# Multiscale Graph Sample and Aggregate Network With Context-Aware Learning for Hyperspectral Image Classification

Yao Ding, Xiaofeng Zhao, Zhili Zhang, Wei Cai, and Nengjun Yang

***Abstract*—Recently, graph convolutional network (GCN) has achieved promising results in hyperspectral image (HSI) classification. However, GCN is a transductive learning method, which is difficult to aggregate the new node. Besides, the existing GCN-based methods divide graph construction and graph classification into two stages ignoring the influence of constructed graph error on classification results. Moreover, the available GCN-based methods fail to understand the global and contextual information of the graph. In this article, we propose a novel multiscale graph sample and aggregate network with a context-aware learning method for HSI classification. The proposed network adopts a multiscale graph sample and aggregate network (graphSAGE) to learn the multiscale features from the local regions graph, which improves the diversity of network input information and effectively solves the impact of original input graph errors on classification. By employing a context-aware mechanism to characterize the importance among spatially neighboring regions, deep contextual and global information of the graph can be learned automatically by focusing on important spatial targets. Meanwhile, the graph structure is reconstructed automatically based on the classified objects as network training, which is able to effectively reduce the influence of the initial graph error on the classification result. Extensive experiments are conducted on three real HSI datasets, which are demonstrated to outperform the compared state-of-the-art methods.**

***Index Terms*—Deep contextual, graph convolutional network (GCN), hyperspectral image classification, multiscale graph.**

## I. Introduction

HYPERSPECTRAL images (HSIs) contain abundant spectral information and spatial information of ground objects simultaneously, which make it possible to distinguish the targets with different materials [1], [2]. As a result, HSI classification, which aims to categorize each image pixel into a certain class, has caused wide attention in various fields, such as military target detection, agriculture monitoring, and disaster prevention and control.

Over the last few decades, extensive research works have been conducted on HSIs classification, which can be summarized into two categories: traditional methods and deep learning methods. The traditional algorithms mainly concentrate on exploring more handcrafted features [3], [4] and transforming original spectral signatures into a learned new feature space [5]–[7]. Besides, some machine learning methods have been adopted for HSIs classification, for instance, K-nearest neighbor [8], random forest [9], and support vector machine (SVM) [10]. However, traditional methods are all based on the handcrafted spectral-spatial features that heavily depend on professional expertise and are quite empirical [11].

To address this shortcoming, deep learning has attracted great attention for HSI classification. Deep learning methods obtain automatically high-level abstract representations by aggregating low-level information, which can avoid complex feature extraction engineering [12]. Recently, recurrent neural networks (RNNs) [13]–[15] have demonstrated their potentials in modeling the spectral-spatial features of HSI. In [13], RNN was first proposed for spectral classification. Besides, Ma *et al*. adopt contextual deep learning to learn the spectral-spatial features [16]. In addition, convolutional neural network (CNN) [17]–[20] has been widely employed in HSI classification tasks. In [19], [21], and [22], the HSIs are classified by using different dimensional convolutions. In [23], the residual blocks were employed to improve the representation ability of CNN, which is capable of extracting spectral signatures and spatial contexts of HSI to improve the classification rate. In [1], a multilayer CNN was adopted to encode spectral-spatial information. Although CNN methods have achieved good performance in some fields, they still suffer from some defects. First of all, CNN needs a lot of training labels, time, and calculation. The HSI has the characteristics of the small amount of label data, which is the obstacle to the CNN network during the weight training. Afterward, the CNN kernel is designed to perform in a regular square, so it cannot adaptively capture the geometric variations of different object regions in an HSI. Last but not the least, the weights of the CNN convolution kernel are fixed. As a result, there will lead to edge missing phenomenon in the process of feature extraction, and misclassifications will probably happen [12].

Different from of CNNs, the graph convolutional network (GCN) conducts semisupervised learning on graph-structured data (social network data and graph-based representations of molecules [1], [25], [26]) and can operate on graph signal directly via a variant of CNNs. Sha *et al*. applied the graph attention network (GAN) to hyperspectral classification [27];

Manuscript received February 18, 2021; revised March 19, 2021 and March 28, 2021; accepted April 13, 2021. Date of publication April 22, 2021; date of current version May 17, 2021. This work was supported in part by the National Natural Science Foundation of China under Grant 41404022 and in part by the National Natural Science Foundation of Shanxi Province under Grant 2015JM4128. *(Xiaofeng Zhao is co-first author.)* (*Corresponding author: Xiaofeng Zhao*.)

The authors are with the Xi'an Research Institute of High Technology, Xi'an 710000, China (e-mail: dingyao.88@outlook.com; xife_zhao@163.com; 157918018@qq.com; caiwei_bu@163.com; yangbcpl@163.com).

Digital Object Identifier 10.1109/JSTARS.2021.3074469

however, the network only uses the attention mechanism and does not learn the multiscale information of the graph. In [28], Wan *et al.* employed a multiscale GCN to extract multiscale graph features; however, the attention mechanism was not used to select multiscale features according to different tasks. Hong *et al*. proposed a graph convolution classification method combining graph convolution and convolution neural network [29], which opened up new ideas for HSI classification. Nevertheless, there still exist some common shortcomings in these GCN-based methods. The existing methods choose GCN as the basic method to build a graph processing network. However, GCN is a whole graph training method, which will bring a huge amount of computation. In the case of selecting some pixels in an HSI for classification, GCN whole image training method will bring huge computational waste. In another word, the GCN-based network will spend most of the computational power on nonclassified pixels. Besides, the existing methods divide graph construction and graph classification into two stages. They do not consider the influence of constructed graph error on classification results and do not automatically build diverse graph networks for different classification objects. Finally, the above-mentioned methods fail to understand the contextual information of the graph since they consider every graph node has the same influence on a classified node while disregarding the differences.

In response to previous problems and further boost the performance of HSI classification, we propose the multiscale graph sample and aggregate network with context-aware learning (MSAGE-CAL) method, where global contextual information among superpixels can be automatically learned in an end-to-end training framework. Different from the GCN method, the proposed network adopts the graphSAGE method, which can reduce the amount of calculation, to solve the problem of calculation consumption. To improve the diversity of network input information and solve the impact of initial input graph errors on classification, the network employs a multiscale learning method. The deep contextual and the global information of the graph can be learned automatically by focusing on important spatial targets via graph attention mechanism so that the network learning is more targeted and the classification is more efficient. When the network is trained, backpropagation can be used to feed the error back to the graphSAGE learning network, readjust the network coefficients, process the multiscale graph, and reconstruct graphs for different classification targets; as a result, different object appearances can be better represented.

To sum up, the main contributions in this article are as follows.

1) MSAGE-CAL for a semisupervised method is proposed.
2) Multiscale graphSAGE convolution is employed to extensively exploit the spatial information and reduce the computational complexity.
3) Context-aware learning and image reconstruction are combined to make pixel features more expressive.

## II. Related Work

Many researchers have published their methods to classify HSIs. In this part, we will review some representative works since they have a lot of relationships with our work.

### A. Deep-Learning-Based HSI Classification

Deep learning has achieved great success in many applications [30]. Recently, deep learning also has a wide range of applications in HSI classification. One main advantage is that deep learning techniques can automatically learn effective feature representations for a problem domain, thereby avoiding complicated handcrafted feature engineering [11]. Chen *et al*. first attempted to utilize stacked autoencoder high-level feature extraction [31]. Subsequently, in [32], the restricted Boltzmann machine and deep belief network were employed for HSI feature extraction and pixel classification, which can retain the good information containing in the original data. Meanwhile, RNN [13]–[15] and generative adversarial networks (GAN) [33], [34] have also begun to be applied in HSI classification. Among these deep learning methods, CNN has demonstrated its outstanding performance for HSI classification because it has fewer parameters than fully connected networks with the same hidden units. Hu *et al*. [19] utilized CNN to extract spectral features and got better performance than SVM. To extract the spectral-spatial features of hyperspectral, many CNN-based methods have emerged. For example, in [35], a two-channel deep CNN was used to extract spectral-spatial features of HSIs. Alternatively, Slavkovikj *et al.* [36] proposed a two-dimensional (2-D) CNN to processing the original hyperspectral data. Additionally, some authors proposed 1-D+2-D CNN [37] architecture for HSI classification. In addition to the 1-D and 2-D architecture, 3-D CNN has shown potentials for HSIs classification, which is capable of learning to recognize more complex 3-D patterns of HSI and needs fewer parameters and layers than 2-D+1-D CNN. For instance, Li *et al*. [38] proposed to use 3-D CNN to handle the hyperspectral cube, which is able to learn spectral-spatial features via 3-D convolutions. Although spectra-spatial features can be extracted by CNN-based automatically, they simply apply the fixed convolution kernels to different regions in an HSI, which may result in undesirable misclassifications. To deal with this problem, people try to use different convolution kernels according to different features. Feng *et al.* [39] adopted different sizes and locations of spatial windows according to sample-specific distribution, which brings new ideas to the classification of CNN-based methods.

### B. Graph Neural Network (GNN)

CNN has achieved great success for graph data processing. However, the CNN algorithm is computationally expensive and runs inefficiently on large-scale graphs. Therefore, a GNN was first proposed by Bruna *et al*. [40], where the neighborhood of every graph node is convolved and a node-level output is produced. After that, people have conducted extensive research works on graph convolution and achieved advanced results [41]. The graph convolutions can be roughly divided into two groups, namely spectral convolutions and spatial convolutions. Spectral convolutions perform convolution that transforms the graph node representations into spectral domain via graph Fourier transform. For instance, in [42], a formulation of CNNs is proposed for spectral graph theory. The spatial-based GNN defines the graph convolution operator based on the neighborhood aggregation

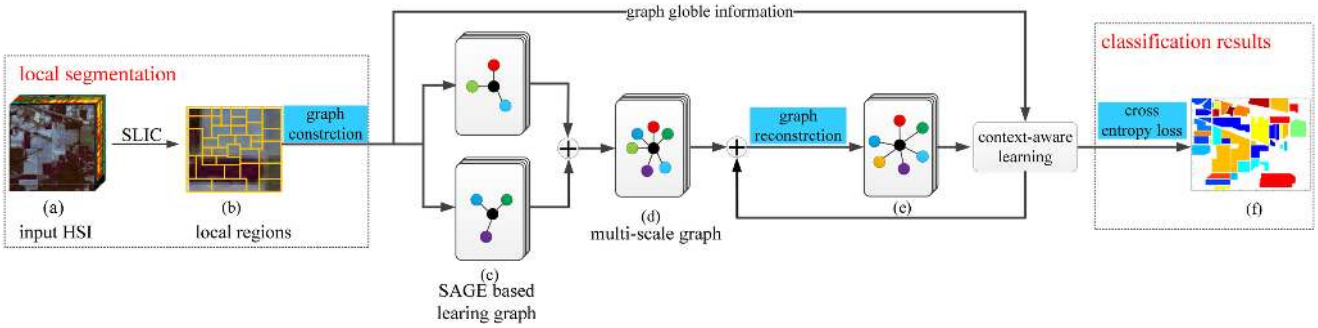


Fig. 1. Overview of MSAGE-CAL network. (a) Input HSI. (b) Local regions (superpixels) segmented by the SLIC algorithm. In (c), the circles and lines represent the superpixel (graph node) and edges, different colors of the nodes represent different land cover types, and the input of the network is spectral characteristic of each node. (c) and (d) Multiscale graph learning mechanism, where the multiscale information can be learned with SAGE mechanism automatically and pairwise importance among the superpixels can be learned with context-aware learning mechanism. (e) Reconstructed graph, where the topological graph information is automatically reconstructed based on contextual-aware learning and loss backpropagation.

[43]. In [26], Hamilton *et al*. proposed an inductive framework called "GraphSAGE," where the weighting function is defined as various aggregators over neighboring nodes. Velickovic *et al.* [44] presented a GAT, which can adaptively learn the weighting function via a self-attention mechanism.

With the fast development of GNN, GNNs have attracted much attention for many fields of application, such as semantic segmentation [45] and natural language processing [46]. Besides, GNNs have adopted for HSIs classification [28], [29]. However, all these works utilize a fixed graph and cannot precisely reflect the intrinsic relationship among the pixels.

## III. Proposed Method

In this section, we will present MSAGE-CAL for HSI semisupervised classification (see Fig. 1). First, simple linear iterative clustering (SLIC) [47] algorithm is adopted to segment the entire HSI [see Fig. 1(a)] into a small number of compact superpixels [see Fig. 1(b)]. Then, multiple spatial levels graphs [see Fig. 1(c)] are constructed over superpixels via graphSAGE. Subsequently, topological graph information [see Fig. 1(d)] is automatically reconstructed via context-aware learning. Finally, the classification result is produced [see Fig. 1(f)] by presenting contextual information via cross-entropy loss. In the following, we will detail the critical steps of MSAGE-CAL by presenting the local region segmentation technique (see Section III-A), elaborating the SAGE-based multiscale graph learning (see Section III-B), describing the context-aware learning and graph reconstruction (see Section III-C), and explaining the MSAGE-CAL manipulation (see Section III-D).

### A. Local Region Segmentation

HSI contains a large number of pixels in the spatial dimension, a huge amount of computation is needed to convolution and classification, sometimes it is unacceptable. To ameliorate this issue, we find neighbor pixels that have a large probability of belonging to the same land cover type. Therefore, the SLIC has adopted to segment the entire image into a small number of local regions and the pixels consisted of regions that have a strong spectral-spatial similarity. Concretely, SLIC conducts image region segmentation via iteratively growing the local clusters by employing a *k*-means algorithm. In this article, the local regions are treated as the graph node, which can significantly reduce the number of graph nodes and improve computational efficiency. Here, the average spectral signatures of the involved pixels in the node (local region) are taken as the feature vector of the node.

**Algorithm 1:** GraphSAGE Embedding Generation (i.e., Forward Propagation) Algorithm

**Input:** Graph $\boldsymbol{G} = (\boldsymbol{V}, \boldsymbol{E})$ ; input features $\{\boldsymbol{x}_v, \forall v \in \boldsymbol{V}\}$; the number of layers of the network $K$; weight matrices $\boldsymbol{W}^k$, $\forall k \in \{1, \ldots, K\}$; nonlinearity $\sigma$; mean aggregator functions AGG; neighborhood function $N : v \to 2^v$

**Output:** Vector representations for all $v \in \boldsymbol{V}$

$h_0 \leftarrow \boldsymbol{x}_v, \forall v \in \boldsymbol{V}$;
for $k = \{1, \ldots, K\}$ *do*
  for $v \in V$ do
    $\boldsymbol{h}^k_{\boldsymbol{N}(v)} \leftarrow \mathrm{AGG}(\{\boldsymbol{h}^{k-1}_u, \forall u \in \boldsymbol{N}(v)\})$;
    $\boldsymbol{h}^k_v \leftarrow \sigma(\boldsymbol{W}_k \cdot CONCAT(\boldsymbol{h}^{k-1}_v, \boldsymbol{h}^k_{N(v)}))$
  end
  $\boldsymbol{h}^k_v \leftarrow \frac{\boldsymbol{h}^k_v}{\|\boldsymbol{h}^k_v\|_2}$ , $v \in V$
end

**Output:** $\boldsymbol{z}_v \leftarrow \boldsymbol{h}^K_v, v \in V$

### B. SAGE-Based Multiscale Graph Learning

Traditional GCN transductive learning requires all nodes to participate in training to get node embedding and it cannot quickly get embedding of new nodes. In other words, GCN can only learn the information about neighboring nodes and cannot naturally generalize to the unknown vertices. Another major disadvantage of conventional GCN is that the graph is fixed throughout the convolution process, which will degrade the final classification performance if the input graph is not accurate [26]. To ameliorate these issues, graphSAGE (SAGE) algorithm is adopted to learn spatial scale information, which can improve the generalization ability of the model for new nodes. The SAGE forward propagation rule is expressed as Algorithm 1.

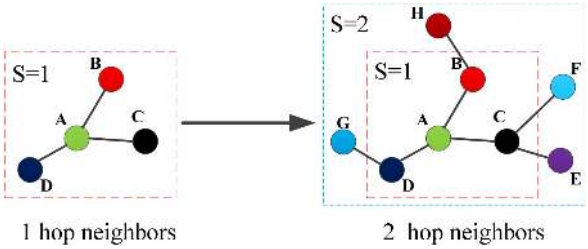


Fig. 2. SAGE aggregate mechanism in the proposed method.

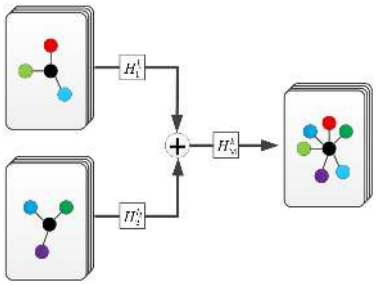

Fig. 3. Overview of multiscale spatial information in our method. Different branches are used to comprise different neighborhood scales. Different colors of the nodes represent different land cover types.

In Algorithm 1, $K$ is the number of layers of the network and also represents the number of hops of adjacent points that can be aggregated at each vertex; $\forall u$ is the eigenvector of the node $u$; $\{\boldsymbol{h}_u^{k-1}, \forall u \in \boldsymbol{N}(v)\}$ denotes the embedding of the neighbor $U$ of the node $V$ in the $k-1$ layer; and $\boldsymbol{h}_v^k$ represents the characteristic of all neighbors of node $v$ at the $k$ level. AGG can be expressed as $\mathrm{AGG} = \sum_{u \in N(v)} \frac{\boldsymbol{h}_u^{k-1}}{|N(v)|}$.

Multiscale information has been widely proved to be very useful for HSI classification [48], [49] because multiscale can get more spatial feature information. In the proposed method, the input graph is learned by graphSAGE (SAGE), and as mentioned above, we can adjust the number of layers of SAGE to control the number of hops that can be aggregated. Fig. 2 demonstrates 1-hop and 2-hop neighbors of a central example A. Then, the receptive field of A at the scale $S$ is formed as

$$\boldsymbol{H}^s\ (\boldsymbol{x}_i) = \sigma\left(\boldsymbol{H}^1\left(\boldsymbol{H}^{s-1}\left(\boldsymbol{x}_i\right), \boldsymbol{x}_s\right)\right) \tag{1}$$

where $S$ denotes the aggregate scale by SAGE, when $S = 1$, it represents the aggregation of the central node A and adjacent nodes, $S = 2$ denotes the aggregation of 2-hop neighbors and 1-hop graph. $\sigma$ is the activate function and $H^0\left(\boldsymbol{x}_i\right) = \boldsymbol{x}_i$, $H^1(\boldsymbol{x}_i)$ is the new node embedding of 1-hop neighbors of $\boldsymbol{x}_i$, as shown in Algorithm 1.

Then we use different branches to comprise different neighborhood scales. Fig. 3 exhibits the multiscale mechanism. The different branches are adopted to comprise different neighborhood scales, which can abstract the multiscale features from the original graph. And then implement a summation operation on branches. And The receptive field of $\boldsymbol{x}_i$ at a different scale is formed as

$$\boldsymbol{H}_M^k\ (x_i) = \boldsymbol{H}_1^{l_1}\ (\boldsymbol{x}_i) \cup \boldsymbol{H}_2^{l_2}\left(\boldsymbol{x}_i\right) \tag{2}$$

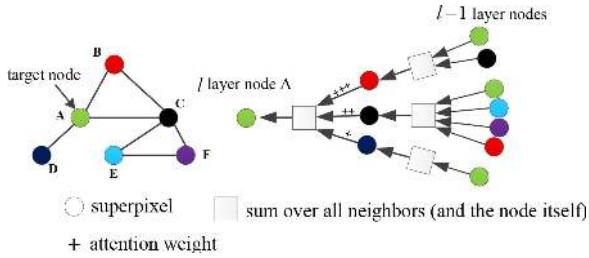


Fig. 4. Graph attention mechanism.

where 12 is the branch index, $l_1$, $l_2$ denote aggregate scales from the $l$ th layer in branches 1 and 2, $\max\ (l_1, l_2) = K$, and $M$ denotes the multiscale.

### C. Context-Aware Learning and Graph Reconstruction

To obtain global contextual features in the graph, the graph attention mechanism is added into the network to abstract different association degrees between different nodes, where the relationship between any two nodes in the graph is calculated via graph attention mechanism. Aiming at getting the corresponding transformation between input and output, a weight matrix is trained for all nodes: $\boldsymbol{W} \in \boldsymbol{R}^{\boldsymbol{F'} \times \boldsymbol{F}}$, which is the relationship between the input features $\boldsymbol{F}$ and the output features $\boldsymbol{F'}$. Node to node correlation can be learned through a network layer

$$e_{ij} = \left(\mathrm{LeakyReLU}\left(\boldsymbol{a}^T\left[\boldsymbol{W}\boldsymbol{x}_i || \boldsymbol{W}\boldsymbol{x}_j\right]\right)\right). \tag{3}$$

Equation (3) shows the importance of node $\boldsymbol{x}_j$ to node $\boldsymbol{x}_i$, $\boldsymbol{a}^T \in R^{2F}$ is the parameter vector of the network, $||$ denotes concatenation operation, and $\mathrm{LeakyReLU}(\cdot)$ is a nonlinear layer.

Then normalizing and converting $e_{ij}$ to a probability output $a_{ij}$ through a softmax function

$$a_{ij} = \frac{\exp\left(\mathrm{LeakyReLU}\left(\boldsymbol{a}^T\left[\boldsymbol{W}\boldsymbol{x}_i || \boldsymbol{W}\boldsymbol{x}_j\right]\right)\right)}{\sum_{k \in N_i} \exp\left(\mathrm{LeakyReLU}\left(\boldsymbol{a}^T\left[\boldsymbol{W}\boldsymbol{x}_i || \boldsymbol{W}\boldsymbol{x}_j\right]\right)\right)}. \tag{4}$$

The graph convolution output of each node can be expressed as follows:

$$\boldsymbol{x}_i^l = \sigma\left(\sum_{j \in N_i} a_{ij} \cdot \boldsymbol{W}^T \boldsymbol{x}_i^{l-1}\right) \tag{5}$$

where $\sigma$ is the activate function, $l$ denotes the network layer, and $a_{ij}$ is learned attention weight.

The graph attention mechanism in MSAGE-CAL is shown in Fig. 4. Global and contextual information can be learned from the graph via an attention mechanism. More importantly, the network would adjust the weight parameters according to the backpropagation loss as the network training. In other words, the network is context-aware. At the same time, the input multiscale graph is reconstructed that has a great influence on subsequent classification. The illustration of graph reconstruction is shown in Fig. 5.

### D. MSAGE-CAL Manipulation

Equation (2) shows multiscale graph learning, which is the input of the subsequent networks. Then a SAGE layer is adopted

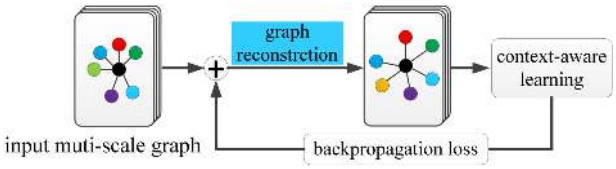


Fig. 5. Illustration of graph reconstruction considered by our method.

to reconstruct the multiscale graph, the reconstructed graph $\boldsymbol{H}_r^k$ can be expressed as follows:

$$\boldsymbol{H}_r^k\ (\boldsymbol{x}_i) = \sigma\left[\boldsymbol{S}\left(\boldsymbol{W}_r \cdot \boldsymbol{H}_M^k\left(\boldsymbol{x}_i\right)\right)\right] \tag{6}$$

where $\boldsymbol{S}$ denotes SAGE mechanism (see Algorithm 1), $r$ means graph reconstruction, $\boldsymbol{W}_r$ is the weight matrix of the network, which can be optimized based on backpropagation loss as network training. In the training process, the relationship between nodes in the graphSAGE layer generated graph will also change as the classification target changes so that we can construct different branches graph node relationships according to different classification targets. In concretely, the network can adjust $\boldsymbol{W}_r$ automatically to preserve the useful node features. Then context-aware learning is conducted on the network and the network output is expressed as follows:

$$\boldsymbol{O} = \boldsymbol{A}\left(\boldsymbol{H}_r^k\left(\boldsymbol{x}_i\right)\right) \tag{7}$$

where $\boldsymbol{A}$ is the context-aware learning mechanism and $\boldsymbol{O}$ is the output of MSAGE-CAL. In our network, the cross-entropy error is adopted to penalize the difference between the network output and the labels of the original labeled examples, namely

$$\boldsymbol{L} = -\sum_{z\in\mathbf{y_G}}\sum_{f=1}^{C}\boldsymbol{Y}_{zf}ln\boldsymbol{O}_{zf} \tag{8}$$

where $\mathbf{y_G}$ is labeled examples set, $C$ denotes the number of classes, and $\boldsymbol{Y}_{zf}$ is the label matrix. The implementation details of our MSAGE-CAL are shown in Algorithm 2.

## IV. Experimental Results

In this section, exhaustive experiments are conducted to validate the performances of the proposed MSAGE-CAL method, and the corresponding algorithm analyses are provided. First of all, MSAGE-CAL is compared with state-of-the-art approaches on three available HSI datasets, where four indices including overall accuracy (OA), average accuracy (AA), kappa coefficient ($\kappa$), and per-class accuracy are adopted to evaluate the proposed performance. Then, the performance of the MSAGE-CAL with the different number of labeled samples is analyzed on OA. Afterward, we show that multiscale graph learning is beneficial to improving performance. Finally, we demonstrate that context-aware learning and graph reconstruction manipulation are advantageous for better classification results.

### A. Dataset Description and Implementation

Three real benchmark datasets, i.e., the University of Pavia (PU), the Kennedy Space Center (KSC), and Salinas, are adopted to evaluate the performance of the MSAGE-CAL method. The PU dataset is employed to validate the algorithm's ability to classify details, the KSC dataset is to verify the classification ability of isolated small objects, and the Salinas dataset is to evaluate the algorithm's ability to classify objects of the different land cover types with a similar spectrum. The three datasets will be described as follows.

*1) University of Pavia (PU):* The first dataset PU is a part of hyperspectral data of the image of Pavia City, Italy, which was acquired by airborne reflection optics spectral imaging system (ROSIS) in 2003. The dataset contains 610 × 340 pixels and 103 bands, including a large number of background pixels, and 42 776 pixels can be applied to classification. The whole map contains nine kinds of features. The amount of data used for training and testing is given in Table I.

TABLE I
Numbers of Labeled and Unlabeled Pixels of All Classes in Pavia University Dataset

| ID | Class | #Labeled | #Unlabeled |
|---|---|---|---|
| 1 | Asphalt | 30 | 6601 |
| 2 | Meadows | 30 | 18619 |
| 3 | Gravel | 30 | 2069 |
| 4 | Trees | 30 | 3034 |
| 5 | Painted metal sheets | 30 | 1315 |
| 6 | Bare soil | 30 | 4999 |
| 7 | Bitumen | 30 | 1300 |
| 8 | Self-Blocking Bricks | 30 | 3652 |
| 9 | Shadows | 30 | 917 |

**Algorithm 2:** Proposed MSAGE-CAL for HSI classification

**Input**: Input image; number of epoch T;learning rate = 0.0005; dropout = 0.2; Adam gradient descent; python = 3.7; pytorch = 1.6.0.

1: Segment the whole image into local regions via SLIC algorithm;
2: Extract the superpixels features and construct the graph;
3: // Train the MSAGE-CAL model
4: **for** t = 1 to T **do**
5: // SAGE-based multiscale graph learning
6: Extract the local region features and construct the local region graph by Eq. (1);
7: Bach normalization, dropout and relu;
8: Construct the multiscale graph by (2);
9: // Context-aware learning and graph reconstruction
10: Reconstruct contextual graph by (6);
11: Bach normalization, dropout and relu;
12: Perform context-aware learning by (7);
13: Calculate the error term according to (8) and update the weight matrices using adam gradient descent;
14: **end for**
15: Conduct label prediction based on the trained network;

**Output**: Predicted label for each pixel.

TABLE II
NUMBERS OF LABELED AND UNLABELED PIXELS OF ALL CLASSES IN KSC DATASET

| ID | Class | #Labeled | #Unlabeled |
|---|---|---|---|
| 1 | Srub | 30 | 731 |
| 2 | Willow swamp | 30 | 213 |
| 3 | CP hammock | 30 | 226 |
| 4 | Slash pine | 30 | 222 |
| 5 | Oak/Broadleaf | 30 | 131 |
| 6 | Hardwood | 30 | 199 |
| 7 | Swamp | 30 | 75 |
| 8 | Graminoid | 30 | 401 |
| 9 | Spartina marsh | 30 | 490 |
| 10 | Cattail marsh | 30 | 374 |
| 11 | Salt marsh | 30 | 389 |
| 12 | Mud flats | 30 | 473 |
| 13 | Water | 30 | 897 |

TABLE III
NUMBERS OF LABELED AND UNLABELED PIXELS OF ALL CLASSES IN SALINAS DATASET

| ID | Class | #Labeled | #Unlabeled |
|---|---|---|---|
| 1 | Broccoli green weed 1 | 30 | 1979 |
| 2 | Broccoli green weed 2 | 30 | 3696 |
| 3 | Fallow | 30 | 1946 |
| 4 | Fallow rough plow | 30 | 1364 |
| 5 | Fallow smooth | 30 | 2648 |
| 6 | Stubble | 30 | 3929 |
| 7 | Celery | 30 | 3549 |
| 8 | Grapes untrained | 30 | 11241 |
| 9 | Soil vineyard develop | 30 | 6137 |
| 10 | Corn Senesced green weeds | 30 | 3248 |
| 11 | Lettuce romianes,4 wk | 30 | 1038 |
| 12 | Lettuce romianes,5 wk | 30 | 1897 |
| 13 | Lettuce romianes,6 wk | 30 | 886 |
| 14 | Lettuce romianes,7 wk | 30 | 1040 |
| 15 | Vineyard untrained | 30 | 7238 |
| 16 | Vineyard vertical trellis | 30 | 1777 |

*2) Kennedy Space Center:* The second dataset KSC was acquired by a 224-band airborne visible/infrared imaging spectrometer (AVIRIS) over the Kennedy Space Center, Florida, on March 23, 1996. After removing water absorption and low SNR bands, 176 bands were used for the analysis. This dataset contains 614 $\times$ 512 pixels and discrimination of land cover is difficult due to the similarity of spectral signatures for certain vegetation types. For classification purposes, 13 classes representing the various land cover types were defined for the site. The amount of data for training and testing is given in Table II.

*3) Salinas:* The third dataset Salinas was collected by the 224-band AVIRIS sensor over the region of Salinas Valley, CA, USA. The Salinas cover comprises 512 lines by 217 samples. After discarding 20 bands that cannot be reflected by water, 204 bands remain. These pixels are divided into 16 categories. The amount of data for training and testing is given in Table III .

### B. Experimental Settings

In the experiments, the proposed MSAGE-CAL algorithm is conducted via Pytorch with Adam's [50] optimizer. For all the three HSI datasets described in Section IV-A, we randomly selected 30 labeled pixels in each class for network training, and the remaining unlabeled pixels are used for network testing. Regarding network details, two neighborhood scales are employed to construct the multiscale graph. In concretely, branch 1 is a 1-scale aggregation graph and branch 2 is a 2-scale aggregation graph. The hyperparameters selection in our MSAGE-CAL is shown in Algorithm 2.

TABLE IV
ARCHITECTURE DETAILS OF PROPOSED NETWORK

| Module | Detail | |
|---|---|---|
| Local segmentation | SLIC | |
| Multiscale graph learning | SAGE (input node spectral dimension-64)<br>BN<br>ReLU | SAGE (input node spectral dimension-112)<br>BN<br>ReLU<br>SAGE (112-64)<br>BN<br>ReLU |
| | summation | |
| Multiscale graph processing | SAGE (64-32)<br>BN<br>ReLU | |
| Context-Aware learning | GAT (32)<br>BN | |
| Output | cross-entropy (target categories) | |

In order to validate the performance of MSAGE-CAL, the other five recent image classification methods are employed to conduct a comparison. Specifically, our network is compared with one CNN-based method, i.e., diverse-region-based deep CNN (DR-CNN) [51], and two GCN-based methods, i.e., spectral-spatial graph convolutional network ($S^2$GCN) [53] and spectral-spatial graph attention network networks ($S^2$GAT) [27]. Meanwhile, two traditional machine learning methods are also adopted, namely RBF-SVM and joint collaborative representation and SVM with decision fusion (JSDF) [52]. The value of $\gamma$ (the spread of the RBF kernel) and $C$ (controlling the magnitude of penalization during the model optimization) in RBF-SVM is optimized in the range of $\gamma = 2^{-3}, 2^{-2}, \ldots, 2^4$ and $C = 2^{-2}, 2^{-1}, \ldots, 2^4$. The architecture details of the proposed network is given in Table VI.

### C. Classification Results

In this section, to demonstrate the effectiveness of the proposed MSAGE-CAL, here we quantitatively and qualitatively evaluate the classification performance by comparing MSAGE-CAL with the aforementioned methods.

*1) Results on the PU Dataset:* The quantitative results achieved by different methods on the PU dataset are given in Table V, where the highest value in each row is highlighted in bold. From the table, we can observe that the proposed MSAGE-CAL achieves better results compared with other models in OA, AA, and $\kappa$ , which validates the effectiveness of the proposed multiscale graphSAGE network with context-aware learning. It is also notable that DR-CNN performs better than RBF-SVM, JSDF, and the nonlocal GCN method. This is because multiscale region-based inputs are exploited in DR-CNN and MDGCN MSAGE-CAL, which can improve the classification accuracy

TABLE V
ACCURACY COMPARISONS FOR THE PAVIA UNIVERSITY SCENE

| ID | DR-CNN[55] | RBF-SVM | JSDF[56] | $S^2$GCN[57] | $S^2$GAT [29] | MSAGE-CAL |
|---|---|---|---|---|---|---|
| 1 | 92.10 | 83.14 | 82.40 | 92.87 | 87.31 | **93.93** |
| 2 | 96.39 | 66.75 | 90.76 | 87.06 | 87.94 | **99.90** |
| 3 | 84.23 | 69.65 | 86.71 | 87.97 | 77.28 | **89.75** |
| 4 | 95.26 | 88.24 | 92.88 | 90.85 | **96.57** | 92.16 |
| 5 | 97.77 | 92.18 | **100.00** | **100.00** | 96.74 | 98.71 |
| 6 | 90.44 | 93.54 | 94.30 | 88.69 | **95.11** | 82.88 |
| 7 | 89.05 | 91.84 | 96.62 | 98.88 | 87.45 | **99.54** |
| 8 | 78.49 | 90.67 | 94.69 | 89.97 | **95.86** | 96.55 |
| 9 | 96.34 | 95.38 | **99.56** | 98.89 | 94.31 | 96.40 |
| OA | 92.62 | 77.65 | 90.82 | 89.74 | 90.56 | **96.14** |
| AA | 91.12 | 85.71 | 93.10 | 92.80 | 90.95 | **94.42** |
| Kappa | 0.90 | 0.77 | 0.88 | 0.87 | 0.90 | **0.97** |

Bold numbers indicate the best performance.

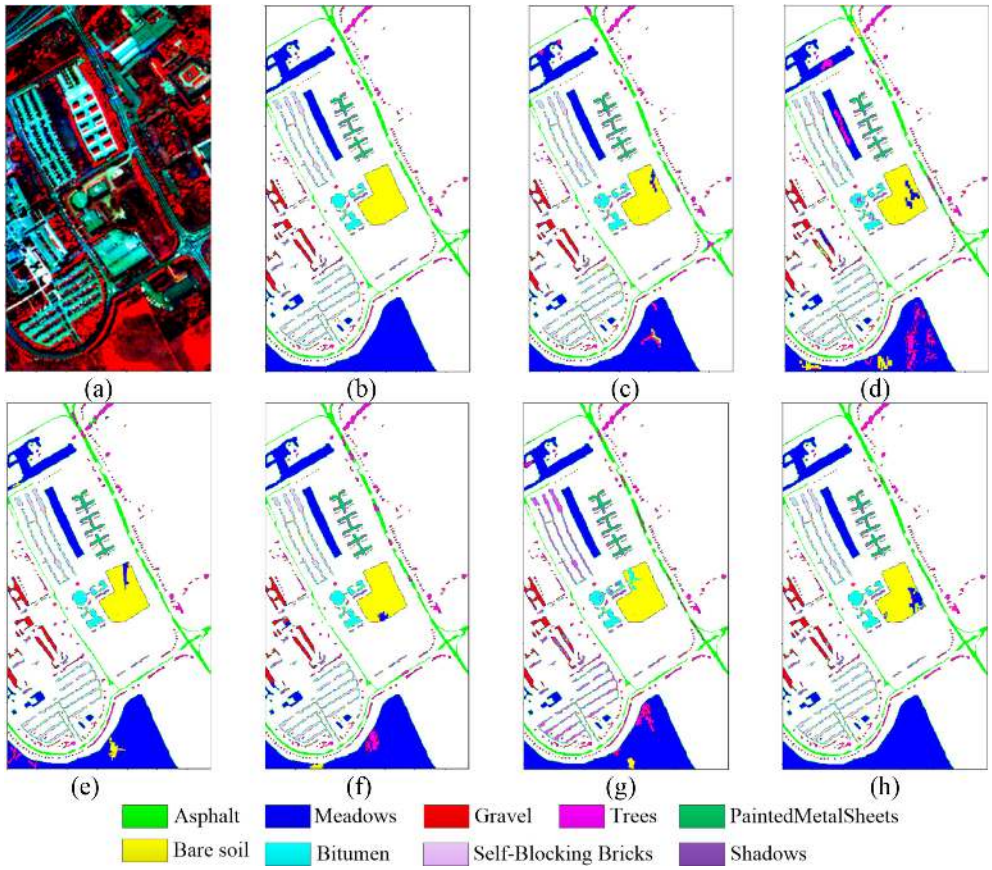


Fig. 6. Classification maps obtained by different methods on Pavia University dataset. (a) False color image. (b) Ground-truth map. (c) DR-CNN. (d) RBF-SVM. (e) JSDF. (f) Nonlocal GCN. (g) S2GCN. (h) MSAGE-CAL.

of the HSIs containing many boundary regions. Although the DR-CNN model has achieved a good result, its classification accuracy in C8 (self-blocking bricks) is significantly lower than MSAGE-CAL, which indicates that our proposed method has good adaptation for HSI classification details.

Fig. 6 shows a visual comparison of the classification results yielded by the mentioned methods above on the Pavia University dataset. As presented in Fig. 6, we can conclude that the proposed MSAGE-CAL method shows fewer misclassifications and get a smoother visual effect when compared with the ground-truth map. Meanwhile, due to the lack of a context-aware learning mechanism, the detailed results produced by compared methods contain many misclassifications.

*2) Results on KSC Dataset:* As given in Table VI, the experimental results of six methods on the KSC dataset have great improvement compared with the performances on the PU dataset. Because the KSC dataset contains less noise and higher spatial resolution than the PU dataset, which is more suitable for landscape classification, it is worth noting that our proposed method gets outperforming results than the compared

TABLE VI
ACCURACY COMPARISONS FOR THE KSC SCENE

| ID | DR-CNN | RBF-SVM | JSDF | $S^2$GCN | $S^2$GAT | MSAGE-CAL |
|---|---|---|---|---|---|---|
| 1 | 98.72 | 93.27 | **100** | 95.12 | 99.16 | 98.86 |
| 2 | 97.97 | 92.14 | 92.07 | 95.15 | 96.27 | **98.59** |
| 3 | 97.49 | 90.27 | 95.13 | 96.17 | 98.30 | **100.00** |
| 4 | 62.46 | 91.74 | 59.01 | 71.17 | 84.62 | **95.95** |
| 5 | 94.66 | 85.10 | 85.34 | **97.71** | 96.23 | 96.95 |
| 6 | **97.65** | 86.23 | 86.48 | 89.95 | 93.11 | 96.48 |
| 7 | **100.00** | 72.98 | 98.93 | 98.22 | 97.18 | **100.00** |
| 8 | 97.42 | 91.33 | 94.76 | 89.10 | 95.67 | **100.00** |
| 9 | 99.93 | 89.17 | **100.00** | 99.59 | 96.89 | 97.55 |
| 10 | 98.84 | 90.62 | **100.00** | 98.04 | **100.00** | 98.93 |
| 11 | **100.00** | 88.35 | **100.00** | 99.23 | **100.00** | **100.00** |
| 12 | **98.94** | 92.46 | 95.52 | 95.63 | 97.96 | 98.31 |
| 13 | **100** | 90.13 | **100** | **100** | **100.00** | **100.00** |
| OA | 97.21 | 88.46 | 97.21 | 95.44 | 96.31 | **98.98** |
| AA | 95.70 | 88.75 | 94.38 | 94.24 | 96.56 | **98.59** |
| Kappa | 0.97 | 0.86 | 0.95 | 0.95 | 0.97 | **0.99** |

Bold numbers indicate the best performance.

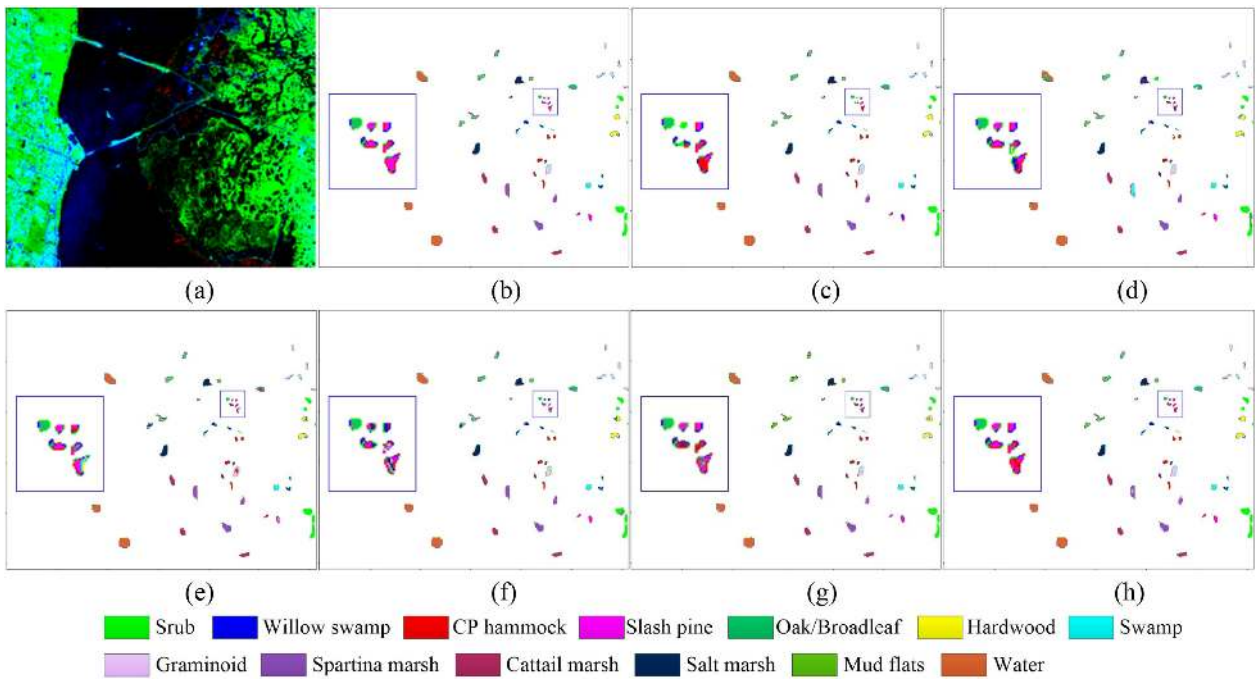


Fig. 7. Classification maps obtained by different methods on KSC dataset. (a) False color image. (b) Ground-truth map. (c) DR-CNN. (d) RBF-SVM. (e) JSDF. (f) $S^2$GCN. (g) MSAGE-CAL.

method and it also validates the performance of MSAGE-CAL. Besides MSAGE-CAL, RBF-SVM misclassifications that occur in the fourth class (Slash pine) are lower than the other method. However, the classification indices of RBF-SVM are the worst. This because RBF-SVM and MSAGE-CAL can extract local features effectively, which is important for local small objects classification. Furthermore, the performances of GCN-based methods present no advantage over other methods. This is due to the transductive learning mechanism adopted by GCN, which is unprofitable for isolated small object detection as KSC. Fig. 7 visualizes the classification results produced by all methods, where some critical regions are enlarged for better performance presentation. We notice that the proposed MSAGE-CAL gets better classification results on these small and difficult regions, which indicate that MSAGE-CAL is suited for small objects classification.

*3) Results on Salinas Dataset:* Table VII provides the quantitative results of different methods on the Salinas dataset. As demonstrated in the table, the classification results in C8 (Grapes untrained) and C15)Vineyard untrained(are lower than the other class because the two land cover types have similar spectral signatures with other classes. Besides, it is also notable that JSDF gets top-level performance among all the methods in terms of AA, which is different in PU and KSC. However, the

TABLE VII
ACCURACY COMPARISONS FOR THE SALIANS SCENE

| ID | DR-CNN | RBF-SVM | JSDF | $S^2$GCN | $S^2$GAT | MSAGE-CAL |
|---|---|---|---|---|---|---|
| 1 | 99.40 | 97.47 | **100.00** | 99.01 | 99.62 | **100.00** |
| 2 | 99.46 | 92.65 | **100.00** | 99.18 | 99.37 | 99.95 |
| 3 | 98.58 | 96.71 | **100.00** | 97.15 | 96.51 | 99.90 |
| 4 | **99.70** | 92.27 | 99.93 | 99.11 | 99.60 | 99.93 |
| 5 | 98.90 | 96.47 | **99.77** | 97.55 | 95.21 | 85.33 |
| 6 | 99.57 | 89.58 | **100.00** | 99.32 | 98.64 | 98.98 |
| 7 | 99.50 | 93.73 | 99.99 | 90.06 | 99.73 | **100.00** |
| 8 | 75.59 | 77.36 | 87.79 | 70.68 | 77.67 | **92.59** |
| 9 | 99.75 | 92.31 | 99.67 | 98.32 | 95.32 | **99.98** |
| 10 | 94.29 | 90.89 | 96.53 | 90.97 | 93.76 | **97.27** |
| 11 | 97.57 | 73.64 | **99.76** | 98.00 | 94.33 | 96.47 |
| 12 | 99.99 | 93.61 | **100.00** | 99.56 | 99.61 | 99.74 |
| 13 | 99.95 | 89.22 | **100.00** | 97.83 | 92.40 | 97.32 |
| 14 | 98.57 | 92.61 | **98.71** | 95.75 | 92.72 | 93.62 |
| 15 | 72.18 | 71.38 | 81.86 | 70.36 | 77.31 | **90.69** |
| 16 | 98.45 | 81.34 | **98.99** | 96.90 | 95.66 | 97.15 |
| OA | 90.35 | 86.75 | 94.67 | 88.39 | 93.67 | **96.87** |
| AA | 95.72 | 88.83 | **97.69** | 94.30 | 94.21 | 96.81 |
| Kappa | 0.89 | 0.86 | 0.94 | 0.87 | 0.93 | **0.97** |

Bold numbers indicate the best performance.

performance on OA and kappa are lower than MSAGE-CAL, which indicate that it is an imbalance among different classes classification. Furthermore, our proposed MSAGE-CAL has archive better performance than the GCN-based method. It is not only because of the shortcoming of the transductive learning mechanism, but also it shows that multiscale input futures have contributed to improving classification accuracy. As a visual comparison demonstrated in Fig. 8, we can observe that our proposed MSAGE-CAL yields smoother visual effectiveness than the other five competitors, which further shows the advantage of MSAGE-CAL. All results show that our proposed method has a good performance on objects of the different land cover types with a similar spectrum.

### *D. Analysis of the Performance of the MSAGE-CAL With Different Number of Labeled Samples*

In this experiment, the classification performances of the six algorithms with different numbers of labeled examples (i.e., pixels) for training are investigated. We vary the number of labeled examples per class from 5 to 30 with an interval of 5 and report the OA performance acquired by six algorithms on PU, KSC, and Salina datasets. The experimental results are demonstrated in Fig. 9. From the results, we can find that the performances on PU, KSC, and Salinas datasets are significantly improved with the increase of labeled examples; besides, the proposed MSAGE-CAL model performs better than the contrast algorithms from beginning to end, which shows the effectiveness of multiscale spatial information on HSI classification. Furthermore, the proposed MSAGE-CAL allows to automatically learn global contextual features and reconstruct the graph based on the classified land cover, which is more robust than using a precomputed fixed graph. It is also worth mentioning that the OA of the proposed MSAGE-CAL is stable with the numbers of labeled examples changing. All these results illustrate the stability and effectiveness of our proposed MSAGE-CAL.

TABLE VIII
OA, AA (%), AND KAPPA COEFFICIENT ACHIEVED BY DIFFERENT MODEL SETTINGS ON PAVIA UNIVERSITY DATASET

| Methods | SAGE-CAL | MSAGE | MSAGE-CAL |
|---|---|---|---|
| OA | 92.52 | 93.24 | 96.14 |
| AA | 90.16 | 93.67 | 94.42 |
| Kappa | 0.92 | 0.93 | 0.97 |

TABLE IX
OA, AA (%), AND KAPPA COEFFICIENT ACHIEVED BY DIFFERENT MODEL SETTINGS ON KSC DATASET

| Methods | SAGE-CAL | MSAGE | MSAGE-CAL |
|---|---|---|---|
| OA | 97.26 | 96.97 | 98.98 |
| AA | 97.34 | 97.18 | 98.59 |
| Kappa | 0.97 | 0.97 | 0.99 |

### *E. Ablation Study*

Our proposed MSAGE-CAL employed SAGE-based multiscale graph learning and context-aware learning mechanism to improve the illustrative ability of the model. In this experiment, we investigate the ablative effect of SAGE-based multiscale graph learning and context-aware learning. For the sake of comparison, we record the classification results produced without using multiscale graph learning information and context-aware learning mechanism, respectively, and the simplified model is denoted as "SAGE-CAL" and "and MSAGE." And the experimental setting is kept identical to Section IV-B. The comparative results are demonstrated in Tables VIII–X. As illustrated in

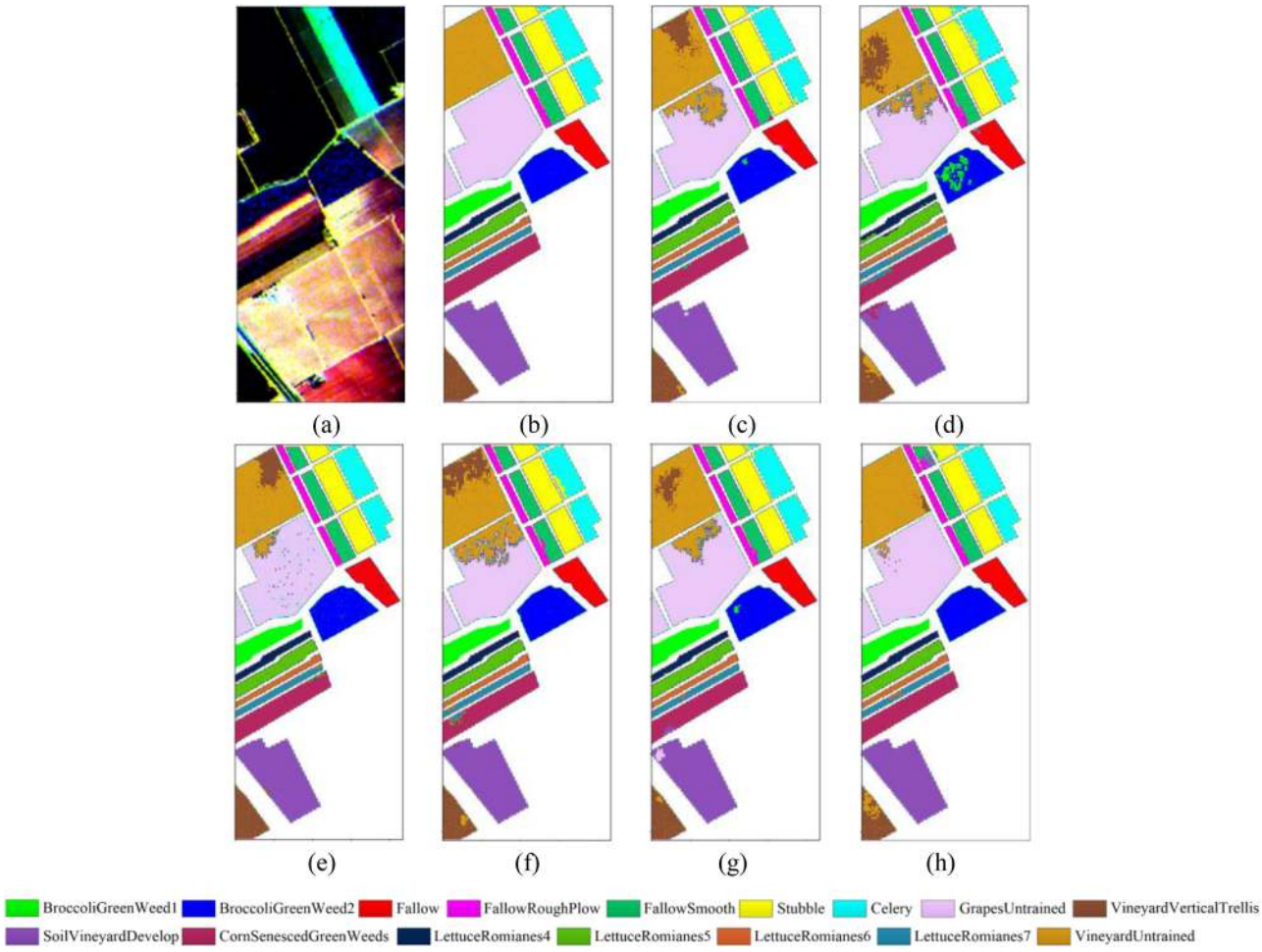


Fig. 8. Classification maps obtained by different methods on Salinans dataset. (a) False color image. (b) Ground-truth map. (c) DR-CNN. (d) RBF-SVM. (e) JSDF. (f) S2GCN. (g) Nonlocal GCN. (h) MSAGE-CAL.

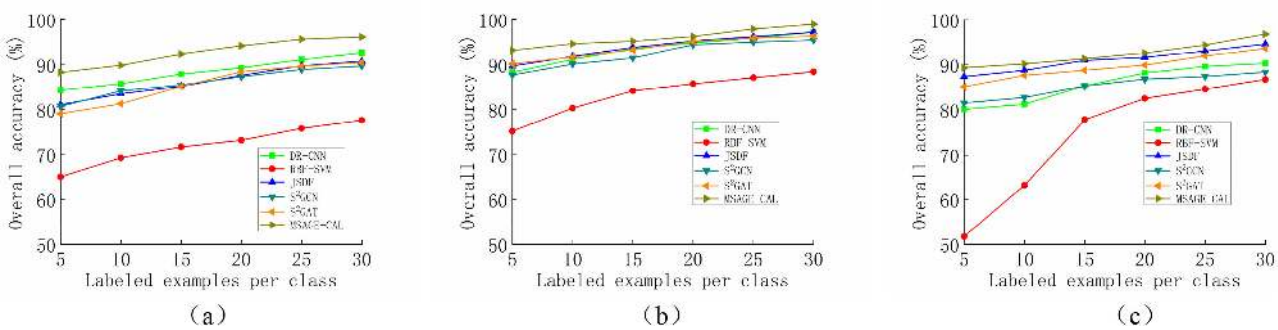


Fig. 9. Overall accuracies of various methods under different numbers of labeled examples per class. (a) PU dataset. (b) KSC dataset. (c) Salinas dataset.

tables, SAGE-based multiscale graph learning and context-aware learning play an important role in the improvement of learning efficiency.

### *F. Running Time*

Table XI demonstrates the running time of different deep methods, including DR-CNN, $S^2$GCN, $S^2$GAT, and our proposed MSAGE-CAL on three datasets (i.e., the PU, the KSC, and the Salinas), where the number of labeled pixels per class is kept identical to the experiments presented in Section VI-B. The results are reported on a server with a 3.70G Intel i9-10900K CPU and a GeForce GTX 1080Ti 11G GPU. From the results, we can get the conclusion that our proposed model is more efficient than the comparison methods, which is owing much to the employment of segmentation operation. The segmentation operation effectively reduces the number of graph nodes and reduces the number of model calculations.

TABLE X
OA, AA (%), AND KAPPA COEFFICIENT ACHIEVED BY DIFFERENT MODEL SETTINGS ON SALINAS DATASET

| Methods | SAGE-CAL | MSAGE | MSAGE-CAL |
|---|---|---|---|
| OA | 93.03 | 92.11 | 96.87 |
| AA | 96.30 | 94.17 | 96.81 |
| Kappa | 0.94 | 0.92 | 0.97 |

TABLE XI
RUNNING TIME COMPARISON (IN SECONDS) OF DIFFERENT METHODS

| Methods | DR-CNN | $S^2$GCN | $S^2$GAT | MSAGE-CAL |
|---|---|---|---|---|
| PU | 3245 | 2821 | 2712 | **1057** |
| KSC | 3376 | 1437 | 1537 | **876** |
| Salinas | 3121 | 3534 | 3462 | **447** |

"PU" denotes the University of Pavia dataset.

## V. CONCLUSION

In this article, we propose a novel multiscale graph sample and aggregate network with a context-aware learning method for HSI classification. The network adopts a multiscale graphSAGE convolution to extensively exploit the spatial information. And the context-aware learning mechanism is employed. Therefore, the network can extract global and contextual information from HSI, which helps to find accurate feature representations more effectively. Meanwhile, the graph structure is reconstructed based on backpropagation as network training, which can effectively reduce the influence of the initial graph error on the classification result. The experimental results on three real HSI datasets show that the proposed MSAGE-CAL is able to yield better performance when compared with various HSI classification methods.

## 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] D. Hong, N. Yokoya, J. Chanussot, and X. X. Zhu, "An augmented linear mixing model to address spectral variability for hyperspectral unmixing," *IEEE Trans. Image Process.*, vol. 28, no. 4, pp. 1923–1938, Apr. 2019.

[3] T. C. Bau, S. Sarkar, and G. Healey, "Hyperspectral region classification using a three-dimensional Gabor filterbank," *IEEE Trans. Geosci. Remote Sens.*, vol. 48, no. 9, pp. 3457–3464, Sep. 2010.

[4] D. Hong, N. Yokoya, N. Ge, J. Chanussot, and X. Zhu, "Learnable manifold alignment (LeMA): A semi-supervised cross-modality learning framework for land cover and land use classification," *ISPRS J. Photogram. Remote Sens.*, vol. 147, pp. 193–205, 2019.

[5] Y. Ding, X. Zhao, Z. Zhang, W. Cai, and N. Yang, "Graph sample and aggregate-attention network for hyperspectral image classification," *IEEE Geosci. Remote Sens. Lett.*, to be published, doi: 10.1109/LGRS.2021.3062944.

[6] Y. Yuan, D. Ma, and Q. Wang, "Hyperspectral anomaly detection via sparse dictionary learning method of capped norm," *IEEE Access,* vol. 7, pp. 16132–16144, 2019.

[7] D. Hong *et al.*, "More diverse means better: Multimodal deep learning meets remote-sensing imagery classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 5, pp. 4340–4354, May 2021.

[8] D. Hong, N. Yokoya, J. Chanussot, J. Xu, and X. Zhu. "Learning to propagate labels on graphs: An iterative multitask regression framework for semi-supervised hyperspectral dimensionality reduction," *ISPRS J. Photogram. Remote Sens.*, vol. 158, pp. 35–49, 2019.

[9] T. Ho, "The random subspace method for constructing decision forests," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 20, no. 8, pp. 832–844, Aug. 1998.

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

[11] D. Hong, N. Yokoya, J. Chanussot, and X. Zhu, "CoSpace: Common subspace learning from hyperspectral-multispectral correspondences," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 7, pp. 4349–4359, Jul. 2019.

[12] Z. Gong, P. Zhong, Y. Yu, W. Hu, and S. Li, "A CNN with multiscale convolution and diversified metric for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 6, pp. 3599–3618, Jun. 2019.

[13] R. Hang, Q. Liu, D. Hong, and P. Ghamisi, "Cascaded recurrent neural networks for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 8, pp. 5384–5394, Aug. 2019.

[14] H. Wu and S. Prasad, "Convolutional recurrent neural networks for hyperspectral data classification," *Remote Sens.*, vol. 9, no. 3, pp. 298–307, 2017.

[15] F. Zhou, R. Hang, Q. Liu, and X. Yuan "Hyperspectral image classification using spectral-spatial LSTMs." *Neurocomputing*, vol. 328, no. 7, pp. 39–47, 2019.

[16] X. Ma, G. Jie, and H. Wang, "Hyperspectral image classification via contextual deep learning," *EURASIP J. Image Video Process.*, vol. 2015, no. 1, Jul. 2015, Art. no. 20.

[17] B. Rasti *et al.*, "Feature extraction for hyperspectral imagery: The evolution from shallow to deep: Overview and toolbox," *IEEE Geosci. Remote Sens.,* vol. 8, no. 4, pp. 60–88, Dec. 2020.

[18] X. Lu, B. Wang, X. Zheng, and X. Li, "Exploring models and data for remote sensing image caption generation," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 4, pp. 2183–2195, Apr. 2018.

[19] 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, Jul. 2015.

[20] N. Kussul, M. Lavreniuk, S. Skakun, and A. Shelestov, "Deep learning classification of land cover and crop types using remote sensing data," *IEEE Geosci. Remote Sens. Lett.*, vol. 14, no. 5, pp. 778–782, May 2017.

[21] K. Makantasis, K. Karantzalos, A. Doulamis, and N. Doulamis, "Deep supervised learning for hyperspectral data classification through convolutional neural networks," in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, Jul. 2015, pp. 4959–4962.

[22] D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri, "Learning spatiotemporal features with 3D convolutional networks," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2015, pp. 4489–4497.

[23] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 1–9.

[24] W. Hamilton, Z. Ying, and J. Leskovec, "Inductive representation learning on large graphs," in *Proc. Conf. Adv. Neural Inf. Process. Syst.*, 2017, pp. 1024–1034.

[25] D. K. Duvenaud *et al.*, "Convolutional networks on graphs for learning molecular fingerprints," in *Proc. Conf. Adv. Neural Inf. Process. Syst.*, 2015, pp. 2224–2232.

[26] C. Wang, S. Pan, R. Hu, G. Long, J. Jiang, and C. Zhang, "Attributed graph clustering: A deep attentional embedding approach," in *Proc. Int. Joint Conf. Artif. Intell.*, 2019, pp. 3670–3676.

[27] A. Sha, B. Wang, X. Wu, and L. Zhang, "Semisupervised classification for hyperspectral images using graph attention networks," *IEEE Geosci. Remote Sens. Lett.*, vol. 18, no. 1, pp. 157–161, Jan. 2020.

[28] 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.

[29] 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*, to be published, doi: 10.1109/TGRS.2020.30151-57.

[30] Z.-Q. Zhao, P. Zheng, S.-T. Xu, and X. Wu, "Object detection with deep learning: A review," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 30, no. 11, pp. 3212–3232, Nov. 2019.

[31] 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.

[32] T. Li, J. Zhang, and Y. Zhang, "Classification of hyperspectral image based on deep belief networks," in *Proc. IEEE Int. Conf. Image Process.*, Oct. 2014, pp. 5132–5136.

[33] J. Feng *et al.*, "Generative adversarial networks based on collaborative learning and attention mechanism for hyperspectral image classification," *Remote Sens.*, vol. 12, no. 7, pp. 1149–1156, 2020.

[34] L. Zhu, Y. Chen, P. Ghamisi, and J. A. Benediktsson, "Generative adversarial networks for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 9, pp. 5046–5063, Sep. 2018.

[35] J. Yang, Y. Zhao, J. C. Chan, and C. Yi, "Hyperspectral image classification using two-channel deep convolutional neural network," in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, Jul. 2016, pp. 5079–5082.

[36] V. Slavkovikj, S. Verstockt, W. De Neve, S. Van Hoecke, and R. Van de Walle, "Hyperspectral image classification with convolutional neural networks," in *Proc. ACM Int. Conf. Multimedia*, Oct. 2015, pp. 1159–1162.

[37] D. Hong, N. Yokoya, G. Xia, J. Chanussot, and X. Zhu. "X-ModalNet: A semi-supervised deep cross-modal network for classification of remote sensing data," *ISPRS J. Photogram. Remote Sens.*, vol. 167, pp. 12–23, 2020.

[38] Y. Li, H. Zhang, and Q. Shen, "Spectral-spatial classification of hyperspectral imagery with 3D convolutional neural network," *Remote Sens.,* vol. 9, no. 1, p. 67, 2017.

[39] J. Feng *et al.*, "Attention multi-branch convolutional neural network for hyperspectral image classification based on adaptive region search," *IEEE Trans. Geosci. Remote Sens.*, to be published, doi: 10.1109/TGRS.2020.3011943.

[40] J. Bruna, W. Zaremba, A. Szlam, and Y. LeCun, "Spectral networks and locally connected networks on graphs," *arXiv:1312.6203*.

[41] Y. Zhong, X. Lin, and L. Zhang, "A support vector conditional random fields classifier with a Mahalanobis distance boundary constraint for high spatial resolution remote sensing imagery," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 7, no. 4, pp. 1314–1330, Apr. 2014.

[42] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," in *Proc. Int. Conf. Learning Represent.*, 2017.

[43] Z. Zhang, P. Cui, and W. Zhu, "Deep learning on graphs: A survey," *IEEE Trans. Knowl. Data Eng.*, to be published, doi: 10.1109/TKDE.2020.2981333.

[44] P. Velickovic, G. Cucurull, A. Casanova, A. Romero, P. Lio, and Y. Bengio, "Graph attention networks," in *Proc. Int. Conf. Learn. Represent.*, 2018.

[45] X. Qi, R. Liao, J. Jia, S. Fidler, and R. Urtasun, "3D graph neural networks for RGBD semantic segmentation," in *Proc. IEEE Int. Conf. Comput. Vis.*, Oct. 2017, pp. 5199–5208.

[46] H. Zhou, T. Young, M. Huang, H. Zhao, J. Xu, and X. Zhu, "Commonsense knowledge aware conversation generation with graph attention." in *Proc. Int. Joint Conf. Artif. Intell.*, 2018, pp. 4623–4629.

[47] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Ssstrunk, "Slic superpixels compared to state-of-the-art superpixel methods," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 11, pp. 2274–2282, Nov. 2012.

[48] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting,"*J. Mach. Learn. Res.*, vol. 15, no. 1, pp. 1929–1958, Jun. 2014.

[49] S. Zhang and S. Li, "Spectral-spatial classification of hyperspectralimages via multiscale superpixels based sparse representation," in *Proc. IEEE Int. Geosci. Remote Sensing Symp.*, Jul. 2016, pp. 2423–2426.

[50] T. Dozat, "Incorporating nesterov momentum into Adam." [Online]. Available: http://cs229.stanford.edu/proj2015/054_report.pdf

[51] 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.

[52] C. Bo, H. Lu, and D. Wang, "Hyperspectral image classification via JCR and SVM models with decision fusion," *IEEE Geosci. Remote Sens. Lett.*, vol. 13, no. 2, pp. 177–181, Feb. 2016.

[53] A. Qin, Z. Shang, J. Tian, Y. Wang, T. Zhang, and Y. Y. Tang, "Spectralspatial graph convolutional networks for semisupervised hyperspectral image classification," *IEEE Geosci. Remote Sens. Lett.*, vol. 16, no. 2, pp. 241–245, Feb. 2019.