Citation:
C. Yu, J. Huang, M. Song, Y. Wang and C. -I. Chang, "Edge-Inferring Graph Neural Network with Dynamic Task-Guided Self-Diagnosis for Few-shot Hyperspectral Image Classification," in IEEE Transactions on Geoscience and Remote Sensing, 2022, doi: 10.1109/TGRS.2022.3196311.

DOI:
https://doi.org/10.1109/TGRS.2022.3196311

Edge-Inferring Graph Neural Network With Dynamic Task-Guided Self-Diagnosis for Few-Shot Hyperspectral Image Classification

Chunyan Yu, *Member, IEEE*, Jiahui Huang, Meiping Song, *Member, IEEE*, Yulei Wang, *Member, IEEE*, and Chein-I Chang, *Life Fellow, IEEE*

***Abstract*— The current hyperspectral image classification (HSIC) model based on the convolutional neural network for feature extraction and softmax classifier has been prone to the barrier of label prediction with limited samples. Substituting for the enormously complicated work of terrain labeling, few-shot learning provides a popular option for HSIC with very few annotated samples. In this article, we proposed a novel edge-inferring framework with the metalearning paradigm for hyperspectral few-shot classification (HSFSC), in which a graph neural network for similarity measurement is first presented to iteratively infer edge labels with the exploitation of instance-level similarity and the distribution-level similarity. Besides, in the metatraining stage, the pixel prediction model and the patch prediction model based on edge-inferring architecture are concretized jointly to improve the classification accuracy of the test samples. Expressly, at the metatesting phase, the dynamic task-guided self-diagnosis strategy is developed for the first time to diagnose the samples separability of the current classification task, which is responsible for dynamically assigning the most reliable results based on the generated reliability grade of the sample. The extensive experimental results and analysis of three hyperspectral image datasets demonstrate the superiority of the proposed HSFSC architecture compared with other advanced methods.**

***Index Terms*— Few-shot classification, graph neural network (GNN), hyperspectral image (HSI), label prediction, task-guided diagnosis.**

NOMENCLATURE

HSI	Hyperspectral image.
HSIC	HSI classification.
HSFSC	Hyperspectral few-shot classification.
DL	Deep learning.
CNN	Convolutional neural network.
GNN	Graph neural network.
FE	Feature extraction.
EIN	Edge-inferring network.
Spatial EIN	Spatial edge-inferring network.
Spectral EIN	Spectral edge-inferring network.
DTD	Dynamic Task-guided self-Diagnosis.
SPAENN	Spatial-feature Extraction Neural Network.
SPEENN	Spectral-feature Extraction Neural Network.
FMN	Feature mapping network.
SPA-network	SPAENN with spatial EIN.
SPE-network	SPEENN with spectral EIN.
IDD	Interclass discrepancy diagnosis.
PCD	Prediction consistency diagnosis.
EIGN	Spatial EIN with SPAEN.
DTDEIN	EIN with DTD.
x_i^{spa}	Patch data of the ith sample.
y_i	Ground-truth label of the ith sample.
x_i^{spe}	Pixel data of the ith sample.
V_i^{spa}	ith node in the graph of G^{spa}.
V_i^{spe}	ith node in the graph of G^{spe}.
$v_{l,i}^{\text{spa}}$	Feature of V_i^{spa} at the lth layer of spatial EIN.
$s_{l,ij}^{\text{spa}}$	Similarity between V_i^{spa} and V_j^{spa}.
$v_{l,i}^{\text{spe}}$	Feature of V_i^{spe} at the lth layer of spectral EIN.
$s_{l,ij}^{\text{spe}}$	Similarity between V_i^{spe} and V_j^{spe}.
$o_{l,ij}^{\text{spa}}$	Instance-level similarity between V_i^{spa} and V_j^{spa}.
$d_{l,ij}^{\text{spa}}$	Distribution similarity between V_i^{spa} and V_j^{spa}.
$o_{l,ij}^{\text{spe}}$	Instance-level similarity between V_i^{spe} and V_j^{spe}.
$d_{l,ij}^{\text{spe}}$	Distribution similarity between V_i^{spe} and V_j^{spe}.

Manuscript received 10 April 2022; revised 22 July 2022; accepted 29 July 2022. Date of publication 4 August 2022; date of current version 19 August 2022. The work of Chunyan Yu was supported by the Science Foundation of Liaoning Province (Surface Project) under Grant LJKZ0065. The work of Meiping Song was supported in part by the National Nature Science Foundation of China under Grant 61971082 and in part by the Fundamental Research Funds for the Central Universities under Grant 3132017124. *(Corresponding author: Meiping Song.)*

Chunyan Yu, Jiahui Huang, Meiping Song, and Yulei Wang are with the Center for Hyperspectral Imaging in Remote Sensing (CHIRS), Information and Technology College, Dalian Maritime University, Dalian 116026, China (e-mail: yucy@dlmu.edu.cn; hjh13009408@163.com; smping@163.com; wangyulei@dlmu.edu.cn).

Chein-I Chang is with the Center for Hyperspectral Imaging in Remote Sensing (CHIRS), Information and Technology College, Dalian Maritime University, Dalian 116026, China, and also with the Remote Sensing Signal and Image Processing Laboratory, Department of Computer Science and Electrical Engineering, University of Maryland at Baltimore County, Baltimore, MD 21250 USA (e-mail: cchang@umbc.edu).

Digital Object Identifier 10.1109/TGRS.2022.3196311

I. INTRODUCTION

BENEFITING from the development of optics and photonics, HSI consists of hundreds of contiguous spectral bands, which is beneficial for object detection with a wealth of abundant information [1]. Up until now, HSIC has played

an important role in the fields of urban planning, vegetation monitoring, and disaster prevention. In the past several decades, various machine learning approaches have been dedicated to implementing the HSIC, including support vector machines [2], random forest [3], and neural networks [4]. Nevertheless, the early staged methods adopted the handcrafted spectral–spatial features that are heavily governed by professional expertise and previous experience. To address this defect, DL [5], [6], [7], [8], [9] emerged as a powerful tool for strong representation ability and achieved great progress for HSIC. Compared with the traditional shallow classification model, the DL models are regarded as a classification model with a multilayer structure [10] to extract rich spatial information automatically. Among them, CNN that is composed of a series of convolution and pooling layers [11], [12] has made extraordinary achievements in the field of HSIC. For instance, Hu *et al.* [13] employed a five-layer 1-D CNN to conduct classification directly in the spectral domain. Chen *et al.* [14] adopted an unsupervised method to construct a DL framework based on a stacked autoencoder to extract high-order features of HSI data. Zhang *et al.* [15] encoded semantic context-aware representation with a diverse region-based CNN to obtain promising features. Notably, all the approaches mentioned above require massive labeled samples to train large-scale networks, which is the most fatal flaw of the DL framework.

Compared with the manual labeling of natural images, the label annotation of HSI is more expensive and difficult in a realistic scenario. Instead of using a fixed number or percentage of labeled sample selection, i.e., 200, 5%, and 10% in DL methods, many approaches [16], [17], [18], [19], [20] have been further proposed to promote the performance of HSIC with limited labeled samples. A tensor-board linear and nonlinear classification models in [21] and [22] are presented for HSI data in the situation of a small number of supervised samples. Especially, the approach of Makantasis *et al.* [22] significantly reduces the number of weight parameters resulting in required few training samples. In recent years, few-shot learning [23], [24] that is dedicated to few-shot image classification has acquired better accuracy compared with previous DL methods. Specifically, few-shot learning is one of the applications of metalearning [25], [26] in supervised learning, which decomposes the dataset into different metatasks to simulate the few-shot task and learn the generalization ability of the model. Given a K-shot C-way classification task, the purpose of the few-shot model is to use K-labeled training data of each class to learn the metaknowledge of C-category classification [27]. Based on the few-shot learning strategy mentioned above, Liu *et al.* [28] learned a generalized metric space by training the CNN model with a large number of labeled data and adapted it to new classes with limited supervised samples at the test phase. A spatial–spectral prototypical network [29] achieved an extraordinary classification effect with only a few labeled samples by learning the spatial–spectral metric space. Furthermore, a novel feature space learner [30] with a strategy of global representation learning is designed to achieve few-shot image classification.

Although the existing CNN-based methods have achieved significant classification performance, the traditional CNN model only conducts convolution on the regular square region and fails to adaptively capture the geometric changes between different object regions in HSI. In contrast, the GNN [31], [32], [37], [38], [39] is specially built to operate on non-Euclidean irregular data based on the predefined graph structure. In [33], GNN was first presented to build neural networks on graphs. After that, GNN-based models [34], [35], [36] have been increasingly popular in various areas and showed great potential ability. In HSIC, many methods applied GNN to achieve promising performance. Wan *et al.* [40] employed a multiscale dynamic GNN to incorporate spatial information at different scales. Hong *et al.* [41] presented a minibatch GNN that allows training a large-sized network in a minibatch fashion. In [42], autoregressive moving average filters that can better capture the global graph structure are applied to a semisupervised GNN for few-shot HSIC. Tong *et al.* [43] defined the feature aggregation function by introducing an attention-weighted graph into GNN for few-shot HSIC. Satorras *et al.* [44] first constructed a dense graph with all samples of the few-shot task where each node concatenated with features and class label, and the label information was propagated by iterative updating node features via the graph attention mechanism network.

The previous GNN approaches for HSIC in few-shot learning mainly rely on the node-labeling framework, which transfers the relation between every pair of nodes in an implicitly way. On the contrary, the edge-labeling framework explicitly performs the clustering by adjusting not only node features information but also edge-label information, which indicates whether the connected two nodes belong to the same class. The first attempt of the edge-labeling framework has been previously found in [45], which inferred the edge labels of the graph for correlation clustering. Typically, an edge-labeling GNN [46] was proposed for few-shot classification tasks under the episodic training framework. However, the edge-labeling framework has never been applied to the field of the few-shot HSIC. In addition, the general few-shot model highlights the distinctive features in the self-designed network and implements the classification directly. Typically, the approaches only strive to refine distinctive characteristics accurately at the network level, ignoring that from the perspective of sample composition with small interclass dissimilarity and large intraclass similarity that may lead to the wrong classification.

In this article, the edge-inferring graph model is presented to perform clustering by inferring edge labels for few-shot HSIC, which enhances the ability to distinguish different categories of nodes with the explicitly exploiting of edge features to maximize interclass dissimilarity. Besides, in the proposed model, both the instance-level similarity and the distribution-level similarity of the connected nodes contribute to the inference of the edge. Moreover, due to the abundant spatial and spectral information of HSIs, the edge-inferring module is adopted to extract features and perform edge inferring for spatial and spectral characteristics, respectively. In addition, considering the sample separability of different tasks in the episode, DTD

is proposed to analyze the separability environment of each query to guarantee the most reliable label decision.

In summary, the main contributions of this article are listed as follows.

1) The edge-inferring-based GNN framework is the first attempt to perform the few-shot classification for HSI, which employs GNN for metric learning by iteratively inferring edge labels to maximize interclass dissimilarity. Notably, the proposed edge-inferring module infers relation between samples with the exploitation of both instance- and distribution-level similarity simultaneously.
2) The strategy of realizing an edge-inferring module for the patch prediction model and pixel prediction model, respectively, is presented to improve the classification accuracy of test samples jointly. In this style, the presented architecture is contributed by employing hierarchical spatial features for patch prediction and spectral information for pixel prediction, respectively.
3) A DTD strategy is designed to assign class annotation accurately, which is implemented by the task-specified episode with discriminate separability dynamically. To the best of our knowledge, it is the first attempt to achieve class prediction in terms of the interclass discrepancy and PCD during the metatesting phase.

II. Proposed Approach

The proposed DTDEIN method mainly consists of three distinctive components, including the FE module, the EIN module, and the DTD block. Specifically, the FE module includes SPAENN and SPEENN that are responsible for extracting the spatial and spectral features, respectively. Particularly, EIN performs edge inferring by node features and edge-label propagation across the graph layers that are embodied in spatial EIN and spectral EIN according to respective extracted features. Fig. 1 exhibits the structure of the spatial EIN, and the structure of the spectral EIN is the same as the spatial EIN with different parameters. The procedure of similarity calculation in EIN is shown in Fig. 2. Besides, the flowchart of the proposed DTDEIN framework for HSFSC in the test phase is illustrated in Fig. 3. Especially, the DTD module is designed to preserve a comprehensive consideration of the patch prediction and the pixel prediction, which diagnoses the sample separability in the current classification task and keeps the global label with a more accurate prediction iteratively. More details of each module are described in the following.

A. Problem Definition

Due to the spatial and spectral characteristics, HSI data are composed of the patch set denoted as D^{spa} and pixel set recorded as D^{spe}. In the setting, the samples of several classes from D^{spa} and D^{spe} are selected to form patch-prediction tasks and pixel-prediction tasks, respectively. Each task includes a support set and a query set. Specifically, for each patch task, C classes are randomly selected, and K examples for each class from the patch set are sampled to form the patch support set and Q examples per class to form the patch query set. Similarly, the pixel task of the same size is obtained. In addition, in the training stage, a query set with data labels is in charge to evaluate the learned classifier, and the training strategy follows the way of episodic training. In the test phase, the samples in the patch task and the pixel task are corresponding.

B. Feature Extractor

In the proposed DTDEIN architecture, two networks named SPAENN and SPEENN are employed for FE to collect information for the subsequent graph construction. Specifically, SPAENN is utilized to extract the spatial features, and SPEENN acts on channel dimensions of original pixel data to capture intrinsic spectral features.

In the implementation, the SPAENN with the parameter set ϑ^{spa} is accomplished by four 3×3 convolution layers consequently and a fully connected layer. For input patch data x_i^{spa}, the produced spatial feature representation with a 1-D vector is obtained by the SPAENN. SPEENN with the parameter set ϑ_i^{spe} is composed of only one full connection layer for implementation simplicity. With pixel data x_i^{spe} fed into the network, spectral feature representation with a 1-D vector is acquired by the SPEENN. The specific parameter settings are described in the experimental analysis section.

C. Edge-Inferring Graph Neural Network

To enhance the interclass discrepancy to better distinguish different categories, we develop an edge-inferring module with the explicit exploitation of edge features to infer a query related to existing support clusters. Edge-inferring architecture is constructed by stacking lth convolutional layers, in which each layer performs edge inferring, including a node update block and an edge update block. Based on edge-inferring architecture, the spatial EIN and the spectral EIN are designed for patch edge predictions and pixel edge predictions. The edge-inferring module on spatial characteristics and spectral characteristics are described in detail in the following parts.

To illustrate the principle of our proposed framework clearly, we list the primary denotations of the subsequent section in the right table.

1) Graph Construction: As demonstrated in Fig. 1, a fully connected graph is constructed according to the rule that each node represents each sample, and each edge represents the similarity between the two connected nodes. Specifically, $G^{\text{spa}} = (V^{\text{spa}}, E^{\text{spa}})$ is constructed with the patch task in the proposed framework, where $V^{\text{SPA}} := \{V_i^{\text{spa}}\}_{i=1,\ldots,N}$ and $E^{\text{SPA}} := \{E_{ij}^{\text{spa}}\}_{i,j=1,\ldots,N}$ denote the set of nodes and edges of G^{spa} and $G^{\text{spe}} = (V^{\text{spe}}, E^{\text{spe}})$ is built by the pixel task, where $V^{\text{SPE}} := \{V_i^{\text{spe}}\}_{i=1,\ldots,N}$ and $E^{\text{SPE}} := \{E_{ij}^{\text{spe}}\}_{i,j=1,\ldots,N}$ denote the set of nodes and edges of G^{spe}, and $N = C \times (K + Q)$ is the total number of nodes in G^{spa} and G^{spe}.

For the initialization, $v_{0,i}^{\text{spe}}$ and $v_{0,i}^{\text{spe}}$ are obtained by SPAENN and SPEENN, respectively. $s_{0,ij}^{\text{spa}}$ and $s_{0,ij}^{\text{spe}}$ are initialized in terms of the following formula:

$$s_{0,ij}^{\text{spa}} = s_{0,ij}^{\text{spe}} = \begin{cases} 1, & \text{if } z_{ij} = 1 \text{ and } i, j \le N \times K \\ 0, & \text{if } z_{ij} = 0 \text{ and } i, j \le N \times K \\ 0.5, & \text{othersize} \end{cases}$$

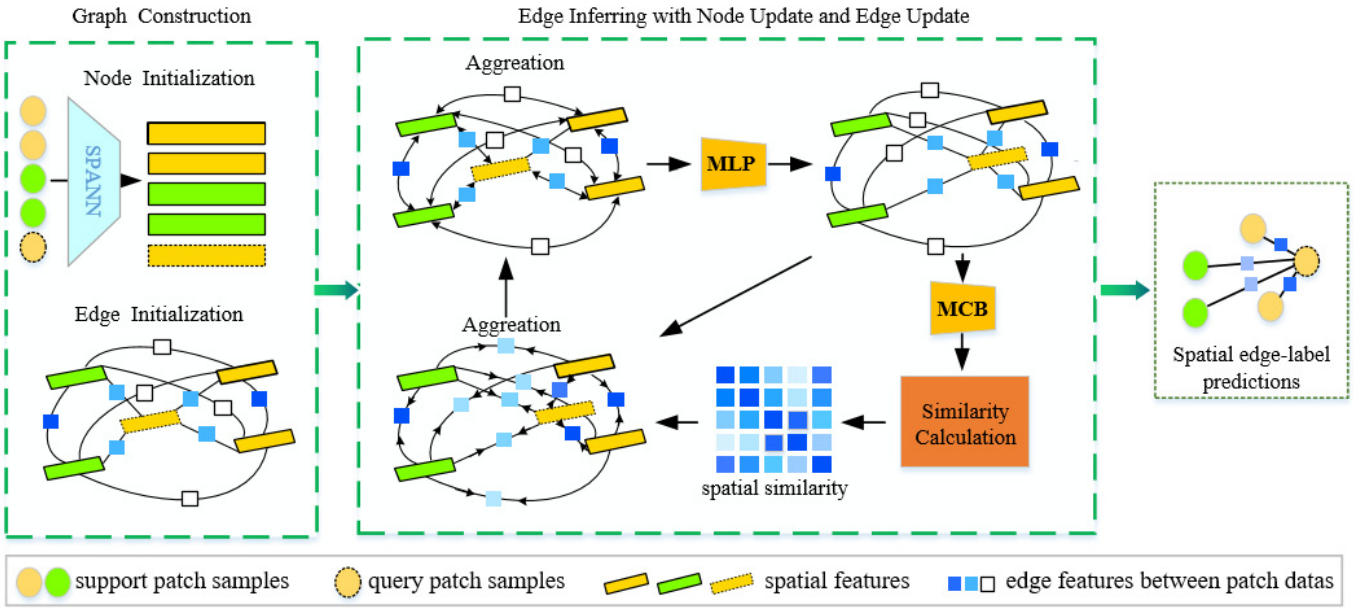


Fig. 1. Framework of the spatial edge-inferring network. In this illustration, a two-way two-shot problem is presented as an example. First, the graph is constructed with the feature vector obtained by SPAENN. Second, the node update and the edge update block are carried out with similarity calculation. Finally, the edge-label predictions are obtained.

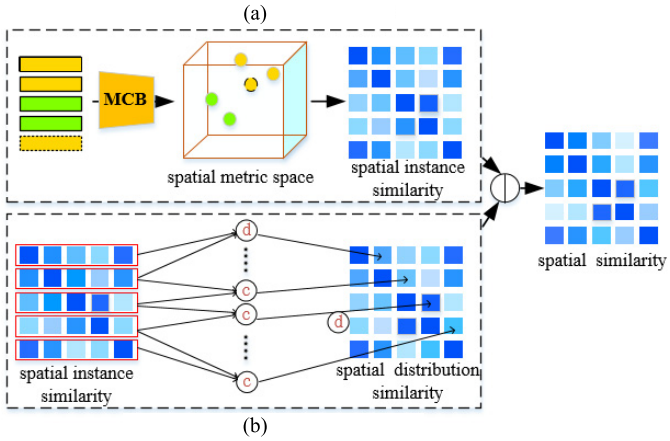


Fig. 2. Spatial similarity calculation with the consideration of instance similarity and distribution similarity. The symbol of represents the operation of (3) for obtaining distribution similarity, and represents the operation of (4) and (5) for obtaining final spatial similarity. (a) Instance-Similarity computation. (b) Distribution-Similarity computation.

where z_{ij} representing the edge label that is defined by

$$z_{ij} = \begin{cases} 1, & \text{if } y_i = y_j \\ 0, & \text{otherwise.} \end{cases}$$

2) Node Updation: According to the graph attention mechanism, edge features are regarded as attention coefficients to aggregate neighbor node information. First, given $v_{l-1,i}^{\text{spa}}$ and $s_{l-1,ij}^{\text{spa}}$ from the $l-1$th layer, the neighborhood feature aggregation is calculated by the following equation:

$$v_{l,i}^{\text{spa}} = f_{l-1} \circ u_{l-1} \circ v_{l-1,i}^{\text{spa}} \tag{1}$$

where the symbol of $\circ$ denotes the concatenation operation, and $f_{l-1} = \sum_j (v_{l-1,j}^{\text{spa}} \cdot s_{l-1,ij}^{\text{spa}})$ and $u_{l-1} = \sum_j (v_{l-1,j}^{\text{spa}} \cdot (1.0 - s_{l-1,ij}^{\text{spa}}))$ represent the aggregated information with neighbor nodes, emphasizing similar information and different information, respectively.

Subsequently, $v_{l,i}^{\text{spa}}$ is further transformed by the following equation:

$$v_{l,i}^{\text{spa}} \leftarrow M_l\left(v_{l,i}^{\text{spa}}; \delta_l^{\text{spa}}\right) \tag{2}$$

where M_l is an MLP with parameters of δ_l^{spa} that is stacked with two convolution blocks.

Similarly, given $v_{l-1,i}^{\text{spe}}$ and $s_{l-1,ij}^{\text{spe}}$ from the layer $l-1$th layer, $v_{i,l}^{\text{spe}}$ is achieved by M_l with the parameters of δ_l^{spe}.

3) Edge Updation: In the proposed network, edge features are updated based on the newly updated node features with the calculation of spatial similarity, which consists of instance-level similarity and distribution-level similarity, as shown in Fig. 2. First, given $v_{l,i}^{\text{spa}}$, a multiple convolution blocks (MCB) with the parameters of ζ_l^{spa} is employed to transfer the nodes to a new embedding-feature space. Next, the Euclidean distance between the nodes is adopted to measure $o_{l,ij}^{\text{spa}}$ in our architecture. Afterward, $d_{l,ij}^{\text{spa}}$ is acquired for each pair node by considering the instance similarity over all nodes in the graph in a one-versus-N manner by the following equation:

$$d_{l,ij}^{\text{spa}} = 1.0 - \sigma\left(\sum_m^N \left|o_{l,im}^{\text{spa}} - o_{l,jm}^{\text{spa}}\right|\right) \tag{3}$$

where σ is the sigmoid function aiming to transform the instance similarity to a certain scope.

Finally, the feature of each edge is updated with the following formulas:

$$s_{l,ij}^{\text{spa}} = o_{l,ij}^{\text{spa}} \cdot d_{l,ij}^{\text{spa}} \cdot s_{l-1,ij}^{\text{spa}} \tag{4}$$

$$s_{l,ij}^{\text{spa}} = g\left(s_{l,i1}^{\text{spa}}, s_{l,i2}^{\text{spa}}, \ldots, s_{l,ij}^{\text{spa}}, \ldots, s_{l,iN}^{\text{spa}}\right) \tag{5}$$

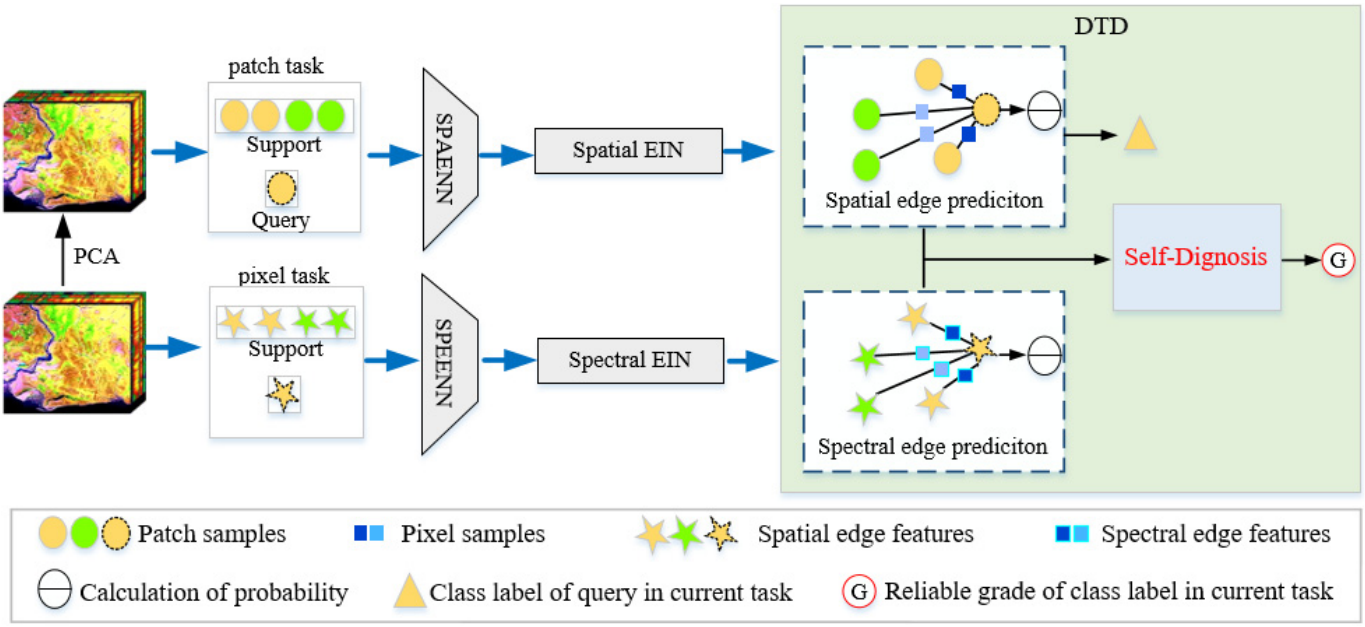


Fig. 3. Procedural diagram of the DTDEIN model in the test phase. First, the samples in the patch task and the pixel task are performed edge inferring with extracted features, respectively, to obtain patch prediction and pixel prediction. Next, DTD realizes prediction improvement through the reliability grade of the label obtained by self-diagnosis.

where $\cdot$ represents the multiplication and g is the L_1 regularization operation.

Likewise, given $v_{l,i}^{\text{spe}}$ and $v_{l,j}^{\text{spe}}$ from the lth layer, the spectral similarity $s_{l,ij}^{\text{spe}}$ is finally obtained by an MLP block with parameters of ζ_l^{spe}.

4) Loss Function: In the metatraining phase, the parameters of $\vartheta^{\text{spa}} \cup \{\delta_l^{\text{spa}}, \zeta_l^{\text{spa}}\}_{l=1}^{L}$ in the SPA-network are trained with the following loss function:

$$L^{\text{spa}} = \sum_{l=1}^{L} \sum_{m=1}^{M} \lambda_l L_e\left(Y_{m,e}, \hat{Y}_{m,e}^{l,\text{spa}}\right) \tag{6}$$

where $Y_{m,e}$ and $\hat{Y}_{m,e}^{l,\text{spa}}$ are ground-truth edge-labels' matrix and edge predictions matrix of the mth task at the lth layer in the SPA-network, respectively, L_e is binary cross-entropy loss in an end-to-end fashion, λ_l represents the coefficient, L means the layer number of spatial-EIN, and M is the number of tasks for one iteration.

In addition, for the SPE-network with the parameters of $\vartheta^{\text{spe}} \cup \{\delta_l^{\text{spe}}, \zeta_l^{\text{spe}}\}_{l=1}^{L}$, the loss function for the episode training is defined by the following formula:

$$L^{\text{spe}} = \sum_{l=1}^{L} \sum_{m=1}^{M} \lambda_l L_e\left(Y_{m,e}, \hat{Y}_{m,e}^{l,\text{spe}}\right) \tag{7}$$

where $\hat{Y}_{m,e}^{l,\text{spe}}$ are edge predictions matrix of the mth task at the lth layer in the SPE-network.

In summary, the process of the training episode is outlined in Algorithm 1.

D. Dynamic Task-Guided Self-Diagnosis

The key to the HSIC framework is to enhance separability by exploiting spectral and spatial features. In this work, at the metatesting phase, due to the randomness and heterogeneity of classification tasks under the metalearning strategy, we propose DTD consisting of IDD and PCD, which carries on self-diagnoses of sample separability with probability distributions of patch task and pixel task, and keeps the most reliable classification results with label decision.

We set label prediction of a query q as c_q and corresponding reliability grade recorded as r_q in the current classification task. In addition, since different tasks may have different prediction labels on the same query q to assign the most reliable classification results as the task changes, we set a global variable y_q to record the most reliable classification results and g_q to record the corresponding reliability grade. The probability of a sample q that belongs to the existing support class c_k is calculated by the following equation:

$$p_{q,k}^{x} = \frac{\sum_{j:j\neq q \wedge y_j = c_k} s_{qj,L}^{x}}{K} \tag{8}$$

where $p_{q,k}^{x}$ is the probability of sample q that belongs to c_k in the x-network, $x \in$ ("spa", "spe"), and $s_{qj,L}^{x}$ represents the edge feature of the final layer.

1) IDD: In this article, IDD refers to the diagnosis of whether the strength of interclass dissimilarity is enough large to distinguish heterogeneous samples. Specifically, the largest and second probabilities on the query sample q recorded as f_q^x and s_q^x can be obtained. IDD acting on sample q is denoted as $\text{IDD}(h_x(q))$ to verify whether the strength of interclass discrepancy contributes to classification depending on the setting critical value θ, which is implemented as follows:

$$\text{IDD}(h_x(q)) = \begin{cases} 1, & h_x(q) >= \theta \\ 0, & \text{otherwise} \end{cases} \tag{9}$$

Algorithm 1 Training Episode Algorithm

Input: The patch tasks and pixel tasks
Output: $\vartheta^{spa} \cup \{\delta_l^{spa}, \zeta_l^{spa}\}_{l=1}^{L} \vartheta^{spe} \cup \{\delta_l^{spe}, \zeta_l^{spe}\}_{l=1}^{L}$
/*The Training Episode in SPA-network
for $t = 1, \ldots, M$
  Randomly generate a patch task
  $v_{0,i}^{spa} \leftarrow$ Spatial feature extraction by SPAENN with x_i^{spa} and ϑ^{spa}
  $G^{spa} = (V^{spa}, E^{spa}) \leftarrow$ Graph construction
  for $l = 1, \ldots, N$ **do**
    for $i = 1, \ldots, N$ **do**
      $v_{l,i}^{spa} \leftarrow$ Node update with $\{v_{l-1,i}^{spa}\}, \{s_{l-1,ij}^{spa}\}; \delta_l^{spa}$
    end
    for $i, j = 1, \ldots, N$ **do**
      $s_{l,ij}^{spa} \leftarrow$ Edge update with $\{v_{l,i}^{spa}\}, \{s_{l-1,ij}^{spa}\}; \zeta_l^{spa}$)
    end
    $\hat{Y}_{t,e}^{l,spa} \leftarrow \{s_{l,ij}^{spa}\}$
  end
end
$L^{spa} \leftarrow$ Calculate loss with Equ. (6)
Update $\vartheta^{spa} \cup \{\delta_l^{spa}, \zeta_l^{spa}\}_{l=1}^{L} \leftarrow \nabla(L^{spa})$.
/* The Training Episode in SPE-network.
for $t = 1, \ldots, M$
  Randomly generate a pixel task
  $v_{0,i}^{spe} \leftarrow$ Spatial feature extraction by SPEENN with x_i^{spe}; ϑ^{spe})
  $G^{spe} = (V^{spe}, E^{spe}) \leftarrow$ Graph construction
  for $l = 1, \ldots, N$ **do**
    for $i = 1, \ldots, N$ **do**
      $v_{l,i}^{spe} \leftarrow$ Node update with $\{v_{l-1,i}^{spe}\}, \{s_{l-1,ij}^{spe}\}; \delta_l^{spe}$
    end
    for $(i, j) = 1, \ldots, N$ **do**
      $s_{l,ij}^{spe} \leftarrow$ Edge update with $\{v_{l,i}^{spe}\}, \{s_{l-1,ij}^{spe}\}; \zeta_l^{spe}$
    end
    $\hat{Y}_{t,e}^{l,spe} \leftarrow \{s_{l,ij}^{spe}\}$
  end
end
$L^{spe} \leftarrow$ Calculate loss with Equ. (7)
Update $\vartheta^{spe} \cup \{\delta_l^{spe}, \zeta_l^{spe}\}_{l=1}^{L} \leftarrow \nabla(L^{spe})$.

where the value of $h_x(q)$ represents the most influential interclass dissimilarity for sample q in the x-network, which is calculated by

$$h_x(q) = f_q^x - s_q^x. \tag{10}$$

2) Predictions Consistency Diagnosis: PCD refers to judging whether the class attributes predicted for spatial features and spectral features are consistent. PCD acting on sample q is denoted as PCD($f(q)$) to verify whether classification results are consistent with the SPA-network and the SPE-network, which is implemented as follows:

$$\mathrm{PCD}(f(q)) = \begin{cases} 1, & f(q) = 0 \\ 0, & \text{otherwise} \end{cases} \tag{11}$$

where $f(q) = c_q^{\mathrm{spa}} - c_q^{\mathrm{spe}}$. c_q^{spa} and c_q^{spe} represent classification results for spatial and spectral, respectively, which

Algorithm 2 DTD

Input: $s_{ij,L}^{spa}$, $s_{ij,L}^{spe}$, y_i g_i
Output: y_i, g_i // global classification label and reliability grade after DTD
// In current classification task
for $q = C \times K + 1, \ldots, C \times K + Q$ **do**
  $p_{q,k}^{spa}, p_{q,k}^{spe} \leftarrow$ Obtain two sets of probability distribution with Equ. (8)
  IDD($h_{spa}(q)$), IDD($h_{spe}(q)$) $\leftarrow$ Calculation with Equ. (9)
  PCD($f(q)$) $\leftarrow$ Calculation with Equ. (11)
  /* Decision label
  $c_q \leftarrow$ Obtain the classification result in the current task
  $r_q \leftarrow$ Obtain the reliability grade of c_q by Table I
  if g_q! = *"I"*
    if r_q == "I"
      $y_q = c_q$, $g_q =$ "I"
    else if g_q! = "II" and r_q == "II"
      $y_q = c_q$, $g_q =$ "II"
    else if g_q == null
      $y_q = c_q$, $g_q =$ "III"
    else Keep the previous prediction.
  end
end

TABLE I
PREDICTION RELIABILITY SETTING OF THE DIFFERENT LEVEL BASIS

Reliability level	I	II	III
IDD (h_{spa}(q))	1	-	-
IDD (h_{spe}(q))	1	-	-
PCD(f(q))	1	1	0

is completed by

$$c_q^{\mathrm{spa}} = \arg\max_k \left(p_{q,k}^x\right), \quad c_q^{\mathrm{spa}} = \arg\max_k \left(p_{q,k}^x\right). \tag{12}$$

3) Label Decision: We adopt the result of the SPA-network c_q^{spa} as label prediction in the current classification task ($c_q = c_q^{\mathrm{spa}}$). The outputs of IDD and PCD correspond to the separability level, which indicates the reliability grade for label prediction in the current classification task. Table I shows corresponding reliability grades with PCD and IDD, in which I, II, and III represent classification results reliability grade and present the order from high-reliability grade to low-reliability grade, and the number "1" represents that the corresponding diagnosis infers that the current classification environment satisfies the corresponding requirement. Therefore, the reliability grade for label prediction r_q is obtained in Table I. As shown in Algorithm 2, following the rule of updating the results of the low-reliability grade with the results of the high-reliability grade, we decide whether update the global label variable with the current label prediction. In this way, each sample is guaranteed to assign the most reliable classification label eventually.

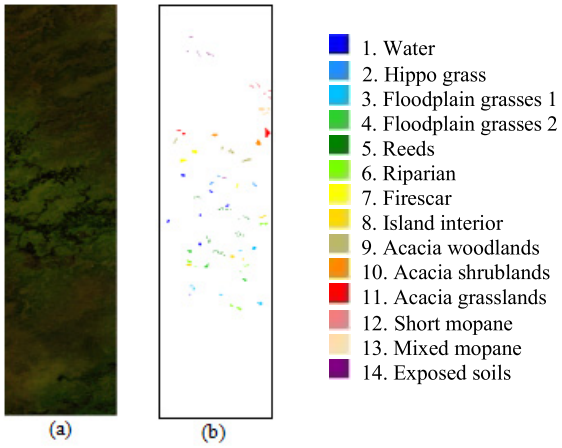


Fig. 4. Data of the Botswana scene. (a) False-color image. (b) Ground-truth image.

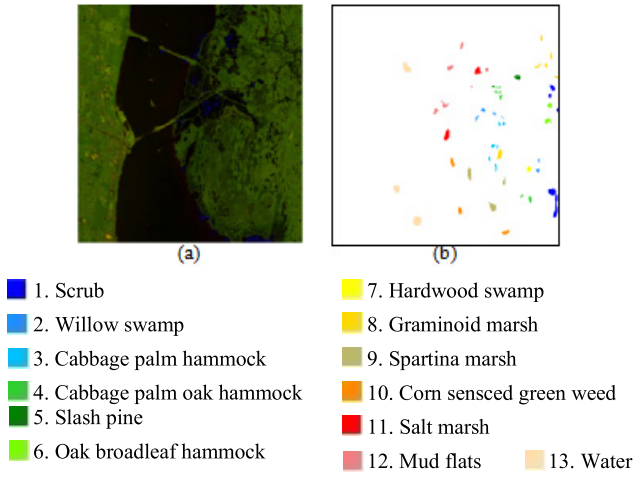


Fig. 5. Data of KSC. (a) False-color image. (b) Ground-truth image.

III. Experiments and Analysis

A. Data Description

In this section, three kinds of extensively popular datasets are adopted to validate the effectiveness of our proposed model. The first one is the Kennedy Space Center (KSC) data that are acquired by the AVIRIS sensor in the USA. The size of the data is 512 × 614, and it contains 176 bands after the water absorption bands removing. This dataset includes 13 land-cover classes, and the false-color image and the ground-truth image map are illustrated in Fig. 4(a) and (b), respectively.

The second dataset is called the Botswana data, which was captured over Botswana in Okavango Delta with the NASA EO-1 Hyperion sensor. The resolution of the data is 1476 × 256 with 145 bands after removing uncalibrated and noisy bands. The false-color image and the ground-truth map cover 13 land-cover classes are shown in Fig. 5(a) and (b).

The last dataset is the Houston data that were collected from the University of Houston campus and the neighboring urban area. This scene contains 144 bands in the 380–1050-nm region and 349 × 1905 pixels. Fig. 6(a) and (b) exhibits a false-color composite image and the ground-truth map consists of 15 labeled classes, respectively.

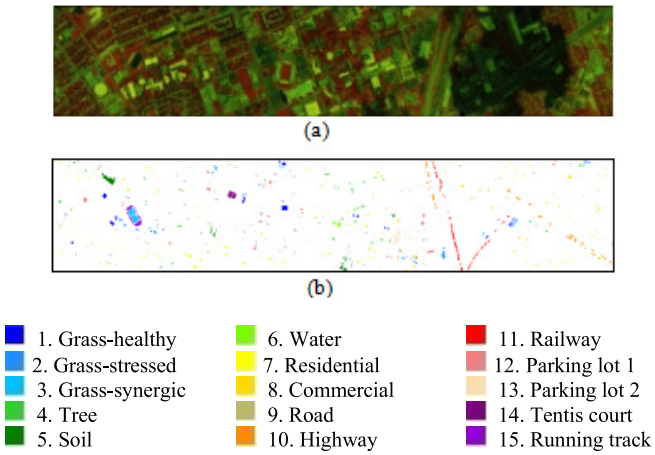


Fig. 6. Data of Houston. (a) False-color image. (b) Ground-truth image.

TABLE II
Hyperparameter Setting of the Proposed Model

Module	Layer setting	Input size	Kernel size	Output size
SPAENN	Cov1	11×11× c	3×3×64	9×9×64
	Cov2	9×9×64	3×3×96	7×7×96
	Cov3	7×7×96	3×3×128	3×3×128
	Cov4	3×3×128	3×3×256	1×1×256
	Linear	1×256	-	1×128
SPEENN	Linear	1×a	-	1×128
MLP	Cov1	1×1×3(b-l)	1×1×128	1×1×128
	Cov2	1×1×128	1×1×96	1×1×96
MCB	Cov1	1×1×96	1×1×128	1×1×128
	Cov2	1×1×128	1×1×128	1×1×128
	Cov3	1×1×128	1×1×96	1×1×96
	Cov4	1×1×96	1×1×96	1×1×96
	Cov5	1×1×96	1×1×3	1×1×3

B. Experimental Configuration

To explore the classification performance of the proposed DTDEIN network for HSFSC, a series of other state-of-the-art models were also exploited for comparison, including GNN [11], EGNN [46], PN [14], SSRN [28], and FADCNN [47]. In addition, the EGNN with the PCD block is denoted as TDEGNN, and the DTDEIN network without the DTD block is called the EIGN method. In our experiment, for EGNN and the extended TDEGNN and EIGN, we accomplished them with the same hyperparameters as our method. For others, we accomplished them with the default parameters of the original approaches.

The hyperparameter setting of the proposed method is listed in Table II. Here, for the Houston dataset, c represents the number of primary components after PCA, and for the KSC dataset, c is the initial band number of the dataset. a is the band number of the dataset, and $b-l$ is the input channel number at the lth layer of MLP. Under the FSCL setting of the proposed method, we cut the HSI into 11 × 11 image size to form a patch dataset D^{spa} through PCA and obtain a

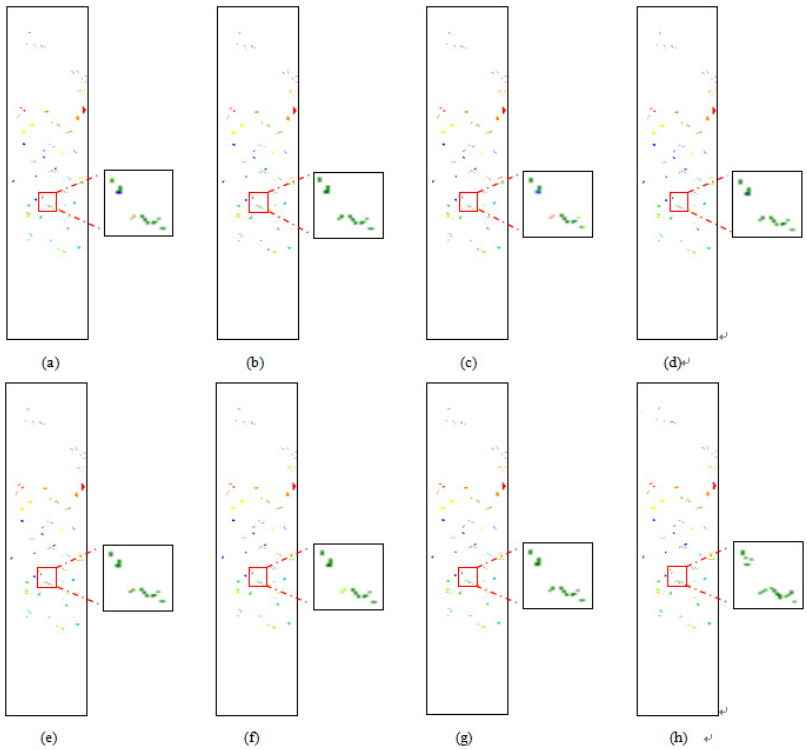


Fig. 7. Classification maps of the Botswana scene with compared methods. (a) PN. (b) SSRN. (c) FADCNN. (d) GNN. (e) EGNN. (f) EGIN. (g) TDEGNN. (h) DTDEIN.

pixel dataset D^{spe} with pixel data x_i^{spe} of HSI. Besides, for the employed Botswana dataset, we randomly select seven labeled samples (i.e., examples) in each class and nine labeled samples (i.e., examples) in each class for KSC and Houston.

In our experiments, the overall accuracy (OA) and the average accuracy (AA) are employed as objective criteria to evaluate the effectiveness of all the compared methods. In the training phase, we conducted a five-way five-shot experiment and sampled one query for each of the five classes randomly. In the testing phase, we conducted a five-way two-shot experiment and sampled five queries for each of the five classes randomly. The Adam optimizer is employed in our DTDEIN with the initial learning rate of 10^{-3}, which is conducted by each iteration with a weight decay of 10^{-6}. The other setting includes M used for metatraining of 20, L for the layer number of 3, and υ^x for PCD of 0.5. In addition, the coefficient parameters are set to $\lambda_1 = 0.5$, $\lambda_2 = 0.5$, and $\lambda_3 = 1.0$. All the methods are conducted five times via PyTorch 1.1 with Python 3.8 with an Intel I5-8400 CPU and 8-GB RAM.

C. Results and Analysis

In the experiment for Botswana, the classification results of OA obtained by different methods on the Botswana dataset are illustrated in Table III, and Fig. 7 demonstrates the visual comparison of the result maps. As shown in Table III, the proposed DTDEIN achieved the highest OA of 99.88% among all the compared methods. Moreover, we observe that the performance of EGNN (96.90%) based on the edge framework is better than GNN based on the node framework of OA (96.2%). TDEGNN increased OA to 98.40%, which proves the effectiveness of the DTD block. The EIGN model achieved the OA of 98.42%; it can be seen that the proposed edge derivation module has a more positive effect than EGNN on the dataset. In addition, from the classification map in Fig. 7, only the sixth class ("Riparian") occurs classification error, which further demonstrates the effectiveness of our proposed DTDEIN.

In the experiment for the KSC, Fig. 8 illustrated colorful result maps of the different models. As shown in Table IV, the proposed DTDEIN still yielded the best classification result (98.41%). TDEGNN outperformed EGNN, and DTDEIN outperformed TDEGNN by a substantial margin, which indicated that the DTD block can improve performance effectively. From the classification results in Fig. 8, all methods without DTD block have a lot of classification errors between the fourth class and the 12th class. Comparatively, TDEGNN improves this problem. Furthermore, the result of the proposed DTDEIN having a DTD block shows fewer error classifications and yields a smoother visual effect. Therefore, it is convincing to conclude that the proposed DTDEIN has stronger robustness than the compared methods.

The last experiment is performed at the Houston. The colorful classification maps are illustrated in Fig. 9. The experimental results of different methods are presented in Table V; similar to the results in Botswana and KSC, EGNN outperformed the GNN. TDEGNN increases accuracies by 12.25% of OA. The highest OA of 91.12% was still yielded by our approach compared with other methods, which validates the strength of our proposed method. In addition, it can be seen that the performance of all methods was worse than that of the KSC and Botswana datasets. The reason is that the Houston dataset has a lower spectral resolution and contains much noise than Salinas and Botswana. Furthermore, from the classification results in Fig. 9, stronger spatial correlation

TABLE III
CLASSIFICATION PERFORMANCE COMPARISON WITH ALL THE COMPARED METHODS OF THE BOTSWANA SCENE

Class	PN	SSRN	FADCNN	GNN	EGNN	EIGN	TDEGNN	DTDEIN
C1	99.26±1.05	**100.00±0.00**	**100.00±0.00**	99.63±0.52	99.75±0.43	**100.00±0.00**	**100.00±0.00**	**100.00±0.00**
C2	**100.00±0.00**	99.01±0.99	**100.00±0.00**	99.5±0.71	99.01±0.99	99.01±1.40	**100.00±0.00**	**100.00±0.00**
C3	97.61±0.86	99.73±0.46	94.52±0.33	97.61±2.81	99.60±0.69	99.20±0.57	**100.00±0.00**	99.93±0.16
C4	**100.00±0.00**	**100.00±0.00**	97.67±0.81	98.37±2.31	**100.00±0.00**	**100.00±0.00**	**100.00±0.00**	**100.00±0.00**
C5	81.04±1.39	95.41±4.44	71.00±1.05	83.28±8.29	83.02±3.03	90.34±3.16	85.13±1.05	98.58±0.64
C6	94.55±0.98	87.85±4.10	72.58±1.35	88.48±4.73	91.94±0.21	99.44±0.79	99.07±1.32	**100.00±0.00**
C7	98.84±0.55	**100.00±0.00**	93.24±0.64	98.46±2.18	99.48±0.59	99.42±0.82	99.42±0.82	**100.00±0.00**
C8	96.55±2.45	**100.00±0.00**	84.24±0.60	99.76±0.35	99.01±0.85	**100.00±0.00**	99.51±0.70	**100.00±0.00**
C9	85.77±3.04	99.04±0.32	66.48±1.18	95.54±5.85	**100.00±0.00**	**100.00±0.00**	99.84±0.23	**100.00±0.00**
C10	97.44±1.01	**100.00±0.00**	97.58±0.70	93.95±1.71	98.39±2.79	98.79±1.15	**100.00±0.00**	**100.00±0.00**
C11	99.12±1.02	99.89±0.18	97.79±0.63	93.44±4.17	95.30±1.24	99.18±1.16	**100.00±0.00**	**100.00±0.00**
C12	97.97±1.59	99.82±0.32	97.93±0.72	98.35±2.34	99.82±0.32	95.58±2.35	**100.00±0.00**	**100.00±0.00**
C13	99.75±0.35	99.88±0.21	95.80±0.55	97.58±1.32	96.89±2.43	97.76±3.17	**100.00±0.00**	**100.00±0.00**
C14	93.69±2.27	85.26±2.78	93.41±1.39	**100.00±0.00**	99.65±0.61	**100.00±0.00**	**100.00±0.00**	**100.00±0.00**
OA	95.45±0.34	98.01±0.14	88.94±0.21	95.38±0.08	96.90±0.46	98.42±0.11	98.60±0.11	**99.88±0.05**
AA	95.83±0.44	97.50±0.29	90.16±0.14	96.00±0.11	97.28±0.13	98.48±0.23	98.79±0.11	**99.89±0.05**

TABLE IV
CLASSIFICATION PERFORMANCE COMPARISON WITH ALL THE COMPARED METHODS OF THE KSC

Class	PN	SSRN	FADCNN	GNN	EGNN	EIGN	TDEGNN	DTDEIN
C1	94.92±1.56	**99.08±0.13**	91.46±0.72	95.27±1.91	94.02±2.56	96.06±0.53	94.09±0.99	98.56±0.19
C2	54.19±3.78	91.98±2.27	62.55±1.65	64.61±1.03	69.14±6.59	75.45±2.05	65.02±3.5	**93.56±0.19**
C3	88.80±1.33	91.22±0.59	83.59±2.35	79.30±2.15	85.74±2.15	85.68±2.71	89.06±0.20	**98.57±0.37**
C4	41.27±7.94	92.86±3.97	60.32±4.77	54.76±2.78	54.77±5.56	65.74±3.31	64.68±1.79	**97.36±0.94**
C5	77.64±7.19	99.38±0.62	75.78±0.63	78.26±1.24	81.99±6.21	77.23±4.54	81.99±0.94	**99.59±0.29**
C6	61.43±2.97	94.11±3.72	72.05±5.90	77.29±2.62	77.95±5.90	85.88±5.48	75.55±2.62	**98.40±0.74**
C7	88.89±0.90	98.58±0.48	98.10±5.24	96.19±0.96	99.05±0.95	98.41±2.24	99.05±0.02	**100.00±0.00**
C8	67.60±7.61	69.49±0.58	80.04±1.16	83.53±1.74	79.12±0.93	75.41±4.61	94.43±0.82	**95.98±1.42**
C9	80.19±2.68	**100.00±0.00**	95.19±7.31	94.04±0.58	94.81±0.39	98.53±0.40	99.23±0.05	**100.00±0.00**
C10	89.35±2.38	92.08±2.23	79.21±3.84	77.48±0.87	82.55±3.34	90.68±3.22	87.38±0.13	**99.67±0.24**
C11	92.38±0.69	98.33±0.00	98.81±3.35	97.61±0.60	99.41±0.60	98.89±1.25	99.52±0.15	**99.84±0.11**
C12	77.20±6.23	95.13±0.50	41.75±9.32	20.48±10.64	47.92±6.17	76.14±3.39	38.37±0.23	**95.56±0.41**
C13	99.89±0.09	**100.00±0.00**	99.35±0.27	93.64±2.86	95.36±3.99	99.71±0.18	99.89±0.01	**100.00±0.00**
OA	82.79±0.03	91.98±2.27	82.69±0.07	80.00±1.35	83.76±1.07	89.22±0.98	86.09±0.11	**98.41±0.18**
AA	77.97±0.31	91.22±0.59	79.86±0.73	77.88±0.99	81.68±1.82	86.45±1.13	83.71±0.13	**98.23±0.21**

and fewer misclassification pixels can be observed in the classification map of the proposed DTDEIN model compared with EGNN and other competitors.

D. Impact of the Number of Labeled Examples

In this experiment, the proposed DTDEIN is investigated under different numbers of labeled examples. In specific, a series of experiments are conducted with labeled samples' number for each class varying from 1 to 13 on three datasets. The performance of the proposed model on three datasets is reported in Fig. 10. It can be observed that, as the labeled samples increase, the accuracy gradually improves. The accuracy of the Botswana dataset can reach more than 95% with one labeled sample. In addition, when the number of samples of each type rises to 7 on Botswana datasets and 9 on KSC and Houston datasets, the accuracy has been significantly improved with stable performance, which implied that our proposed model can achieve high accuracy with very few samples. All observations indicate the effectiveness and stability of our DTDEIN method.

E. Semisupervised Few-Shot Classification

In this section, we designed a semisupervised experiment where the model is trained with the training set and applied to the new dataset without fine-tuning to verify the generalization of the proposed method. In this experiment, the EGNN [48] model is chosen as a comparison. In our semisupervised experiment, we use Pavia Centre and KSC as the training

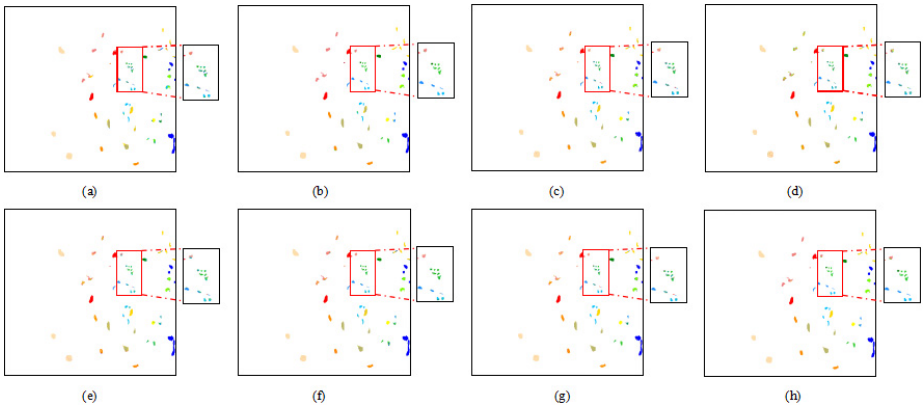


Fig. 8. Classification maps of the KSC with compared methods. (a) PN. (b) SSRN. (c) FADCNN. (d) GNN. (e) EGNN. (f) EGIN. (g) TDEGNN. (h) DTDEIN.

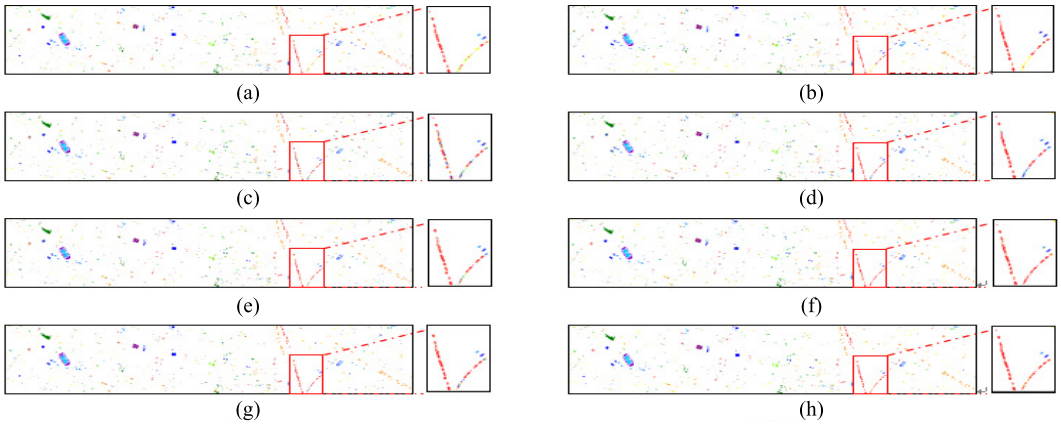


Fig. 9. Classification maps of the Houston with compared methods. (a) PN. (b) SSRN. (c) FADCNN. (d) GNN. (e) EGNN. (f) EGIN. (g) TDEGNN. (h) DTDEIN.

TABLE V
CLASSIFICATION PERFORMANCE COMPARISON WITH ALL THE COMPARED METHODS OF THE HOUSTON

Class	PN	SSRN	FADCNN	GNN	EGNN	EIGN	TDEGNN	DTDEIN
C1	89.18±2.93	78.02±0.96	84.54±4.36	71.70±4.24	89.85±0.96	88.85±3.4	95.20±2.16	**96.48±0.48**
C2	76.79±4.57	76.95±6.62	83.82±4.63	73.13±3.03	88.60±0.16	89.00±2.40	88.92±2.18	**96.41±0.24**
C3	97.37±0.18	91.83±0.15	93.40±2.30	89.96±2.87	90.67±0.43	99.36±0.08	91.97±0.07	**100±0.00**
C4	90.70±0.56	71.55±1.61	83.85±0.89	75.72±4.51	83.73±0.76	86.21±1.97	69.13±1.78	**98.15±0.12**
C5	**97.85±0.87**	89.41±3.34	88.41±0.81	87.84±0.12	89.33±0.12	96.86±1.37	85.83±1.21	96.86±0.85
C6	86.77±1.57	76.16±1.70	68.15±2.00	82.46±8.16	70.15±0.00	84.16±0.16	80.00±0.93	**87.08±0.31**
C7	44.37±4.95	62.39±3.32	59.55±9.99	52.84±9.35	45.9±1.66	64.36±4.50	48.34±1.63	**76.34±1.03**
C8	59.43±4.36	51.89±0.44	54.26±4.26	51.29±0.65	58.40±0.12	60.41±2.61	60.04±1.09	**62.62±0.2**
C9	67.89±2.20	67.29±1.16	50.48±8.71	33.79±3.99	58.75±0.44	66.14±0.96	71.73±0.17	**86.67±1.52**
C10	75.12±3.37	81.95±3.87	82.11±3.63	81.01±1.27	86.11±0.37	97.72±0.08	97.96±0.16	**99.19±0.17**
C11	59.87±1.28	71.70±4.33	60.41±9.53	73.52±1.30	48.79±1.10	80.33±0.33	70.28±1.14	**92.47±1.82**
C12	73.56±6.40	69.55±2.48	79.57±4.06	85.32±4.91	83.22±0.41	90.59±0.81	83.21±0.21	**98.22±1.38**
C13	71.86±3.48	65.35±0.53	62.37±7.57	68.87±0.54	54.27±0.54	79.53±5.12	76.33±4.27	**90.83±0.96**
C14	89.26±1.00	97.08±0.58	83.06±4.09	91.36±6.20	86.92±0.24	92.64±3.39	88.32±2.22	**96.96±2.8**
C15	83.33±4.61	66.52±4.25	89.55±1.07	85.45±2.58	88.71±0.23	94.47±0.99	95.45±0.53	**95.61±0.38**
OA	75.67±1.82	73.31±0.01	74.24±0.34	71.23±134	74.44±0.15	83.61±0.42	78.87±0.67	**91.12±0.54**
AA	77.57±1.70	74.51±0.09	74.90±0.15	73.62±0.57	74.90±0.16	84.71±0.44	80.18±0.68	**91.59±0.52**

dataset to train the model, and Salinas and Botswana are used as the test dataset to evaluate the generalization ability.

In specific, for our DTDEIN, we employ the five-way five-shot setting for training and five-way two-shot setting for

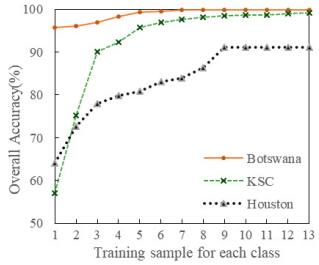


Fig. 10. Overall accuracies of our method with different numbers of the labeled examples per class for the three datasets.

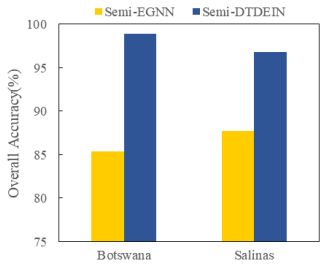


Fig. 11. Accuracy value of the Botswana and Salinas scenes with semi-DTDEIN model and semi-EGNN model.

TABLE VI
ABLATION STUDY ON THE THREE HSI DATASETS

Module	DEIGN			√	√	√
	EIGN	√	√			
	IDD		√		√	√
	PCD			√		√
HSI Data	Botswana	98.30	99.30	99.66	99.60	99.88
	KSC	88.60	94.72	94.87	95.47	98.66
	Houston	83.19	88.50	89.07	89.27	91.12

testing and seven labeled samples for each type of test set. For EGNN, a five-way five-shot setting is used for both training and testing. The comparison results on Salinas and Botswana datasets are delivered in Fig. 11, in which semi-DTDEIN and semi-EGNN represent that used in semisupervised. As can be seen from Fig. 11, the proposed method (semi-DTDEIN) is much better than semi-EGNN by a big gap. More specifically, the proposed method achieved an OA of 98.92% for Botswana, and 96.76% for Salinas is higher than the OA of 80.60% for Botswana and 81.34% for Salinas achieved by EGNN in the semisupervised experiment. Therefore, it proves that the proposed method has a strong generalization ability.

F. Ablation Study

To verify the contribution of the double-network mechanism (spatial network and spectral network) and DTD block in the presented approach to the ultimate classification result, the ablation study is carried out on the three datasets with the edge-inferring module as the baseline part. Individually, Table VI exhibits the results of the experiments, where DEIGN represents double-network structure, and the EIGN represents spatial network structure. All methods involving DTD block decide label according to the priority label decision strategy mentioned in Section I.

1) Double-Network: To demonstrate the contribution of the double-network clearly, Table VI is demonstrated as two parts in different colors separately. The red part shows the method using a single network (spatial network), and the blue part shows the method using the double-network structure. It is observed that the OA in blue is more competitive than in red, especially for the Houston and the KSC data. In specific, the DEIGN + PCD model is better than the model EIGN, and the DEIGN + IDD approach also outperformed the model EIGN + IDD. Therefore, we conclude that the double-network presented in this article is effective, especially for the dataset with low spatial resolution and more noise.

2) Dynamic Task-Guided Self-Diagnosis: To show the effectiveness of the DTD block, we conduct another experiment to compare the classification performance provided with or without the DTD block. As can be seen in Table VI, all the models with the DTD block show better performance than the ones without it. In specific, the OAs of the single network structure model in the first two adjacent columns (EIGN and EIGN + IDD) testify that the IDD block is more conducive to the simpler HSIC model. The OA improves 1.0% for Botswana, 6.12% for KSC, and 5.31% for Houston compared with the great role in the single network structure. For the three columns on the right in the table, the double-network structure with PCD yielded excellent performance. With IDD participating in the double-network model (DEIGN + PCD), the OA enhances 0.28% for Botswana, 3.79% for KSC, and 2.05% for Houston.

IV. CONCLUSION

In this article, a novel edge-inferring GNN for metalearning with DTD is proposed for HSFSC. Specifically, the edge-labeling network performs edge inferring to perform clustering explicitly with both instance-level similarity and the distribution-level similarity, which has great advantages in exploring complex data relationships with limited labeled samples. Besides, aiming at spatial and spectral features of HSI, the edge-inferring architecture is embodied in two edge-inferring models, which jointly contributes to classification in the test phase. Furthermore, we design a DTD for label prediction accurately at the metatest phase, which joins two different levels of information to keep the most reliable classification results.

REFERENCES

[1] Y. Chen, X. Zhao, and X. Jia, "Spectral-spatial classification of hyperspectral data based on deep belief network," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 8, no. 6, pp. 2381–2392, Jun. 2015.

[2] B.-C. Kuo, C.-S. Huang, C.-C. Hung, Y.-L. Liu, and I.-L. Chen, "Spatial information based support vector machine for hyperspectral image classification," in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, Jul. 2010, pp. 832–835.

[3] P. O. Gislason, J. A. Benediktsson, and J. R. Sveinsson, "Random Forests for land cover classification," *Pattern Recognit. Lett.*, vol. 27, no. 4, pp. 294–300, 2006.

[4] J. A. Benediktsson, P. H. Swain, and O. K. Ersoy, "Neural network approaches versus statistical methods in classification of multisource remote sensing data," in *Proc. 12th Can. Symp. Remote Sens. Geosci. Remote Sens. Symp.*, vol. 2, Jul. 1989, pp. 489–492.

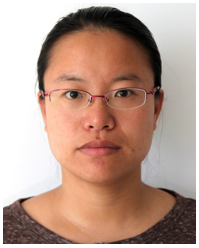
[5] Z. Liu, P. Luo, X. Wang, and X. Tang, "Deep learning face attributes in the wild," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 3730–3738.
[6] F. Zhang, B. Du, and L. Zhang, "Saliency-guided unsupervised feature learning for scene classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 4, pp. 2175–2184, Apr. 2015.
[7] P. Zhong, Z. Gong, S. Li, and C.-B. Schönlieb, "Learning to diversify deep belief networks for hyperspectral image classification," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 55, no. 6, pp. 3516–3530, Jun. 2017.
[8] L. Mou, P. Ghamisi, and X. X. Zhu, "Deep recurrent neural networks for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 7, pp. 3639–3655, Jul. 2017.
[9] S. Li, W. Song, L. Fang, Y. Chen, P. Ghamisi, and J. A. Benediktsson, "Deep learning for hyperspectral image classification: An overview," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 9, pp. 6690–6709, Apr. 2019.
[10] P. Ghamisi, J. Plaza, and Y. S. Chen, "Advanced spectral classifiers for hyperspectral images: A review," *IEEE Geosci. Remote Sens. Mag.*, vol. 5, no. 1, pp. 8–32, Mar. 2017.
[11] Y. Chen, H. Jiang, C. Li, X. Jia, and P. Ghamisi, "Deep feature extraction and classification of hyperspectral images based on convolutional neural networks," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 10, pp. 6232–6251, Jul. 2016.
[12] S. Hao, W. Wang, Y. Ye, T. Nie, and L. Bruzzone, "Two-stream deep architecture for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 4, pp. 2349–2361, Apr. 2018.
[13] W. Hu, Y. Huang, L. Wei, F. Zhang, and H. Li, "Deep convolutional neural networks for hyperspectral image classification," *J. Sensors*, vol. 2015, pp. 1–12, Jan. 2015.
[14] Y. Chen, Z. Lin, X. Zhao, G. Wang, and Y. Gu, "Deep learning-based classification of hyperspectral data," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 7, no. 6, pp. 2094–2107, Jun. 2014.
[15] M. Zhang, W. Li, and Q. Du, "Diverse region-based CNN for hyperspectral image classification," *IEEE Trans. Image Process.*, vol. 27, no. 6, pp. 2623–2634, Jun. 2018.
[16] S. Yu, S. Jia, and C. Xu, "Convolutional neural networks for hyperspectral image classification," *Neurocomputing*, vol. 219, pp. 88–98, Jan. 2017.
[17] X. Ma, H. Wang, J. Geng, and J. Wang, "Hyperspectral image classification with small training set by deep network and relative distance prior," in *Proc. IEEE Int. Geosci. Remote Sens. Symp. (IGARSS)*, Jul. 2016, pp. 3282–3285.
[18] W. Li, G. Wu, F. Zhang, and Q. Du, "Hyperspectral image classification using deep pixel-pair features," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 2, pp. 844–853, Feb. 2017.
[19] B. Fang, Y. Li, H. Zhang, and J. C.-W. Chan, "Semi-supervised deep learning classification for hyperspectral image based on dual-strategy sample selection," *Remote Sens.*, vol. 10, no. 4, p. 574, Apr. 2018.
[20] Y. Zhan, D. Hu, and Y. Wang, "Semisupervised hyperspectral image classification based on generative adversarial networks," *IEEE Geosci. Remote Sens. Lett.*, vol. 15, no. 2, pp. 212–216, Feb. 2018.
[21] K. Makantasis, A. D. Doulamis, N. D. Doulamis, and A. Nikitakis, "Tensor-based classification models for hyperspectral data analysis," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 12, pp. 6884–6898, Dec. 2018.
[22] K. Makantasis, A. Doulamis, N. Doulamis, A. Nikitakis, and A. Voulodimos, "Tensor-based nonlinear classifier for high-order data analysis," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Apr. 2018, pp. 2221–2225.
[23] J. Snell, K. Swersky, and R. Zemel, "Prototypical networks for few-shot learning," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, Mar. 2017, pp. 1–11.
[24] F. Sung, Y. Yang, L. Zhang, T. Xiang, P. H. S. Torr, and T. M. Hospedales, "Learning to compare: Relation network for few-shot learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 119–1208.
[25] C. Lemke, M. Budka, and B. Gabrys, "Metalearning: A survey of trends and technologies," *Artif. Intell. Rev.*, vol. 44, pp. 117–130, Jul. 2013.
[26] M. Jamal, G.-J. Qi, and M. Shah, "Task agnostic meta-learning for few-shot learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, May 2019, pp. 11711–11719.
[27] B. Xi, J. Li, Y. Li, and Q. Du, "Semi-supervised graph prototypical networks for hyperspectral image classification," in *Proc. IEEE Int. Geosci. Remote Sens. Symp. IGARSS*, Jul. 2021, pp. 2851–2854.
[28] B. Liu, X. Yu, A. Yu, P. Zhang, G. Wan, and R. Wang, "Deep few-shot learning for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 4, pp. 2290–2304, Apr. 2019.
[29] H. Tang, Y. Li, X. Han, Q. Huang, and W. Xie, "A spatial–spectral prototypical network for hyperspectral remote sensing image," *IEEE Geosci. Remote Sens. Lett.*, vol. 17, no. 1, pp. 167–171, Jan. 2020.
[30] C. Zhang, J. Yue, and Q. Qin, "Global prototypical network for few-shot hyperspectral image classification," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 13, pp. 4748–4759, 2020.
[31] F. Monti, M. Bronstein, and X. Bresson, "Geometric matrix completion with recurrent multi-graph neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 3697–3707.
[32] F. Scarselli, M. Gori, A. C. Tsoi, M. Hagenbuchner, and G. Monfardini, "Computational capabilities of graph neural networks," *IEEE Trans. Neural Netw.*, vol. 20, no. 1, pp. 81–102, Jan. 2009.
[33] M. Gori, G. Monfardini, and F. Scarselli, "A new model for learning in graph domains," in *Proc. IEEE Int. Joint Conf. Neural Netw.*, vol. 2, Jul. 2005, pp. 729–734.
[34] G. Camps-Valls, T. V. B. Marsheva, and D. Zhou, "Semi-supervised graph-based hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 45, no. 10, pp. 3044–3054, Oct. 2007.
[35] Y. Gao, R. Ji, P. Cui, Q. Dai, and G. Hua, "Hyperspectral image classification through bilayer graph-based learning," *IEEE Trans. Image Process.*, vol. 23, no. 7, pp. 2769–2778, Jul. 2014.
[36] W. Wang, L. Ma, M. Chen, and Q. Du, "Joint correlation alignment-based graph neural network for domain adaptation of multitemporal hyperspectral remote sensing images," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 14, pp. 3170–3184, 2021.
[37] J. Bruna, W. Zaremba, A. Szlam, and Y. LeCun, "Spectral networks and locally connected networks on graphs," 2013, *arXiv:1312.6203*.
[38] M. Defferrard, X. Bresson, and P. Vandergheynst, "Convolutional neural networks on graphs with fast localized spectral filtering," in *Proc. Adv. Neural Inf. Process. Syst.*, 2016, pp. 3844–3852.
[39] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," in *Proc. Int. Conf. Learn. Represent.*, 2017, pp. 1–14.
[40] S. Wan, C. Gong, P. Zhong, B. Du, L. Zhang, and J. Yang, "Multiscale dynamic graph convolutional network for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 5, pp. 3162–3177, May 2020.
[41] D. Hong, L. Gao, J. Yao, B. Zhang, A. Plaza, and J. Chanussot, "Graph convolutional networks for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 7, pp. 5966–5978, Jul. 2021.
[42] Y. Ding, X. Zhao, Z. Zhang, W. Cai, N. Yang, and Y. Zhan, "Semi-supervised locality preserving dense graph neural network with ARMA filters and context-aware learning for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, pp. 1–12, 2022, doi: 10.1109/TGRS.2021.3100578.
[43] X. Tong, J. Yin, B. Han, and H. Qv, "Few-shot learning with attention-weighted graph convolutional networks for hyperspectral image classification," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Oct. 2020, pp. 1686–1690.
[44] V. G. Satorras, G. Victor, and B. Joan, "Few-shot learning with graph neural networks," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2018, pp. 1–13.
[45] S. Kim, S. Nowozin, P. Kohli, and C. D. Yoo, "Higher-order correlation clustering for image segmentation," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, 2011, pp. 1–9.
[46] J. Kim, T. Kim, S. Kim, and C. D. Yoo, "Edge-labeling graph neural network for few-shot learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 11–20.
[47] C. Yu, R. Han, M. Song, C. Liu, and C.-I. Chang, "Feedback attention-based dense CNN for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, pp. 1–16, 2022.

Chunyan Yu (Member, IEEE) received the Ph.D. degree in environmental engineering from Dalian Maritime University, Dalian, China, in 2012.

She is currently an Associate Professor with the Information Science and Technology College, Dalian Maritime University. Her research interests include image segmentation, hyperspectral image classification, and pattern recognition.

Jiahui Huang received the B.S. degree in computer science and technology from Shandong Agricultural University, Tai'an, China, in 2020. She is currently pursuing the M.A. degree in computer science and technology with Dalian Maritime University, Dalian, China.

Her research interests include few-shot hyperspectral image processing and deep learning.



Meiping Song (Member, IEEE) received the Ph.D. degree from the College of Computer Science and Technology, Harbin Engineering University, Harbin, China, in 2006.

From 2013 to 2014, she was a Visiting Associate Research Scholar with the University of Maryland at Baltimore County, Baltimore, MD, USA. She is currently a Professor with the College of Information Science and Technology, Dalian Maritime University, Dalian, China. Her research includes remote sensing and hyperspectral image processing.

Yulei Wang (Member, IEEE) received the B.S. and Ph.D. degrees in signal and information processing from Harbin Engineering University, Harbin, China, in 2009 and 2015, respectively.

She is awarded by the China Scholarship Council in 2011 as a joint Ph.D. Student to study at the Remote Sensing Signal and Image Processing Laboratory, University of Maryland at Baltimore County, Baltimore, MD, USA, for two years. She is currently an Associate Professor with the Center for Hyperspectral Imaging in Remote Sensing (CHIRS), Information Science and Technology College, Dalian Maritime University, Dalian, China. Her research interests include hyperspectral image processing and vital signs' signal processing.

Chein-I Chang (Life Fellow, IEEE) received the B.S. degree in mathematics from Soochow University, Taipei, Taiwan, in 1973, the M.S. degree in mathematics from the Institute of Mathematics, National Tsing Hua University, Hsinchu, Taiwan, in 1975, the M.A. degree in mathematics from the State University of New York at Stony Brook, Stony Brook, NY, USA, in 1977, the M.S. and M.S.E.E. degrees from the University of Illinois at Urbana–Champaign, Urbana, IL, USA, in 1980 and 1982, respectively, and the Ph.D. degree in electrical engineering from the University of Maryland, College Park, MD, USA, in 1987.

He has been with the University of Maryland at Baltimore County (UMBC), Baltimore, MD, since 1987, where he is currently a Professor with the Department of Computer Science and Electrical Engineering. He has been holding the Chang Jiang Scholar Chair Professorship and has been the Director of the Center for Hyperspectral Imaging in Remote Sensing (CHIRS), Dalian Maritime University, Dalian, China, since 2016, and the Hua Shan Scholar Chair Professorship at Xidian University, Xi'an, China, since 2016. He has been a Feng-Tai Chair Professor with the National Yunlin University of Science and Technology, Douliu, Taiwan, since 2017. His research interests include remote sensing and hyperspectral image processing.