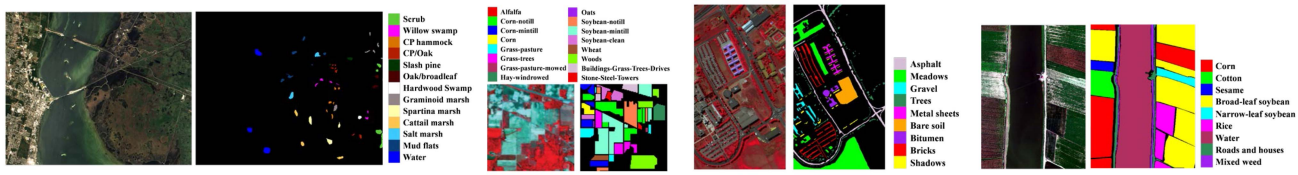
$$\text{FL}\,(p) = \sum_{i=1}^{n} \alpha(1-p_i)^{\gamma}\ L\,(p_i, y_i) \tag{22}$$

where p_i denotes the probability obtained by the ith sample after the activation function, y_i denotes the true label of the ith sample, α is the category balancing factor, which can be set as desired; and γ is the modulation factor, used to reduce the contribution of easily classified samples. When a sample is misclassified (i.e., p_i is small), the modulation factor $(1-p_i)$ is close to 1, meaning that the loss remains unaffected. Conversely, when a sample is correctly classified (i.e., p_i approaches 1), $(1-p_i)$ approaches 0, and the weight of that sample is tuned down.

Large kernel convolution naturally extracts more information due to its larger receptive field, but this can lead to oversmoothing, where the receptive field of the kernel is too large, causing local information to be oversmoothed during processing, thereby losing fine-grained features and boundary information, ultimately affecting classification accuracy. Label smoothing [64] is a regularization strategy that suppresses overfitting by assigning a small portion of the weight from the true labels to other classes, thus smoothing the label distribution and reducing dependence on a single class. We incorporate label smoothing into the focal loss function, introducing slight noise into the label distribution to avoid overfitting to the training data. When label smoothing is applied, the true label y_i is smoothed to y_s, denoted as follows:

$$y_s = (1-\epsilon)\ y_i + \frac{\epsilon}{n} \tag{23}$$

where ϵ is the label smoothing parameter, taking values in the range [0, 1]. The combined loss function with label smoothing



(a) (b) (c) (d)

Fig. 4. Pseudocolor images and ground-truth maps of experimental datasets. (a) KSC. (b) IP. (c) PU. (d) LK.

TABLE I
TRAINING AND TEST SAMPLES FOR KSC DATASET

Class	Class Name	Training	Test
1	Scrub	23	738
2	Willow swamp	13	230
3	CP hammock	14	242
4	CP/Oak	14	238
5	Slash pine	5	156
6	Oak/broadleaf	12	217
7	Hardwood Swamp	9	96
8	Graminoid marsh	28	403
9	Spartina marsh	41	479
10	Cattail marsh	29	375
11	Salt marsh	31	388
12	Mud flats	39	464
13	Water	81	846
	Total number	339	4872

is expressed as follows:

$$\mathrm{LFS}\left(y_s, p_i\right) = \sum_{i=1}^{Nn} \alpha \left[\left(1-\epsilon\right) y_i + \frac{\epsilon}{n}\right] \left(1-p_i\right)^{\gamma} \log\left(p_i\right). \tag{24}$$

Overall, the LFS loss function addresses the challenges of training on imbalanced datasets by developing the CE loss. Adding label smoothing to the focal loss function achieves regularization, mitigates the oversmoothing effects of large kernel convolution, and enhances the model's ability to capture fine-grained features.

III. Experimental Results and Analysis

A. Data Description

In our experiments, four hyperspectral datasets, including the Kennedy Space Center (KSC), Indian Pines (IP), Pavia University (PU), and WHU-Hi-Longkou (LK), are utilized for evaluating the performance of the proposed method. The pseudocolor images of these datasets and their corresponding ground-truth maps are shown in Fig. 4(a)–(d). The training and test samples of four datasets are shown in Tables I– IV.

1) *KSC* [65]: The dataset was acquired by the airborne visible/infrared imaging spectrometer (AVIRIS) sensor over the KSC, Florida, USA. It contains 176 active bands with 512×614 pixels after removing absorption and low signal-to-noise ratio bands. The spectral coverage is 0.4–2.5 μm, with a spatial resolution of 18 m. After removing background pixels, 5211 pixels are retained, referring to 13 land-cover types.

TABLE II
TRAINING AND TEST SAMPLES FOR IP DATASET

Class	Class Name	Training	Test
1	Alfalfa	20	26
2	Corn-notill	90	1338
3	Corn-mintill	90	740
4	Corn	90	147
5	Grass-pasture	90	393
6	Grass-trees	90	640
7	Grass-pasture-mowed	10	18
8	Hay-windrowed	90	388
9	Oats	5	15
10	Soybean-notill	90	882
11	Soybean-mintill	90	2365
12	Soybean-clean	90	503
13	Wheat	90	115
14	Woods	90	1175
15	Buildings-Grass-Trees-Drives	40	346
16	Stone-Steel-Towers	40	53
	Total number	1105	9144

TABLE III
TRAINING AND TEST SAMPLES FOR PU DATASET

Class	Class Name	Training	Test
1	Asphalt	50	6581
2	Meadows	50	18,599
3	Gravel	50	2049
4	Trees	50	3014
5	Painted metal sheets	50	1295
6	Bare Soil	50	4979
7	Bitumen	50	1280
8	Self-Blocking Bricks	50	3632
9	Shadows	50	897
	Total number	450	42,326

TABLE IV
TRAINING AND TEST SAMPLES FOR LK DATASET

Class	Class Name	Training	Test
1	Corn	50	34,461
2	Cotton	50	8324
3	Sesame	50	2981
4	Broad-leaf soybean	50	63,162
5	Narrow-leaf soybean	50	4101
6	Rice	50	11,804
7	Water	50	67,006
8	Roads and houses	50	7074
9	Mixed weed	50	5179
	Total number	450	300,829

2) *IP* [65]: The dataset was acquired by the AVIRIS sensor at the IP proving ground in northwestern Indiana. After removing absorption bands, the image consists of 200 spectral bands with 145×145 pixels. The spectral coverage is 0.4–2.5 μm, with a spatial resolution of 20 m. After

removing background pixels, 10 249 pixels are retained, covering 16 land-cover types.

3) *PU* [65]: The dataset was obtained from the reflectance optical system imaging spectrometer sensor over the University of Pavia, Northern Italy. After removing 12 noisy and water-absorbing bands, 103 bands of data with 610×340 pixels were retained. The spectral coverage is 0.43–0.86 μm, with a spatial resolution of 1.3 m. After removing background pixels, 42 776 pixels are retained, representing 9 land-cover types.
4) *LK* [66]: The dataset was acquired in Longkou City, Hubei Province, China, on 17 July 2018, using an 8 mm focal length Headwall Nano-Hyperspec image sensor mounted on a DJI Matrice 600 Pro UAV. The image size is 550×400 pixels with 270 bands. The spectral coverage is 0.4–1.0 μm, with a spatial resolution of 0.463 m. After removing background pixels, 301 279 pixels are retained, including 9 land-cover types.

B. Experimental Settings

1) Implementation Details: The proposed Hyper-LKCNet is implemented using an NVIDIA GeForce RTX 4090 graphics card. In each repetition, a random training set is generated from the data, and the remaining reference samples are used as the test set. An adaptive moment estimation (Adam) optimizer is used, with the optimal learning rate set to 10^{-4} based on the classification results. The number of epochs for the KSC, IP, PU, and LK datasets is 100, 80, 60, and 40, respectively. The number of principal components for the KSC, IP, PU, and LK datasets is 20, 30, 15, and 20, respectively.

2) Metrics: To evaluate the performance of the proposed framework, we use four quantitative metrics: per-class accuracy, overall accuracy (OA), average accuracy (AA), and kappa coefficient (κ). Each experiment is repeated at least ten times.

3) Comparison Algorithms:

a) *2D-CNN* [67]: 2D-CNN is used for baseline comparison. The model is implemented using the DeepHyperX database and trained from scratch with stochastic gradient descent (momentum = 0.9 and weight decay = 0.0005). The batch size is 60, and the epochs are 30.
b) *3D-CNN* [68]: This network consists of 3-D convolutional layers followed by batch normalization and max-pooling layers. The 3-D convolution block sizes are 8, 16, and 32, with a filter size of 3×3×3. The learning rate is 0.1, decay period is 0.09, batch size is 100, and epochs are 50.
c) *Spectral–Spatial Residual Network (SSRN)* [69]: SSRN is a 3-D convolution-based classification model with a spatial residual network. The learning rate is 0.001, optimized with the root-mean-square prop, and the epochs are 200. The patch size of each pixel is 7×7.
d) *HybridSN* [70]: This is a hybrid network combining spectral–spatial 3D-CNN and spatial 2D-CNN. The method runs for 100 epochs with a learning rate of 0.001 on 3-D patches with spatial dimensions of 25×25.
e) *Center-to-Surrounding Interactive Learning* (*CSIL*) [71]: This multiscale spatial classification network is based on a hierarchical region sampling strategy, center transformer, and surrounding transformer. It is trained for 70 epochs with a learning rate of 0.0001 and a spatial dimension of 27×27.
f) *Online Spectral Information Compensation Network* (OSICN) [72]: OSICN is a novel online spectral information compensation network. The input HSI cube size is 28×28×c (where c is the number of bands). It is trained with a learning rate of 0.0001 with the Adam optimizer for 200 epochs.
g) *HSI Classification Full Model* (HSIC-FM) [73]: This is a classification network using representative features and multidomain feature fusion. It is trained with a learning rate of 0.001 using the Adam optimizer for 150 epochs. The input HSI cube size is 27×27×1.
h) *HSI Classification Specifically Designed Model* (HSIC-SDM) [74]: This model contains three branches for frequency feature learning, spatial feature learning, and spectral feature learning. The model is trained for 200 epochs with a learning rate of 0.0005. The spatial patch size is set to 11×11.
i) *Channel-Layer-Oriented Lightweight Spectral–Spatial Network (CLONLN)* [75]: This is a lightweight end-to-end compact and efficient model. The model is trained for 300 epochs with a learning rate of 0.05. The spatial patch size is set to 13×13.

C. Classification Results

The quantitative results are presented in Tables V– VIII, comparing the classification performance of our algorithm with other state-of-the-art (SOTA) methods on four datasets. The classification maps are shown in Figs. 5– 8.

For the KSC dataset, as shown in Table V and Fig. 5, the 2D-CNN shows poor performance due to its inability to capture spectral information, resulting in salt and pepper noise across all the classes. 3D-CNN is an improvement over 2D-CNN, but still not very effective. SSRN, using 3-D convolutional layers with residual attention, undergoes ineffective residual linkage with limited training samples, leading to noticeable errors in the classification map. HybridSN, which combines 3D-CNN and 2D-CNN, demonstrates stronger spatial–spectral feature extraction, yielding improved results. However, the accuracy is still lower than that of the proposed method. HSIC-SDM and CLONLN achieve high OA, but their AAs are lower due to poor classification of certain classes (e.g., Class 7 “Hardwood Swamp” for HSIC-SDM and Class 4 “CP/Oak” for CLONLN), resulting in block errors in the classification maps. CSIL effectively suppresses salt and pepper noise and achieves acceptable performance for many classes, but oversmoothing phenomenon causes the degradation of accuracy. Both OSICN and HSIC-FM show good quantitative performance, with less salt and pepper noise in the classification map; however, OSICN struggles with Class 5 (80.13% OA), and HSIC-FM struggles with Class 1 (below 80.00% OA). Our Hyper-LKCNet outperforms other SOTA methods, achieving the highest OA, AA, and κ scores. The KSC dataset poses unique challenges due to its decentralized and

TABLE V
QUANTITATIVE RESULTS FOR KSC DATASET

Class	2D-CNN	3D-CNN	SSRN	HybridSN	CSIL	OSICN	HSIC-FM	HSIC-SDM	CLOLN	Ours
1	97.04 ±1.78	87.36 ±3.16	82.46 ±4.46	99.23 ±0.23	83.89 ±5.14	94.99 ±1.10	74.17 ±6.48	99.51 ±0.15	99.41 ±0.18	**100.00 ±0.00**
2	24.74 ±8.40	37.63 ±7.21	72.54 ±5.78	91.14 ±1.32	83.26 ±3.67	89.13 ±2.35	**100.00 ±0.00**	90.98 ±2.76	98.22 ±0.72	98.71 ±0.24
3	14.15 ±6.13	79.02 ±3.44	78.21 ±4.03	97.45 ±0.63	**99.19 ±0.23**	97.93 ±0.97	91.46 ±3.01	96.10 ±1.06	84.47 ±2.33	96.59 ±0.84
4	18.32 ±7.12	23.76 ±6.75	56.83 ±6.38	93.28 ±1.08	87.60 ±3.38	**96.64 ±0.88**	93.39 ±2.74	83.91 ±2.54	71.02 ±3.41	83.91 ±1.35
5	6.98 ±4.28	16.28 ±5.43	93.58 ±1.76	99.11 ±0.26	**100.00 ±0.00**	80.13 ±4.90	**100.00 ±0.00**	99.22 ±0.24	82.71 ±3.01	**100.00 ±0.00**
6	0.55 ±0.21	12.02 ±3.98	57.86 ±5.24	97.43 ±0.35	**100.00 ±0.00**	92.17 ±1.53	92.24 ±1.98	99.18 ±0.12	86.60 ±2.78	**100.00 ±0.00**
7	2.38 ±1.12	78.57 ±2.93	42.32 ±5.99	88.40 ±1.97	**100.00 ±0.00**	**100.00 ±0.00**	**100.00 ±0.00**	46.43 ±6.22	88.30 ±2.78	91.67 ±0.99
8	88.41 ±2.87	75.07 ±4.37	72.43 ±4.56	99.12 ±0.16	90.26 ±2.01	96.03 ±1.04	98.57 ±0.63	**99.28 ±0.09**	96.79 ±0.76	99.13 ±0.10
9	3.85 ±2.01	71.63 ±3.88	54.85 ±5.34	96.33 ±0.62	99.22 ±0.24	**99.79 ±0.09**	96.86 ±1.36	91.47 ±2.20	99.75 ±0.12	99.52 ±0.14
10	50.15 ±4.84	60.06 ±4.78	43.76 ±6.65	93.47 ±1.03	93.91 ±1.63	99.47 ±0.28	99.75 ±0.22	97.83 ±0.81	99.85 ±0.08	**100.00 ±0.00**
11	99.10 ±0.50	91.64 ±1.77	68.78 ±4.61	99.20 ±0.15	**100.00 ±0.00**	94.59 ±1.23	99.27 ±0.31	99.40 ±0.12	**100.00 ±0.00**	**100.00 ±0.00**
12	65.17 ±5.05	52.24 ±5.98	72.13 ±3.88	97.28 ±0.63	99.80 ±0.10	**100.00 ±0.00**	97.16 ±1.09	97.76 ±0.53	99.50 ±0.15	99.63 ±0.07
13	94.47 ±1.38	96.90 ±0.76	75.52 ±3.48	**100.00 ±0.00**	**100.00 ±0.00**	**100.00 ±0.00**	**100.00 ±0.00**	**100.00 ±0.00**	**100.00 ±0.00**	**100.00 ±0.00**
OA(%)	59.85 ±1.70	69.82 ±1.59	68.58 ±0.39	97.28 ±0.68	94.84 ±0.92	96.65 ±0.40	94.33 ±1.70	96.05 ±2.34	96.04 ±1.20	**98.66 ±0.75**
AA(%)	43.49 ±1.82	60.17 ±1.64	66.97 ±0.52	95.94 ±0.75	95.16 ±1.04	96.70 ±0.50	95.61 ±1.82	92.39 ±2.64	92.84 ±1.32	**97.60 ±0.84**
κ(%)	54.52 ±1.87	66.11 ±1.77	67.92 ±0.60	96.97 ±0.81	94.26 ±1.22	96.28 ±0.65	93.71 ±1.86	95.60 ±2.76	95.60 ±1.40	**98.50 ±0.94**

sparse sample distribution. Our large kernel convolutional model is able to extract long-distance spatial features, highlighting global information extraction, and focusing effectively on sparse samples with large receptive field, resulting in better classification performance. Similarly, existing methods generally suffer from insufficient classification stability on the KSC dataset. In contrast, the standard deviation of our method is less than ±1.0 in 9 out of 13 categories, and the standard deviations of categories 1, 5, 6, 11, and 13 are close to zero, indicating that its classification results are robust to random initialization and training set perturbations. This stability advantage can be attributed to the spatial context consistency constraint provided by the large-scale receptive field and the adaptive calibration of the attention mechanism to the heterogeneity of spectral channels.

As shown in Table VI and Fig. 6, the IP dataset is characterized by uneven distribution of labeled samples, with some categories having fewer samples. The LSF loss function and AGF classifier in our method effectively address this imbalance, maximizing interclass variation. As a result, our method achieves outstanding classification performance, with OA, OA, and κ scores exceeding 99%, 98%, and 0.99, respectively, the highest among all compared methods. In addition, classification accuracies of all classes are above 90% . The AGF classifier's vector normalization ensures a balanced distribution of category accuracies, improving the overall stability and reliability. Other SOTA methods improve the classification accuracy of the IP dataset by increasing the spatial correlation between adjacent pixels relative to SSRN, thereby achieving better feature learning. However, they failed to fully address uneven accuracy distribution across categories, resulting in misclassifications in certain classes. The proposed method also shows significant classification stability on the IP dataset, with standard deviations of 14 out of 16 ground feature categories being less than ±0.50. It is worth noting that despite the small number of oat samples in category 9, the proposed method still controls its standard deviation at ±0.83%, which is significantly lower than that of SSRN and HybridSN.

On the PU dataset, our method maintains the highest classification accuracy, particularly in accuracy per class, with six classes achieving 100.00% accuracy and all classification accuracies above 95.00%, as shown in Table VII. Qualitatively, our method obtains clear contours and accurate spatial distributions in the image while suppressing salt and pepper noise. It preserves object continuity without blurring boundaries, demonstrating the ability of the attention mechanism to extract discriminative features at different granularities. This results in classification maps with fine local details. What is more, our model and CLOLN model are different in structure, but both of them effectively improve the classification accuracy through multiscale feature extraction and attention mechanism. CLOLN uses dual single-channel 3-D convolution and channel attention mechanism, and our method achieves a good balance in the extraction of global and local information through large kernel convolution and MSCA attention mechanism. Therefore, both methods achieve similar

TABLE VI
QUANTITATIVE RESULTS FOR IP DATASET

Class	2D-CNN	3D-CNN	SSRN	HybridSN	CSIL	OSICN	HSIC-FM	HSIC-SDM	CLOLN	Ours
1	64.85 ±8.81	82.46 ±2.46	88.45 ±0.38	85.82 ±0.76	**100.00 ±0.00**	**100.00 ±0.00**	98.65 ±2.45	**100.00 ±0.00**	98.90 ±0.15	99.22 **±0.10**
2	80.16 ±5.11	87.52 ±2.25	97.83 ±1.24	95.48 ±1.91	97.01 ±0.45	93.75 ±1.21	82.59 ±2.28	86.13 ±2.85	97.40 ±0.32	**98.85 ±0.18**
3	84.62 ±11.87	92.25 ±2.21	95.26 ±0.57	97.57 ±0.54	90.81 ±1.84	98.82 ±0.26	99.32 ±2.19	98.64 ±2.65	95.61 ±0.54	**99.91 ±0.06**
4	76.87 ±2.98	85.12 ±1.78	94.74 ±0.12	96.13 ±0.23	**100.00 ±0.00**	**100.00 ±0.00**	**100.00 ±0.00**	**100.00 ±0.00**	93.31 ±0.65	**100.00 ±0.00**
5	82.14 ±2.65	75.58 ±2.18	92.94 ±0.64	94.62 ±0.35	97.71 ±0.38	99.55 ±0.17	99.61 ±2.21	**99.87** ±1.47	96.60 ±0.44	99.26 ±0.23
6	89.52 ±0.78	86.12 ±2.23	98.53 ±0.11	99.06 ±0.26	**100.00 ±0.00**	94.44 ±0.90	99.66 ±0.24	99.40 ±0.22	98.12 ±0.25	99.32 ±0.15
7	85.38 ±5.88	98.84 ±0.28	98.24 ±0.21	94.72 ±1.83	94.44 ±1.48	99.24 ±0.28	**100.00 ±0.00**	**100.00 ±0.00**	95.34 ±0.78	98.75 ±0.21
8	**100.00 ±0.00**	93.46 ±0.12	**100.00 ±0.00**	94.85 ±0.88	**100.00 ±0.00**	**100.00 ±0.00**	99.48 ±8.53	98.30 ±2.53	**100.00 ±0.00**	99.85 ±0.05
9	58.04 ±9.92	62.12 ±8.42	84.52 ±1.81	88.28 ±0.11	**100.00 ±0.00**	**100.00 ±0.00**	**100.00 ±0.00**	93.75 ±1.84	90.00 ±1.29	92.86 ±0.83
10	85.48 ±2.43	79.48 ±1.53	97.38 ±1.08	88.58 ±1.72	95.35 ±0.66	**99.88 ±0.09**	82.26 ±2.96	83.87 ±1.58	89.98 ±0.45	99.45 ±0.13
11	92.68 ±4.11	95.72 ±2.14	98.72 ±1.52	97.76 ±1.26	95.77 ±0.58	91.96 ±1.81	98.40 ±1.95	97.84 ±1.02	96.94 ±0.37	**99.56 ±0.09**
12	67.92 ±7.23	97.62 ±1.95	94.32 ±1.09	96.26 ±0.98	96.62 ±0.43	**99.22 ±0.22**	97.47 ±2.01	96.21 ±1.65	90.71 ±0.92	98.49 ±0.37
13	85.94 ±1.06	97.34 ±0.49	98.92 ±0.21	95.32 ±0.05	**100.00 ±0.00**	99.26 ±0.18	**100.00 ±0.00**	**100.00 ±0.00**	99.51 ±0.12	98.78 ±0.18
14	95.86 ±1.02	97.82 ±1.98	99.23 ±0.09	99.16 ±0.27	98.72 ±0.25	99.83 ±0.08	99.95 ±0.39	99.31 ±0.02	**100.00 ±0.00**	**100.00 ±0.00**
15	89.48 ±4.75	87.42 ±1.56	95.27 ±0.24	90.86 ±0.11	99.42 ±0.12	**100.00 ±0.00**	99.68 ±0.56	**100.00 ±0.00**	99.87 ±0.06	99.91 ±0.04
16	87.68 ±10.02	92.84 ±0.67	96.63 ±0.90	97.32 ±0.30	**100.00 ±0.00**	**100.00 ±0.00**	96.62 ±1.79	97.30 ±0.39	99.47 ±0.10	96.54 ±0.52
OA(%)	83.48 ±1.45	92.14 ±1.64	96.14 ±0.39	95.74 ±0.34	96.80 ±0.20	96.83 ±0.27	95.18 ±0.78	95.44 ±0.62	96.51 ±0.78	**99.42 ±0.08**
AA(%)	82.91 ±1.98	88.23 ±1.53	94.49 ±0.61	94.49 ±0.52	97.87 ±0.52	96.12 ±0.49	97.11 ±1.42	96.91 ±0.68	90.42 ±1.25	**98.80 ±0.28**
κ(%)	79.67 ±1.39	88.73 ±1.88	95.12 ±0.70	94.86 ±0.88	96.31 ±0.59	96.37 ±0.55	94.50 ±1.60	94.79 ±0.75	97.26 ±1.30	**99.34 ±0.43**

TABLE VII
QUANTITATIVE RESULTS FOR PU DATASET

Class	2D-CNN	3D-CNN	SSRN	HybridSN	CSIL	OSICN	HSIC-FM	HSIC-SDM	CLOLN	Ours
1	97.67 ±1.54	96.54 ±0.88	95.64 ±1.19	99.34 ±0.24	94.00 ±1.37	88.12 ±1.23	87.40 ±1.65	99.77 ±0.18	99.77 ±0.12	**100.00 ±0.00**
2	98.52 ±0.78	98.84 ±0.56	98.28 ±0.97	**100.00 ±0.00**	95.80 ±1.18	91.70 ±1.56	89.97 ±2.01	99.99 ±0.08	99.94 ±0.15	**100.00 ±0.00**
3	85.36 ±2.49	88.92 ±1.84	87.85 ±2.01	99.48 ±0.31	98.54 ±0.73	95.46 ±1.02	84.29 ±2.98	90.93 ±1.58	99.55 ±0.23	**100.00 ±0.00**
4	96.82 ±1.18	94.68 ±1.63	93.61 ±1.75	98.32 ±0.45	95.65 ±0.89	98.74 ±0.66	99.04 ±0.78	94.58 ±1.26	98.38 ±0.32	**99.43 ±0.11**
5	99.42 ±0.34	99.52 ±0.27	99.32 ±0.34	99.41 ±0.17	**100.00 ±0.00**	99.07 ±0.42	99.85 ±0.15	99.92 ±0.10	99.48 ±0.23	**100.00 ±0.00**
6	98.32 ±0.58	99.12 ±0.33	98.92 ±0.38	**100.00 ±0.00**	99.22 ±0.27	95.74 ±0.98	92.65 ±1.38	**100.00 ±0.00**	**100.00 ±0.00**	**100.00 ±0.00**
7	93.74 ±1.76	91.53 ±2.09	87.67 ±2.38	97.32 ±0.68	**100.00 ±0.00**	98.44 ±0.73	96.48 ±1.16	98.30 ±0.54	99.85 ±0.09	99.48 ±0.17
8	95.64 ±1.13	92.35 ±1.52	90.26 ±1.87	99.47 ±0.30	91.00 ±1.47	97.14 ±0.86	94.85 ±1.13	98.41 ±0.37	99.27 ±0.27	**100.00 ±0.00**
9	88.68 ±2.71	96.36 ±0.64	93.65 ±1.20	96.12 ±0.62	99.67 ±0.28	**100.00 ±0.00**	**100.00 ±0.00**	92.50 ±1.82	97.35 ±0.44	97.56 ±0.31
OA(%)	95.63 ±1.50	96.98 ±0.85	95.82 ±1.28	99.22 ±0.26	95.97 ±1.34	93.37 ±1.23	91.39 ±1.60	98.76 ±0.18	99.66 ±0.12	**99.79 ±0.16**
AA(%)	94.91 ±1.84	95.32 ±1.12	93.91 ±1.57	98.52 ±0.43	97.10 ±0.95	93.30 ±0.98	93.84 ±3.44	97.15 ±0.35	99.19 ±0.25	**99.28 ±0.18**
κ(%)	95.14 ±1.72	94.25 ±1.05	94.23 ±1.48	98.96 ±0.38	94.68 ±1.64	91.32 ±1.45	88.74 ±2.04	98.36 ±0.22	99.66 ±0.09	**99.72 ±0.12**

TABLE VIII
QUANTITATIVE RESULTS FOR LK DATASET

Class	2D-CNN	3D-CNN	SSRN	HybridSN	CSIL	OSICN	HSIC-FM	HSIC-SDM	CLOLN	Ours
1	97.78 ±3.28	92.88 ±2.61	99.61 ±0.17	94.20 ±1.83	99.12 ±0.22	99.34 ±0.24	75.92 ±6.48	**99.95 ±0.05**	98.09 ±0.38	99.83 ±0.08
2	97.29 ±1.82	90.51 ±3.02	72.35 ±4.87	98.54 ±0.67	97.33 ±0.74	98.59 ±0.38	76.71 ±5.14	99.53 ±0.15	96.38 ±0.91	**99.95 ±0.03**
3	**100.00 ±0.00**	99.93 ±0.05	53.16 ±7.38	95.64 ±1.46	**100.00 ±0.00**	98.55 ±0.42	71.22 ±6.88	**100.00 ±0.00**	**100.00 ±0.00**	99.58 ±0.12
4	83.80 ±4.34	77.78 ±4.89	96.37 ±1.27	60.41 ±6.09	92.96 ±1.83	91.43 ±2.12	77.52 ±4.39	96.17 ±1.05	96.16 ±1.16	**99.75 ±0.12**
5	97.01 ±1.18	88.13 ±2.81	60.94 ±5.50	89.69 ±3.22	**100.00 ±0.00**	95.66 ±1.63	87.29 ±3.25	99.97 ±0.06	88.14 ±2.12	**100.00 ±0.00**
6	2.36 ±1.55	88.65 ±3.54	93.16 ±1.76	79.88 ±4.24	97.31 ±0.61	99.27 ±0.19	80.94 ±4.17	97.35 ±0.73	**99.89 ±0.07**	98.67 ±0.27
7	65.95 ±6.04	95.14 ±1.16	99.89 ±0.04	97.58 ±0.76	99.46 ±0.21	99.09 ±0.36	96.39 ±1.72	97.17 ±0.62	99.11 ±0.24	**99.96 ±0.03**
8	97.18 ±1.62	93.57 ±1.97	78.51 ±4.12	85.25 ±3.72	94.27 ±1.38	**98.52 ±0.41**	67.16 ±6.00	91.77 ±1.83	86.10 ±3.21	97.03 ±0.70
9	89.14 ±2.87	93.86 ±1.52	90.24 ±2.38	85.88 ±2.66	91.48 ±1.45	**98.26 ±0.58**	90.31 ±2.76	96.40 ±0.99	95.91 ±0.82	95.01 ±1.12
OA(%)	77.25 ±10.19	88.67 ±1.44	91.95 ±0.96	83.62 ±2.84	96.82 ±0.76	96.93 ±0.68	83.67 ±2.53	97.33 ±0.40	97.16 ±0.59	**99.56 ±0.41**
AA(%)	81.17 ±8.72	91.16 ±1.22	74.65 ±3.16	87.45 ±2.36	96.88 ±0.65	97.73 ±0.63	80.38 ±3.08	97.59 ±0.33	96.25 ±0.71	**98.86 ±0.46**
κ(%)	71.07 ±9.86	85.52 ±1.30	89.27 ±1.12	79.44 ±3.06	95.85 ±0.84	95.99 ±0.75	79.15 ±2.89	96.52 ±0.45	95.52 ±0.66	**99.42 ±0.44**

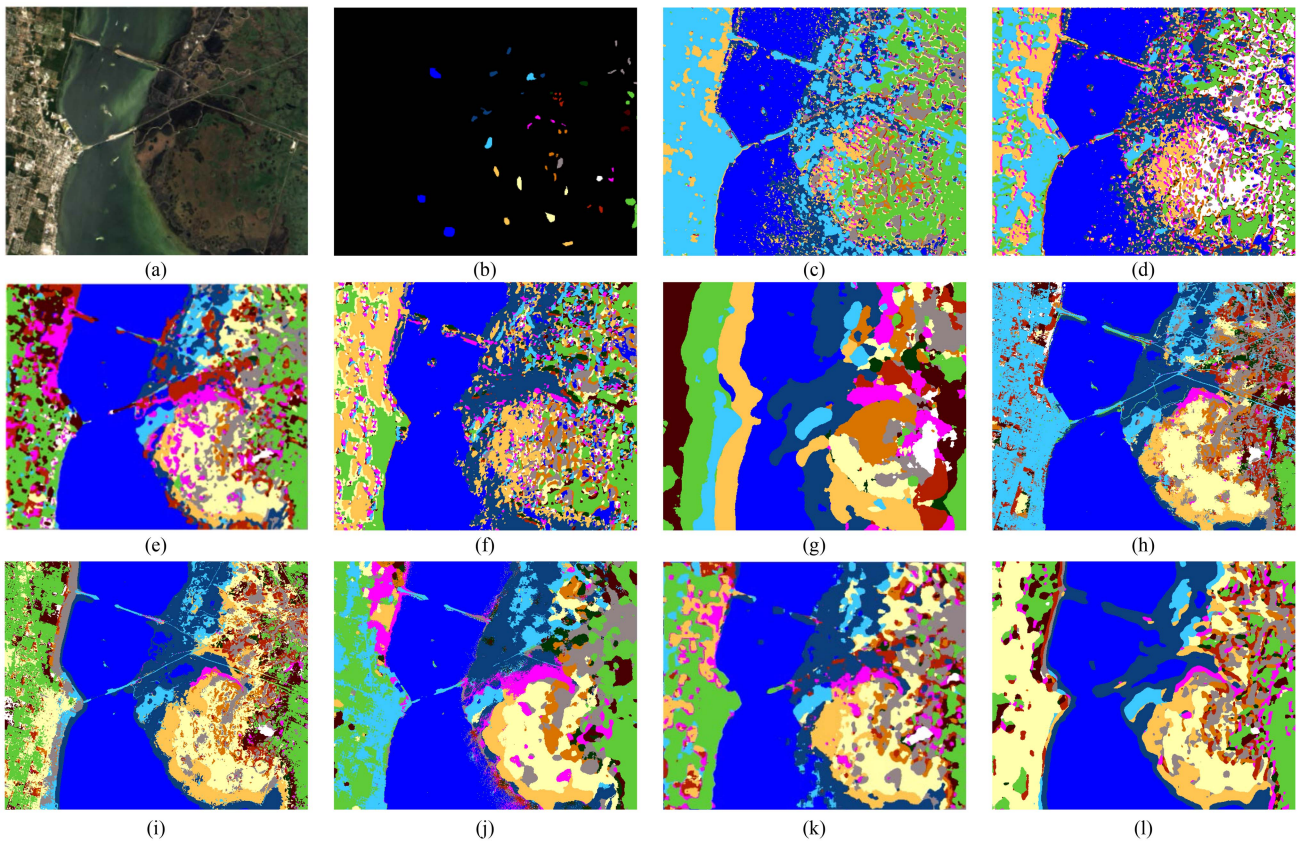


Fig. 5. Classification maps for the KSC dataset. (a) False-color image. (b) Ground-truth map. (c) 2D-CNN. (d) 3D-CNN. (e) SSRN. (f) HybridSN. (g) CSIL. (h) OSICN. (i) HSIC-FM. (j) HSIC-SDM. (k) CLONLN. (l) Ours.

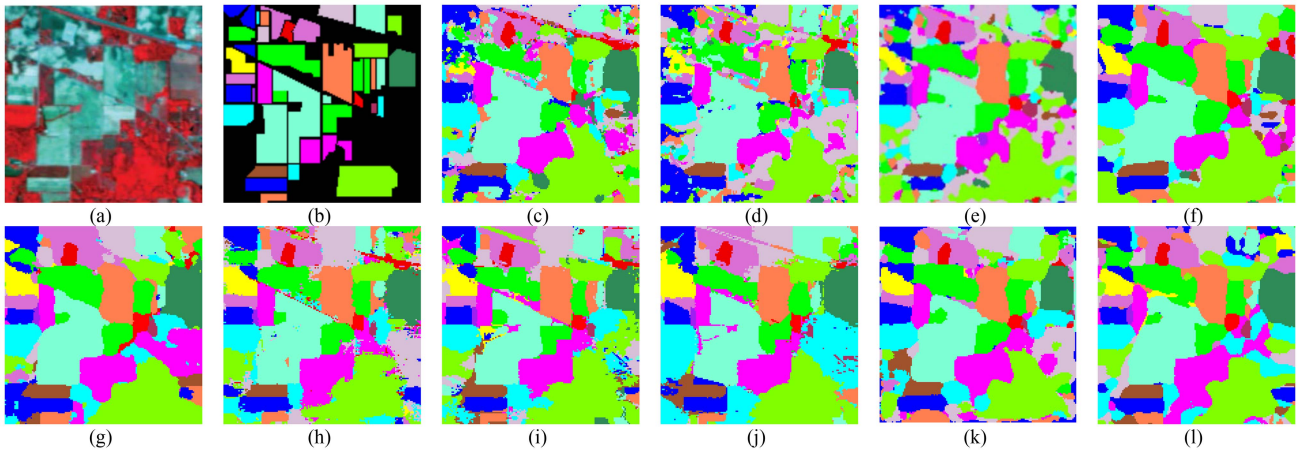


Fig. 6. Classification maps for the IP dataset. (a) False-color image. (b) Ground-truth map. (c) 2D-CNN. (d) 3D-CNN. (e) SSRN. (f) HybridSN. (g) CSIL. (h) OSICN. (i) HSIC-FM. (j) HSIC-SDM. (k) CLONLN. (l) Ours.

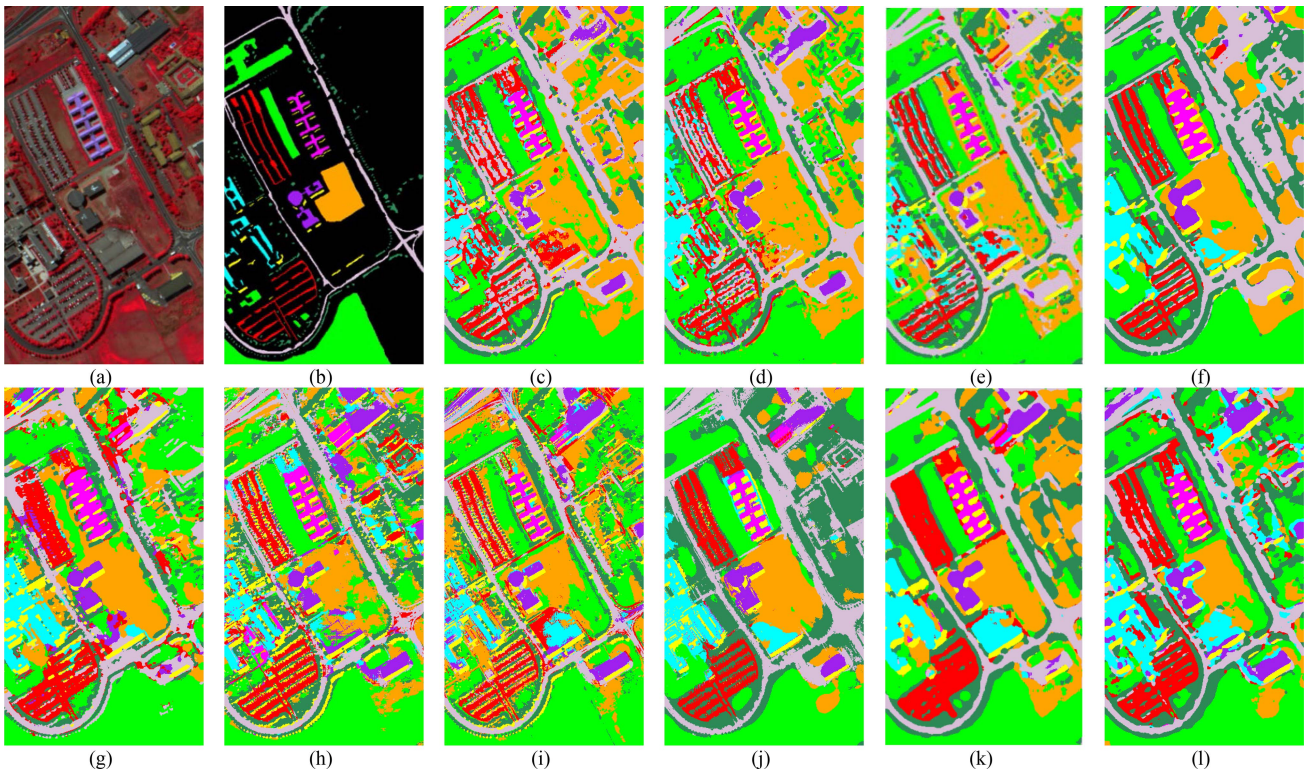


Fig. 7. Classification maps for the PU dataset. (a) False-color image. (b) Ground-truth map. (c) 2D-CNN. (d) 3D-CNN. (e) SSRN. (f) HybridSN. (g) CSIL. (h) OSICN. (i) HSIC-FM. (j) HSIC-SDM. (k) CLONLN. (l) Ours.

classification accuracy on the PU dataset. However, Hyper-LKCnet further improves generalization by introducing AGF classifier and LSF loss function, so it achieves higher classification accuracy while maintaining a similar classification accuracy distribution.

The experimental results on the LK dataset that the training sample is only 0.15% of the labeled sample further demonstrate our model's excellence. Our method outperforms others by 2.23% in OA, achieving 99.56%, compared with the second-best OA of 97.33% . It also proves superior in AA, gaining balanced performance across all classes. Although OSICN performs well (AA of 97.73%), it is still outperformed by our model. Other methods produce many misclassifications and noise when classifying sesame and narrow-leaf soybeans, making it difficult to distinguish between similar land covers with fewer training samples. The above phenomenon indicates that the discriminative information learned by these models is insufficient for the accurate classification.

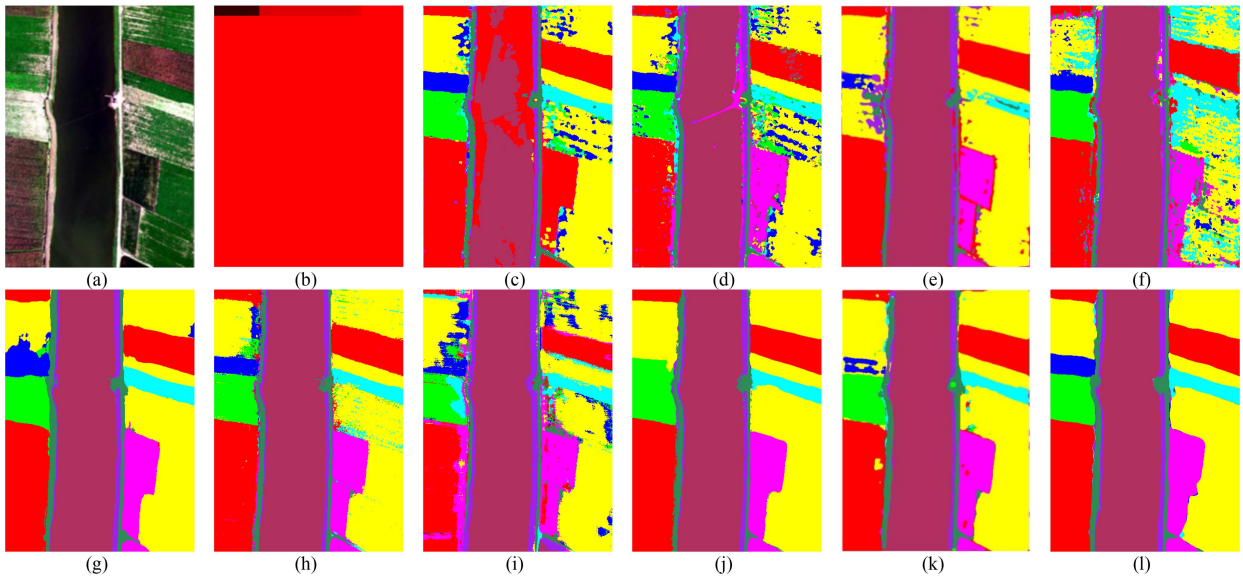


Fig. 8. Classification maps for the LK dataset. (a) False-color image. (b) Ground-truth map. (c) 2D-CNN. (d) 3D-CNN. (e) SSRN. (f) HybridSN. (g) CSIL. (h) OSICN. (i) HSIC-FM. (j) HSIC-SDM. (k) CLONLN. (l) Ours.

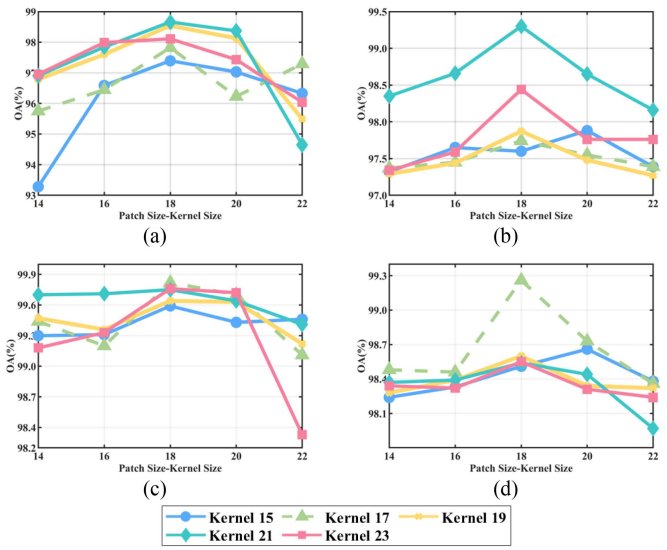


Fig. 9. Influence of patch size and kernel size. (a) KSC. (b) IP. (c) PU. (d) LK. The horizontal axis represents the difference in size after the kernel size is subtracted from the patch size, while the vertical axis represents classification accuracy (OA).

On both PU and LK datasets, our method shows excellent classification stability. It has obvious advantages for categories with complex spectral features and categories with more noise interference.

In summary, our method demonstrates the advantages across all the datasets, with robust classification performance in both accuracy and visualization. The OA on all four datasets exceeds 98.00%, with fewer training samples. Notably, our method excels in handling sample imbalance, with a significant improvement in classification accuracy. For example, on the challenging KSC and IP datasets, our approach outperforms the existing SOTA methods by at least 2.00%.

TABLE IX
RESULTS OF ABLATION EXPERIMENTS

Dataset	Large Kernel	MSCA	LSF loss	AGF	OA (%)	AA (%)	κ (%)
KSC	√	-	-	-	88.13	74.02	86.68
	√	√	-	-	94.48	91.25	93.84
	√	√	√	-	96.97	96.17	96.63
	√	√	-	√	97.96	97.42	97.73
	√	-	√	√	96.33	95.12	95.91
	√	√	√	√	**98.66**	**97.60**	**98.50**
IP	√	-	-	-	89.36	82.69	87.81
	√	√	-	-	94.85	96.45	94.12
	√	√	√	-	95.27	96.56	94.60
	√	√	-	√	97.73	96.94	97.41
	√	-	√	√	96.52	97.70	96.03
	√	√	√	√	**99.42**	**98.80**	**99.34**
PU	√	-	-	-	87.48	87.98	83.73
	√	√	-	-	91.82	91.96	89.22
	√	√	√	-	96.88	96.93	95.89
	√	√	-	√	97.51	96.65	96.68
	√	-	√	√	93.69	93.83	91.71
	√	√	√	√	**99.79**	**99.28**	**99.72**
LK	√	-	-	-	84.89	88.38	80.70
	√	√	-	-	90.94	94.94	88.41
	√	√	√	-	96.91	97.48	95.96
	√	√	-	√	98.44	98.64	97.96
	√	-	√	√	94.58	96.08	92.99
	√	√	√	√	**99.56**	**98.86**	**99.42**

D. Ablation Study

In this section, we evaluate the impact of each proposed component in the Hyper-LKCNet framework on the classification performance of four datasets.

The experimental results are presented in Table IX. Bold values highlight the best performance across different configurations in the ablation study.MSCA specifies that the Multi-Scale Co-Attention (MSCA) mechanism is applied. If MSCA is not used, a standard convolutional layer without attention is

TABLE X
MODEL COMPLEXITY ANALYSIS ON FOUR DATASETS

Method	Metric	SSRN	HybridSN	CSIL	HSIC-FM	OSICN	Ours
KSC	Params (M)	**2.48**	4.84	112.74	16.91	105.26	4.68
	FLOPs (M)	146.50	9.68	3.71×10^4	1.05×10^5	56.87	**9.35**
	OA (%)	68.58	97.28	94.84	94.33	96.65	**98.66**
IP	Params (M)	7.51	**5.12**	112.65	17.69	105.38	14.98
	FLOPs (M)	634.25	**10.24**	3.70×10^4	8.91×10^5	57.02	29.95
	OA (%)	96.14	95.74	96.80	95.18	96.83	**99.42**
PU	Params (M)	**1.85**	4.84	108.25	14.50	104.97	2.44
	FLOPs (M)	143.34	9.69	3.71×10^4	3.61×10^5	56.45	**4.88**
	OA (%)	95.82	99.22	95.97	91.39	93.37	**99.79**
LK	Params (M)	**3.93**	4.94	105.26	19.99	105.57	4.68
	FLOPs (M)	227.32	9.87	3.70×10^4	3.64×10^5	57.40	**9.35**
	OA (%)	91.95	83.62	96.82	83.67	96.93	**99.56**

employed as an alternative. LSF loss indicates that the LSF loss function is used; when LSF loss is not used, the standard CE loss function is applied. AGF denotes the use of the AGF classifier; if AGF is not used, a fully connected layer classifier is used as a substitute.

The experimental results demonstrate the effectiveness of our attention mechanism. For example, on the KSC dataset, the MSCA mechanism boosts the OA to 94.48%, an improvement of 6.35% over the original large kernel convolution. On the imbalanced IP dataset, the combination of the LSF loss function and AGF classifier significantly improves results, with AA reaching 97.70%, highlighting the structure's ability to handle sample imbalance. When all three modules are used together, optimal classification results are achieved. Similar improvements are observed across other datasets, confirming the generalizability of our method.

IV. DISCUSSION

A. Analysis of Model Complexity

To assess the complexity and efficiency of the proposed Hyper-LKCNet, we compare three performance metrics, trainable parameters and FLOPs, against SSRN, HybridSN, CSIL, HSIC-FM, and OSICN. We also evaluate the model's efficiency in relation to its OA.

The results in Table X show that our model generally has fewer parameters than other methods, such as 4.68M on KSC and 2.44M on PU, which is significantly lower than methods, such as CSIL (which has over 100M parameters). Despite having fewer training parameters, our model achieves the best classification results, proving that the optimization of large kernel convolution using depth-separable convolution instead of dense convolution is effective. Although SSRN has fewer parameters, our model achieves significantly superior classification accuracy with lower FLOPs. In terms of FLOPs, our model maintains the lowest FLOPs on the KSC, PU, and LK datasets, with the only suboptimal performance on the IP dataset. These results confirm that our method not only demonstrates excellent generalization ability and classification performance on imbalanced HSI datasets but also efficiently utilizes computational resources, which broadens its potential applications.

Based on 3-D convolution, in order to reduce the computational complexity, we use depthwise-separable convolution to replace traditional convolution. Depthwise-separable convolution decomposes the standard convolution operation into two stages: first, convolution is performed on each input channel, and then convolution is performed on all output channels. This operation greatly reduces the amount of computation and the number of parameters, effectively reducing complexity while maintaining high accuracy.

In addition, the AGF classifier and LFS loss function also help improve the computational efficiency of the model. The LFS loss function avoids overcomputation of easy-to-classify samples by focusing on difficult-to-classify samples, thereby reducing the amount of calculation. The AGF classifier can improve computational efficiency by optimizing the geometric configuration of the feature space while ensuring classification accuracy, further reducing computational complexity.

B. Influences of the Patch Size and the Kernel Size

The key parameters in our experiments include patch size and kernel size (kernel size refers to the size of the large kernel used in depth-separable convolution). The input kernel size impacts the receptive field of the convolution, reflecting the model's ability to capture global spatial features, while the patch size determines the slice scale of the input image, affecting the overall feature extraction. To explore the relationship between these parameters, we examined the effects of different patch sizes and kernel size combinations. The results are shown in Fig. 9.

Across the four datasets, it can be seen that OA generally increases and then decreases as the difference between patch size and kernel size increases. Initially, larger patch sizes provide more spatial information, leading to an increase in OA. As the patch size continues to increase, it is more likely to introduce noise or capture edge regions, leading to oversmoothing and a decline in OA. In addition, larger patch sizes require more memory and computational resources. We also observe that different datasets benefit from different large kernel sizes, especially the IP dataset, where a kernel size of 21×21 significantly outperforms other sizes in terms of OA.

Based on the above analysis, choosing the appropriate patch size and kernel size is crucial for optimal performance. In this study, we set the patch size to 39×39 and kernel size to 21×21 for the KSC, IP, and PU datasets, and the patch size to 35×35 and kernel size to 17×17 for the LK dataset.

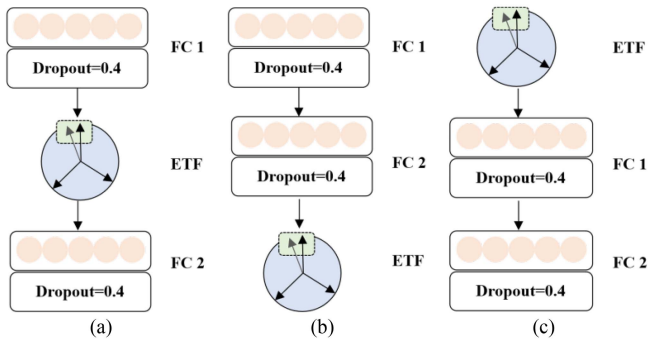


Fig. 10. Three strategies for ETF positions. (a) Strategy 1. (b) Strategy 2. (c) Strategy 3.

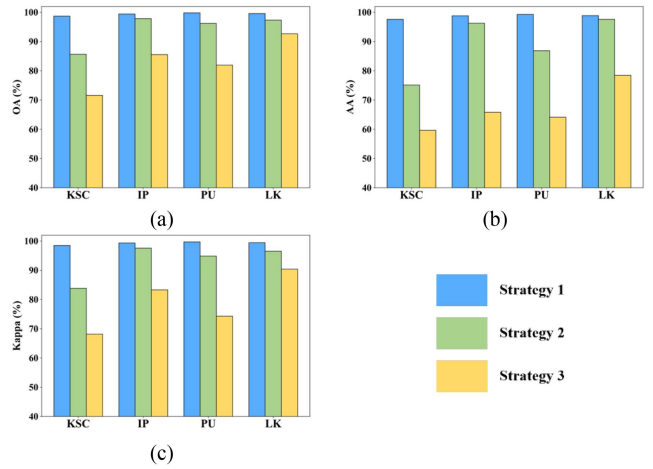


Fig. 11. Results of different strategies for four datasets.

C. Position of the ETF Classifier

The AGF classifier in our framework is a tandem combination of fully connected layers and an ETF classifier, using three strategies, as shown in Fig. 10. We evaluated the performance of the ETF classifiers by placing them in different positions within the AGF framework. For example, in strategy 1, we introduce the ETF classifier after the first fully connected layer, followed by another fully connected layer to achieve feature vector normalization.

The experimental results, as shown in Fig. 11, indicate that strategy 1 outperforms the others across all datasets. In a simple single fully connected layer design, the expression of features may be too smooth, resulting in the loss of detailed information, especially when processing high-dimensional data. The introduction of the second fully connected layer helps the model further adjust the feature space to avoid oversmoothing and retain more discriminative information, thereby improving classification accuracy. This demonstrates that the ETF classifier effectively learns representative and separable features from the first fully connected layer, enhancing feature differentiation through orthogonal transformations and isotropic variance processing. The second fully connected layer then activates more contextual information and features. While strategy 2 is slightly less effective than strategy 1, this may be because placing the ETF after two fully connected layers causes some important features to be lost or distorted, making it difficult for the ETF to extract effective feature information. Strategy 3 performs the worst because the ETF classifier cannot take full advantage of the fully connected layer for feature combination and nonlinear mapping. Placing the ETF classifier at the front makes it difficult to extract useful information from features that have not been fully integrated and transformed. It receives incomplete or inaccurate information from the previous stages of the fully connected layer when generating feature vectors, causing the ETF graphs to be offset and the feature vectors to be misaligned, leading to degraded classification performance. The position of ETF has the greatest impact on AA, with a reduction in OA directly affecting the stability and accuracy of AA.

V. Conclusion

In this article, we propose Hyper-LKCNet for HSI classification, which explores the potential of large kernel convolution of HSI and defines a paradigm to fully utilize the advantages of large kernel convolution. The proposed approach can promote global and local feature extraction, obtaining better outcome compared with advanced CNN and transformer algorithms. In detail, the standard convolution operation is decomposed into depth convolution and point-by-point convolution using depth-separable large kernel convolution to reduce parameters and computation while maintaining efficiency. To accommodate the unique characteristics of HSIs, such as limited labels and numerous bands, we introduce a hybrid attention mechanism to simultaneously capture spatial and spectral features effectively. In addition, sample imbalance is addressed with the AGF classifier, inspired by neural collapse, which improves classification accuracy. The integration of LSF loss further enhances model generalization, yielding strong results across four datasets.

In the future, we aim to expand kernel sizes to fully exploit the advantages of large receptive fields. In addition, our focus will extend beyond single-scene classification of hyperspectral data, with interest in domain generalization [76], [77] for cross-scene classification.

References

[1] L. I. Lebedev, Y. V. Yasakov, T. H. Utesheva, V. P. Gromov, A. V. Borusjak, and V. Turlapov, "Complex analysis and monitoring of the environment based on Earth sensing data," *Comput. Opt.*, vol. 43, no. 2, pp. 282–295, 2019.

[2] X. Kang, Z. Wang, P. Duan, and X. Wei, "The potential of hyperspectral image classification for oil spill mapping," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 9, Sep. 2022, Art. no. 5538415.

[3] X. Chen, X. Zheng, Y. Zhang, and X. Lu, "Remote sensing scene classification by local–global mutual learning," *IEEE Geosci. Remote Sens. Lett.*, vol. 19, Feb. 2022, Art. no. 6506405.

[4] M. O. Yusuf, D. Srivastava, and R. Kushwaha, "Resolution invariant urban scene classification using multiview learning paradigm," *Digit. Signal Process.*, vol. 139, 2023, Art. no. 104078.

[5] J. Deng et al., "Quantitative estimation of wheat stripe rust disease index using unmanned aerial vehicle hyperspectral imagery and innovative vegetation indices," *IEEE Trans. Geosci. Remote Sens.*, vol. 61, Jul. 2023, Art. no. 4406111.

[6] Y. Tian, Y. Wang, D. Krishnan, J. B. Tenenbaum, and P. Isola, "Rethinking few-shot image classification: A good embedding is all you need?," in *Proc. 16th Eur. Conf. Comput. Vis.*, Glasgow, U.K., 2020, pp. 266–282.

[7] X. Jiang, L. Xiong, Q. Yan, Y. Zhang, X. Liu, and Z. Cai, "Unsupervised dimensionality reduction for hyperspectral imagery via Laplacian regularized collaborative representation projection," *IEEE Geosci. Remote Sens. Lett.*, vol. 19, Feb. 2022, Art. no. 6007805.

[8] X. Lu, W. Zhang, and X. Li, "A hybrid sparsity and distance-based discrimination detector for hyperspectral images," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 3, pp. 1704–1717, Mar. 2017.

[9] I. Petromichelakis and I. A. Kougioumtzoglou, "Addressing the curse of dimensionality in stochastic dynamics: A Wiener path integral variational formulation with free boundaries," *Proc. Roy. Soc. A*, vol. 476, no. 2243, 2020, Art. no. 20200385.

[10] Y. Zhang, X. Zheng, and X. Lu, "Pairwise comparison network for remote-sensing scene classification," *IEEE Geosci. Remote Sens. Lett.*, vol. 19, Dec. 2021, Art. no. 6505105.

[11] L. Samaniego, A. Bárdossy, and K. Schulz, "Supervised classification of remotely sensed imagery using a modified k-NN technique," *IEEE Trans. Geosci. Remote Sens.*, vol. 46, no. 7, pp. 2112–2125, Jul. 2008.

[12] J. Ediriwickrema and S. Khorram, "Hierarchical maximum-likelihood classification for improved accuracies," *IEEE Trans. Geosci. Remote Sens.*, vol. 35, no. 4, pp. 810–816, Jul. 1997.

[13] S. Kumar, J. Ghosh, and M. M. Crawford, "Best-bases feature extraction algorithms for classification of hyperspectral data," *IEEE Trans. Geosci. Remote Sens.*, vol. 39, no. 7, pp. 1368–1379, Jul. 2001.

[14] C.-I. Chang and S. Wang, "Constrained band selection for hyperspectral imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 44, no. 6, pp. 1575–1585, Jun. 2006.

[15] L. M. Bruce, C. H. Koger, and J. Li, "Dimensionality reduction of hyperspectral data using discrete wavelet transform feature extraction," *IEEE Trans. Geosci. Remote Sens.*, vol. 40, no. 10, pp. 2331–2338, Oct. 2002.

[16] T. V. Bandos, L. Bruzzone, and G. Camps-Valls, "Classification of hyperspectral images with regularized linear discriminant analysis," *IEEE Trans. Geosci. Remote Sens.*, vol. 47, no. 3, pp. 862–873, Mar. 2009.

[17] X. Chen, Y. Zhang, Y. Dong, and B. Du, "Spatial-spectral contrastive self-supervised learning with dual path networks for hyperspectral target detection," *IEEE Trans. Geosci. Remote Sens.*, vol. 62, Apr. 2024, Art. no. 5515612.

[18] Y. Bengio, P. Lamblin, D. Popovici, and H. Larochelle, "Greedy layer-wise training of deep networks," in *Proc. 20th Int. Conf. Neural Inf. Process. Syst.*, 2006, pp. 153–160.

[19] W.-S. Hu, H.-C. Li, L. Pan, W. Li, R. Tao, and Q. Du, "Spatial–spectral feature extraction via deep ConvLSTM neural networks for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 6, pp. 4237–4250, Jun. 2020.

[20] J. Yang, B. Du, C. Wu, and L. Zhang, "Automatically adjustable multi-scale feature extraction framework for hyperspectral image classification," in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, 2021, pp. 3649–3652.

[21] A. Dosovitskiy, "An image is worth 16×16 words: Transformers for image recognition at scale," in *Proc. Int. Conf. Learn. Represent.*, Virtual Event, Austria, May 3–7, 2021.

[22] Z. Xue, X. Tan, X. Yu, B. Liu, A. Yu, and P. Zhang, "Deep hierarchical vision transformer for hyperspectral and LiDAR data classification," *IEEE Trans. Image Process.*, vol. 31, pp. 3095–3110, Apr. 2022.

[23] Y. Peng, Y. Zhang, B. Tu, Q. Li, and W. Li, "Spatial–spectral transformer with cross-attention for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, Sep. 2022, Art. no. 5537415.

[24] Y. Qing, W. Liu, L. Feng, and W. Gao, "Improved transformer net for hyperspectral image classification," *Remote Sens.*, vol. 13, no. 11, 2021, Art. no. 2216.

[25] S. Amini, S. Homayouni, A. Safari, and A. A. Darvishsefat, "Object-based classification of hyperspectral data using random forest algorithm," *GeoSpatial Inf. Sci.*, vol. 21, no. 2, pp. 127–138, 2018.

[26] Y. Zhang, P. Duan, L. Liang, X. Kang, J. Li, and A. Plaza, "PFS3F: Probabilistic fusion of superpixel-wise and semantic-aware structural features for hyperspectral image classification," *IEEE Trans. Circuits Syst. Video Technol.*, to be published, doi: 10.1109/TCSVT.2025.3556548.

[27] J. Yang, B. Du, and C. Wu, "Hybrid vision transformer model for hyperspectral image classification," in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, 2022, pp. 1388–1391.

[28] L. Liang, Y. Zhang, S. Zhang, J. Li, A. Plaza, and X. Kang, "Fast hyperspectral image classification combining transformers and SimAM-based CNNs," *IEEE Trans. Geosci. Remote Sens.*, vol. 61, Aug. 2023, Art. no. 5522219.

[29] S. Liu et al., "More convnets in the 2020s: Scaling up kernels beyond 51×51 using sparsity," 2022, *arXiv:2207.03620*.

[30] Y. Chen, J. Liu, X. Zhang, X. Qi, and J. Jia, "Largekernel3d: Scaling up kernels in 3D sparse CNNs," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2023, pp. 13488–13498.

[31] X. Ding, X. Zhang, J. Han, and G. Ding, "Scaling up your kernels to 31×31: Revisiting large kernel design in CNNs," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 11953–11965.

[32] X. Ding et al., "UniRepLKNet: A universal perception large-kernel ConvNet for audio, video, point cloud, time-series and image recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2024, pp. 5513–5524.

[33] R. Azad et al., "Beyond self-attention: Deformable large kernel attention for medical image segmentation," in *Proc. IEEE/CVF Winter Conf. Appl. Comput. Vis.*, 2024, pp. 1276–1286.

[34] A. G. Howard et al., "Mobilenets: Efficient convolutional neural networks for mobile vision applications," 2017, *arXiv:1704.04861*.

[35] P. Wu, Z. Wang, B. Zheng, H. Li, F. E. Alsaadi, and N. Zeng, "AGGN: Attention-based glioma grading network with multi-scale feature extraction and multi-modal information fusion," *Comput. Biol. Med.*, vol. 152, 2023, Art. no. 106457.

[36] Y. Zhang, L. Liang, J. Mao, Y. Wang, and L. Jia, "From global to local: A dual-branch structural feature extraction method for hyperspectral image classification," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 18, pp. 1778–1791, 2025.

[37] J. Lin, F. Gao, X. Shi, J. Dong, and Q. Du, "SS-MAE: Spatial–spectral masked autoencoder for multisource remote sensing image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 61, Nov. 2023, Art. no. 5531614.

[38] P. Sun, L. Wu, K. Zhang, X. Chen, and M. Wang, "Neighborhood-enhanced supervised contrastive learning for collaborative filtering," *IEEE Trans. Knowl. Data Eng.*, vol. 36, no. 5, pp. 2069–2081, May 2024.

[39] S. Feng et al., "Transformer-based cross-domain few-shot learning for hyperspectral target detection," *IEEE Trans. Geosci. Remote Sens.*, vol. 63, 2025, Art. no. 5501716.

[40] G. Haixiang, L. Yijing, J. Shang, G. Minyun, H. Yuanyue, and G. Bing, "Learning from class-imbalanced data: Review of methods and applications," *Expert Syst. Appl.*, vol. 73, pp. 220–239, 2017.

[41] L. Abdi and S. Hashemi, "To combat multi-class imbalanced problems by means of over-sampling techniques," *IEEE Trans. Knowl. Data Eng.*, vol. 28, no. 1, pp. 238–251, Jan. 2015.

[42] K.-B. Lin, W. Weng, R. K. Lai, and P. Lu, "Imbalance data classification algorithm based on SVM and clustering function," in *Proc. 9th Int. Conf. Comput. Sci. Educ.*, 2014, pp. 544–548.

[43] J. Li, Q. Du, Y. Li, and W. Li, "Hyperspectral image classification with imbalanced data based on orthogonal complement subspace projection," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 7, pp. 3838–3851, Jul. 2018.

[44] T. Sun, L. Jiao, J. Feng, F. Liu, and X. Zhang, "Imbalanced hyperspectral image classification based on maximum margin," *IEEE Geosci. Remote Sens. Lett.*, vol. 12, no. 3, pp. 522–526, Mar. 2014.

[45] M. Galar, A. Fernandez, E. Barrenechea, and F. Herrera, "EUSBoost: Enhancing ensembles for highly imbalanced data-sets by evolutionary undersampling," *Pattern Recognit.*, vol. 46, no. 12, pp. 3460–3471, 2013.

[46] J. Peng, L. Li, and Y. Y. Tang, "Maximum likelihood estimation-based joint sparse representation for the classification of hyperspectral remote sensing images," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 30, no. 6, pp. 1790–1802, Jun. 2018.

[47] J. Peng, W. Sun, and Q. Du, "Self-paced joint sparse representation for the classification of hyperspectral images," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 2, pp. 1183–1194, Feb. 2018.

[48] Y. Tian, D. Su, S. Lauria, and X. Liu, "Recent advances on loss functions in deep learning for computer vision," *Neurocomputing*, vol. 497, pp. 129–158, 2022.

[49] Y. Tian and Y. Zhang, "A comprehensive survey on regularization strategies in machine learning," *Inf. Fusion*, vol. 80, pp. 146–166, 2022.

[50] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 1800–1807.

[51] D. Ouyang et al., "Efficient multi-scale attention module with cross-spatial learning," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, 2023, pp. 1–5.

[52] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 7132–7141.

[53] S. Woo, J. Park, J.-Y. Lee, and I. S. Kweon, "CBAM: Convolutional block attention module," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 3–19.

[54] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, "Pyramid scene parsing network," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 6230–6239.
[55] Q. Hou, D. Zhou, and J. Feng, "Coordinate attention for efficient mobile network design," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 13708–13717.
[56] V. Nair and G. E. Hinton, "Rectified linear units improve restricted Boltzmann machines," in *Proc. 27th Int. Conf. Mach. Learn.*, 2010, pp. 807–814.
[57] V. Kothapalli, "Neural collapse: A review on modelling principles and generalization," 2022, *arXiv:2206.04041*.
[58] Y. Yang et al., "Neural collapse inspired feature-classifier alignment for few-shot class incremental learning," 2023, *arXiv:2302.03004*.
[59] Y. Yang, S. Chen, X. Li, L. Xie, Z. Lin, and D. Tao, "Inducing neural collapse in imbalanced learning: Do we really need a learnable classifier at the end of deep neural network?," in *Proc. 36th Int. Conf. Neural Inf. Process. Syst.*, 2022, pp. 37991–38002.
[60] C. Fu, B. Du, and L. Zhang, "Do we need learnable classifiers? A hyperspectral image classification algorithm based on attention-enhanced ResBlock-in-ResBlock and ETF classifier," *IEEE Trans. Geosci. Remote Sens.*, vol. 62, Jan. 2024, Art. no. 5505013.
[61] Y. Shi et al., "Towards understanding and mitigating dimensional collapse in heterogeneous federated learning," 2022, *arXiv:2210.00226*.
[62] Z. Zhang and M. R. Sabuncu, "Generalized cross entropy loss for training deep neural networks with noisy labels," in *Proc. 32nd Int. Conf. Neural Inf. Process. Syst.*, 2018, pp. 8792–8802.
[63] X. Li et al., "Generalized focal loss: Learning qualified and distributed bounding boxes for dense object detection," in *Proc. 34th Int. Conf. Neural Inf. Process. Syst.*, 2020, pp. 21002–21012.
[64] R. Müller, S. Kornblith, and G. Hinton, "When does label smoothing help?," in *Proc. 33rd Int. Conf. Neural Inf. Process. Syst.*, 2019, pp. 4694–4703.
[65] Hyperspectral Remote Sensing Scenes, 2017. [Online]. Available: https://www.ehu.eus/ccwintco/index.php/Hyperspectral_Remote_Sensing_Scenes
[66] Y. Zhong, X. Hu, C. Luo, X. Wang, J. Zhao, and L. Zhang, "WHU-Hi: UAV-borne hyperspectral with high spatial resolution (H2) benchmark datasets and classifier for precise crop identification based on deep convolutional neural network with CRF," *Remote Sens. Environ.*, vol. 250, 2020, Art. no. 112012.
[67] V. Sharma, A. Diba, T. Tuytelaars, and L. Gool, "Hyperspectral CNN for image classification and band selection, with application to face recognition," KU Leuven, ESAT, Leuven, Belgium, Tech. Rep. KUL/ESAT/PSI/1604, 2016.
[68] A. B. Hamida, A. Benoit, P. Lambert, and C. Ben Amar, "3-D deep learning approach for remote sensing image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 8, pp. 4420–4434, Aug. 2018.
[69] Z. Zhong, J. Li, Z. Luo, and M. Chapman, "Spectral–spatial residual network for hyperspectral image classification: A 3-D deep learning framework," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 2, pp. 847–858, Feb. 2017.
[70] S. K. Roy, G. Krishna, S. R. Dubey, and B. B. Chaudhuri, "HybridSN: Exploring 3-D–2-D CNN feature hierarchy for hyperspectral image classification," *IEEE Geosci. Remote Sens. Lett.*, vol. 17, no. 2, pp. 277–281, Feb. 2019.
[71] J. Yang, B. Du, and L. Zhang, "From center to surrounding: An interactive learning framework for hyperspectral image classification," *ISPRS J. Photogramm. Remote Sens.*, vol. 197, pp. 145–166, 2023.
[72] J. Yang, B. Du, Y. Xu, and L. Zhang, "Can spectral information work while extracting spatial distribution?—An online spectral information compensation network for HSI classification," *IEEE Trans. Image Process.*, vol. 32, pp. 2360–2373, Jan. 2023.
[73] J. Yang, B. Du, and L. Zhang, "Overcoming the barrier of incompleteness: A hyperspectral image classification full model," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 35, no. 10, pp. 14467–14481, Oct. 2024.
[74] H. Shi, Y. Zhang, G. Cao, and D. Yang, "Fortifying centers and edges: Multidomain feature learning meets hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 62, Apr. 2024, Art. no. 5513516.
[75] C. Li, B. Rasti, X. Tang, P. Duan, J. Li, and Y. Peng, "Channel-layer-oriented lightweight spectral-spatial network for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 62, Jan. 2024, Art. no. 5504214.
[76] Y. Cui, L. Zhu, C. Zhao, L. Wang, and S. Gao, "Lightweight spectral-spatial feature extraction network based on domain generalization for cross-scene hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 62, Jul. 2024, Art. no. 5525514.
[77] B. Qin, S. Feng, C. Zhao, B. Xi, W. Li, and R. Tao, "FDGNet: Frequency disentanglement and data geometry for domain generalization in cross-scene hyperspectral image classification," *IEEE Trans. Neural Netw. Learn. Syst.*, to be published, doi: 10.1109/TNNLS.2024.3445136.

Rong Liu (Member, IEEE) received the B.S. degree in surveying and mapping engineering from Wuhan University, Wuhan, China, in 2013, and the Ph.D. degree in photogrammetry and remote sensing from the State Key Laboratory of Information Engineering in Surveying, Mapping, and Remote Sensing, Wuhan University, Wuhan, in 2018.

She is currently an Associate Professor with the School of Geography and Planning, Sun Yat-sen University, Guangzhou, China. She was a Postdoctoral Researcher or a Senior Researcher with Remote Sensing Technology Institute, German Aerospace Center (DLR), Munich, Germany, and also with Signal Processing in Earth Observation, Technical University of Munich, Munich, from 2018 to 2021. Her research interests include remote sensing image processing, machine learning, and evolutionary computation.

Zhilin Li is currently working toward the B.S. degree in geographic information science with the School of Geography and Planning, Sun Yat-sen University, Guangzhou, China.

Her research interests include hyperspectral image classification, deep learning, and remote sensing image analysis.

Jiaqi Yang received the B.S. degree in geographic information science from Dalian Maritime University, Dalian, China, in 2019, and the M.E. degree in geomatics engineering from the State Key Laboratory of Information Engineering in Surveying, Mapping and Remote Sensing, Wuhan University, Wuhan, China, in 2021, and the Ph.D. degree in photogrammetry and remote sensing from Wuhan University, Wuhan, China, in 2024.

Her research interests include deep learning, satellite image processing, and hyperspectral image classification.

Jian Sun received the B.E. degree in surveying and mapping engineering from the School of Geodesy and Geomatics, Wuhan University, Wuhan, China, in 2013.

Since then, he has been long engaged in the application and promotion of surveying, mapping, and geographic information technology in urban planning and management, as well as the research on remote sensing image processing and intelligent surveying and mapping.

Quanwei Liu (Student Member, IEEE) received the B.S. degree in exploration technology and engineering from the Henan University of Engineering, Zhengzhou, China, in 2020, and the M.S. degree in resources and environment from the China University of Geosciences, Wuhan, China, in 2023. He is currently working toward the Ph.D. degree in multimodal image fusion and hyperspectral image classification with the College of Science and Engineering, James Cook University, Cairns, QLD, Australia.

His research interests include hyperspectral image processing, multimodal remote sensing fusion, and deep learning.