Multiscale Superpixel-Guided Weighted Graph Convolutional Network for Polarimetric SAR Image Classification

Ru Wang, Yinju Nie, and Jie Geng, *Member, IEEE*

***Abstract*—Polarimetric synthetic aperture radar (PolSAR) has attracted more attentions because of its excellent observation ability, and PolSAR image classification has become one of the significant tasks in remote sensing interpretation. Various types and sizes of land cover objects lead to misclassification, especially in the boundaries of different categories. To solve these issues, a multiscale superpixel-guided weighted graph convolutional network (MSGWGCN) is proposed for classifying PolSAR images. In the proposed MSGWGCN, multiscale superpixel features are imported into the weighted graph convolutional network to obtain higher level representation, which can make full use of land cover object information in PolSAR images. Moreover, to fuse pixel-level features at different scales, a multiscale feature cascade fusion module is built, which plays an important role in preserving classification details. Experiments on three PolSAR datasets indicate that the proposed MSGWGCN performs better than other advanced methods on PolSAR classification task.**

***Index Terms*—Feature representation, graph convolutional networks, polarimetric synthetic aperture radar (PolSAR) image classification, superpixels.**

I. Introduction

POLARIMETRIC synthetic aperture radar (PolSAR) has become more and more popular in many fields, since it has an excellent ability to observe in all-time and all-weather operating modes. PolSAR images contain abundant scattering information, which can reflect physical characteristics of targets. Therefore, PolSAR images have been applied to ship detection [1], oil spill detection [2], target recognition [3], and so on [4], [5]. PolSAR image classification, a significant subject in SAR image interpretation, has shown its value in forest monitoring [6], urban planning [7], and other applications.

Manuscript received 3 September 2023; revised 14 November 2023 and 22 December 2023; accepted 9 January 2024. Date of publication 17 January 2024; date of current version 31 January 2024. This work was supported in part by the National Natural Science Foundation of China under Grant 62271396, in part by the Shaanxi Key Research and Development Program under Grant 2023-YBGY-220, in part by the China Postdoctoral Science Foundation under Grant 2021TQ0271 and Grant 2021M700110, in part by the Young Talent Fund of Xi'an Association for Science and Technology under Grant 959202313089, and in part by the Practice and Innovation Funds for Graduate Students of Northwestern Polytechnical University under Grant PF2023014. *(Corresponding author: Jie Geng.)*

The authors are with the School of Electronics and Information, Northwestern Polytechnical University, Xi'an 710129, China (e-mail: wangrunwpu@126.com; nieyinjunwpu@126.com; gengjie@nwpu.edu.cn).

Digital Object Identifier 10.1109/JSTARS.2024.3355290

Traditional methods for PolSAR image classification mainly obtain discriminative features based on target decomposition theory [8]. Krogager decomposition model separates the scattering matrix into three components, corresponding to three scattering mechanisms of helix, diplane, and sphere [9]. In Freeman decomposition [10], polarimetric covariance matrix is decomposed into three scattering components, double-bounce, surface, and canopy. On the basis of Freeman decomposition, Yamaguchi et al. [11] introduced the fourth scattering component, helix scattering power, which is more helpful to classify the PolSAR images. In addition, Cloude decomposition [12] is also common algorithm for analyzing PolSAR images. The statistical characteristics of PolSAR images also play an important role in the classification task, such as Wishart distribution [13] and K distribution [14]. Moreover, some traditional classifiers including support vector machine (SVM) [15] and decision tree [16] are designed for PolSAR classification. However, PolSAR targets with complex imaging mechanisms are difficult to be represented by these traditional features, which results in low classification accuracy.

Deep learning has proved its huge advantages in many fields, such as remote sensing [17], [18], medical, computer vision, and so on. Similarly, many researchers have made exploration on the application of deep neural networks in PolSAR image classification [19], [20]. Zhou et al. [21] extracted high-level features from the coherency matrix by using a deep neural network consisted of two convolution and two fully connected layers, which is suitable for PolSAR images. Chen et al. [22] effectively utilized the roll-invariant features of PolSAR targets and the hidden features of rotation domain to train the deep CNN model, which can further improve the classification performance. Wang et al. [23] imported the features belonging to different scattering mechanisms into different CNN models separately, and obtained the final classification features through a fusion network. Moreover, in order to fully extract the global information of sample patches, Dong et al. [24] introduced the ViT model into PolSAR classification task to achieve better performance. These methods are used to analyze PolSAR data in real domain, but the phase information is also valuable. Zhang et al. [25] used a complex-valued CNN model that can combine amplitude and phase information to reduce misclassification. Tan et al. [26] built a triplet complex-value network to learn feature representations of PolSAR data in the complex domain,

which can constrain the intraclass and interclass features among pixels. In addition, to overcome the problem of high cost of PolSAR image labeling, some unsupervised learning methods appeared on later. For instance, Ren et al. [27] took learning mutual information from PolSAR multimodal data as a pretext task, which can provide prior knowledge for classifier. Zhang et al. [28] utilized contrastive learning to obtain feature representations of PolSAR data and obtained good results in few-shot PolSAR classification.

Most of the existing methods import the sampling patches with fixed shapes into the networks. In this way, pixels belonging to different categories may be contained in a sampling patch, especially at the boundaries of objects, thus classification errors would be caused. As a result of oversegmentation, superpixels enable the contained pixels to keep homogeneous and reduce the influence of speckle noises, which can help to solve the problem caused by fixed sampling patches. At the same time, superpixels have advantages in retaining the boundary information of images due to the homogeneity of pixels within them [29], [30]. Several methods based on superpixel processing have achieved good performance in PolSAR image classification. An adaptive fuzzy superpixel algorithm used for PolSAR images is invented by Guo et al. [31], in which the correlation of pixel polarimetric information is considered. Gadhiya et al. [32] established a superpixel-driven optimized Wishart network and utilized the features of PolSAR image pixels and superpixels for classification. Morever, Geng et al. [33] combined semisupervised learning with the proposed superpixel restrained deep neural network to classify the PolSAR images with fewer labels.

Although superpixel based methods are proved to be useful in PolSAR image classification, the scale of superpixel segmentation would easily affect the classification results. Due to the various types and sizes of objects, it is difficult to cover the features of all objects with a single segmentation scale. Small-scale superpixels can protect local features of objects and have more detailed boundary information, while larger scale superpixels contain richer contextual information, which is conducive to strengthen the feature representation of the objects. Superpixels with different segmentation scales have different advantages in feature information, and they are able to provide appropriate segmentation scales for all objects in the PolSAR image. Therefore, it is beneficial to apply the multiscale superpixel features in classifying PolSAR images.

In addition, most of the existing methods take fixed sampling patches as input, and use CNN to extract features for classification. Only local information can be utilized in these methods. Polarimetric scattering information in the PolSAR image is complex, so it is difficult to correctly predict the categories of pixels by using only local information. While, the GCN-based methods take irregular superpixels as processing units, which is conducive to extract more accurate regional features. Furthermore, GCN updates features by aggregating the information of adjacent nodes according to the graph structure, and has the ability to learn more global information of images. Hence, GCN, which can combine the local information and global information, has advantages in classifying the PolSAR image.

Based on the above-mentioned issues, a multiscale superpixel-guided weighted graph convolutional network (MSGWGCN) is proposed for PolSAR image classification. First, superpixel segmentation is employed to segment a PolSAR image into superpixel sets of three different scales. Then, the adjacency matrices are generated according to the correlations between the averaged polarimetric features of superpixels, and the weighted graph convolutional network is utilized to obtain the multiscale superpixel feature representations. Finally, in order to get the final pixel-level features, a multiscale feature cascade fusion module (MFCFM) is developed to fuse the features at different segmentation scale, which has the ability to take advantage of multiscale information to get satisfactory classification details. The main contributions of our article are listed as follows.

1) The proposed model develops the weighted graph convolutional network to obtain more accurate feature representations of multiscale superpixels, which assigns learned weights to the adjacency matrix. It is capable to use multiscale information of the PolSAR image, so that it can be adapted well to land cover objects of different sizes, and help to achieve superior classification results on the boundary.
2) An MFCFM is built to obtain the final pixel-level feature representations, in which multiscale features are fused in the cascaded manner by a parallel multikernel CNN fusion operation. It can make a contribution to the classification details and improve the classification performance.
3) The superiority of MSGWGCN is validated on three PolSAR datasets. It is proved that multiscale superpixels as input can obtain better classification results than a single scale.

The rest of this article is organized as follows. In Section II, the related work is introduced briefly. Section III presents the proposed network MSGWGCN in details. Section IV reports the experimental results and analysis on three PolSAR datasets. Finally, Section V concludes this article.

II. Related Work

In this section, the detailed works related to graph convolutional network and PolSAR data processing are introduced.

A. Graph Convolutional Network

Graph convolutional network [34] has shown compelling advantages in graph data from many fields, such as graph classification [35], recommendation system [36], node classification [37], and so on. In addition, GCN has also been introduced into the PolSAR filed. Cheng et al. [38] used GCN to get the feature representations of PolSAR image superpixels, and fused the classification results of two different scales to achieve the optimized result. Liu et al. [39] proposed a method to classify the PolSAR images by using adaptive GCN, which can take into account both image data and spatial structures. In order to combine the features of neighboring nodes in different hops, Ren and Zhou [40] employed an evolving weighted graph

convolutional network to learn and extract the node features for classification.

Define an undirected graph structure as $G(V, E)$, in which V and E represent the node sets and edge sets, respectively. $\mathbf{A}$ is employed to stand for the adjacency matrix of G, and $\mathbf{A}(i, j)$ is equal to 1 when an edge exists between the ith and jth nodes. According to spectral graph theory [41], in the Fourier domain, the spectral convolution of signals $\mathbf{x}$ and filters $\mathbf{f}$ can be denoted as

$$\mathbf{f} \otimes \mathbf{x} = \mathbf{U}\mathbf{f}\mathbf{U}^{\top}\mathbf{x} \tag{1}$$

where $\otimes$ stands for the convolutional operation, $\top$ represents the transpose operation. $\mathbf{U}$ represents the eigenvector matrix obtained by decomposing the normalized Laplacian matrix $\mathbf{L} = \mathbf{I} - \mathbf{D}^{-\frac{1}{2}}\mathbf{A}\mathbf{D}^{-\frac{1}{2}}$, in which $\mathbf{D}$ denotes the node degree matrix, $\mathbf{D}(i, i) = \sum_j \mathbf{A}(i, j)$. Laplacian matrix eigenvalues are viewed as the components of the filters, so $\mathbf{f}$ is further expressed as $f(\mathbf{\Lambda})$. Since it takes a lot of computing resources to compute the eigendecomposition of Laplacian matrix $\mathbf{L}$, the R-order truncated expansion of Chebyshev polynomials is recommended to approximate $f(\mathbf{\Lambda})$ [42]. After that, the spectral convolution is expressed as follows:

$$\mathbf{f} \otimes \mathbf{x} = \sum_{r=0}^{R} \theta'_r T_r(\tilde{\mathbf{L}})\mathbf{x} \tag{2}$$

where $T_r(\tilde{\mathbf{L}})$ and θ'_r represent the r-order expression and corresponding coefficient of Chebyshev polynomials, respectively. $\tilde{\mathbf{L}} = (2/\lambda)\mathbf{L} - \mathbf{I}$ stands for the Laplacian matrix after normalization. Generally, the order R is limited to 1 and λ approximates to 2. In this case, the single parameter θ is further employed to replace the two free parameters θ'_0, θ'_1 in the expression, so that overfitting can be prevented by reducing the calculated parameters. Equation (2) can be written as

$$\mathbf{f} \otimes \mathbf{x} = \theta(\mathbf{I} + \mathbf{D}^{-\frac{1}{2}}\mathbf{A}\mathbf{D}^{-\frac{1}{2}})\mathbf{x}. \tag{3}$$

Finally, the propagation formula of graph convolutional network is shown as follows:

$$\mathbf{X}^{(l+1)} = \sigma(\tilde{\mathbf{D}}^{-\frac{1}{2}}\tilde{\mathbf{A}}\tilde{\mathbf{D}}^{-\frac{1}{2}}\mathbf{X}^{(l)}\mathbf{W}^{(l)}) \tag{4}$$

where $\tilde{\mathbf{A}}$ denotes the adjacency matrix that holds self-loop information, $\tilde{\mathbf{A}} = \mathbf{A} + \mathbf{I}$. $\tilde{\mathbf{D}}$ stands for the node degree matrix corresponding to $\tilde{\mathbf{A}}$, $\mathbf{W}^{(l)}$ represents the trainable filter matrix, $\mathbf{X}^{(l)}$ and $\mathbf{X}^{(l+1)}$ stand for the input and output features of the lth GCN layer, and $\sigma(\cdot)$ denotes the activation function.

B. PolSAR Data Preprocess

For a PolSAR image, the Sinclair scattering matrix $\mathbf{S}$ [43] is employed to reflect the polarimetric features of each image pixel, which can be denoted as follows:

$$\mathbf{S} = \begin{bmatrix} S_{hh} & S_{hv} \\ S_{vh} & S_{vv} \end{bmatrix} \tag{5}$$

where h and v stand for horizontal and vertical polarizations, respectively. $S_{ij}(i, j \in \{h, v\})$ denotes the polarimetric component generated by electromagnetic wave transmitting in the direction j and receiving in direction i.

In order to obtain a better polarimetric feature represents of targets, the Sinclair scattering matrix $\mathbf{S}$ is usually transformed on the Pauli basis, which is described as follows:

$$\begin{aligned}\mathbf{S} = &\frac{a_1}{\sqrt{2}}\begin{bmatrix}1 & 0\\ 0 & 1\end{bmatrix} + \frac{a_2}{\sqrt{2}}\begin{bmatrix}1 & 0\\ 0 & -1\end{bmatrix} \\ &+\frac{a_3}{\sqrt{2}}\begin{bmatrix}0 & 1\\ 1 & 0\end{bmatrix} + \frac{a_4}{\sqrt{2}}\begin{bmatrix}0 & -i\\ i & 0\end{bmatrix}\end{aligned} \tag{6}$$

where a_1, a_2, a_3, and a_4 denote the components occupied by each scattering mechanism in the Pauli basis.

Furthermore, the Pauli scattering vector denoted as $\mathbf{p} = [a_1, a_2, a_3, a_4]^{\top}$ can be obtained, and based on the reciprocity theorem [44], $S_{hv} = S_{vh}$. So the Pauli scattering vector can be written as

$$\mathbf{p} = \frac{1}{\sqrt{2}}\begin{bmatrix} S_{hh} + S_{vv} \\ S_{hh} - S_{vv} \\ 2S_{hv} \end{bmatrix}. \tag{7}$$

The coherence matrix $\mathbf{T}$ used in this work can be calculated from the Pauli scattering vector $\mathbf{p}$. The expression is shown as follows:

$$\mathbf{T} = \langle \mathbf{p} \cdot \mathbf{p}^{*\top} \rangle = \frac{1}{2}\begin{bmatrix} \langle |S_a|^2 \rangle & \langle S_a S_b^* \rangle & \langle S_a S_c^* \rangle \\ \langle S_a^* S_b \rangle & \langle |S_b|^2 \rangle & \langle S_b S_c^* \rangle \\ \langle S_a^* S_c \rangle & \langle S_b^* S_c \rangle & \langle |S_c|^2 \rangle \end{bmatrix} \tag{8}$$

where $S_a = S_{hh} + S_{vv}$, $S_b = S_{hh} - S_{vv}$, and $S_c = 2S_{hv}$, $*$ stands for the conjugate operation, $\top$ denotes the transpose operation, and $\langle \cdot \rangle$ means the set average in space or time. According to the method proposed in [45], the coherence matrix $\mathbf{T}$ can be transformed into a 6-D vector, which is taken as the input of our network and is also used as the feature basis of superpixel segmentation.

III. Methodology

In this section, the proposed MSGWGCN will be introduced in details, whose overall framework is displayed in Fig. 1. First, a PolSAR image is segmented into homogenous regions of three different scales by superpixel segmentation. Then, under the constraint of adjacency matrices, the weighted graph convolutional network is used to obtain the multiscale superpixel feature representations. Finally, MFCFM is utilized to obtain the final feature representations at pixel level for classification, which has effective performance in fusing feature maps of different segmentation scales. The pseudocode of the proposed method is provided in Algorithm 1.

A. Multiscale Superpixel Segmentation

PolSAR image usually consists of a large number of pixels, so it would cost much computing resources if pixels are treated as nodes in generating a graph. Superpixel segmentation has the ability to divide PolSAR data into homogeneous regions, which enables to construct a graph on superpixel-level nodes. Due to the complex structure and different sizes of land cover objects in PolSAR images, a single segmentation scale is not suitable

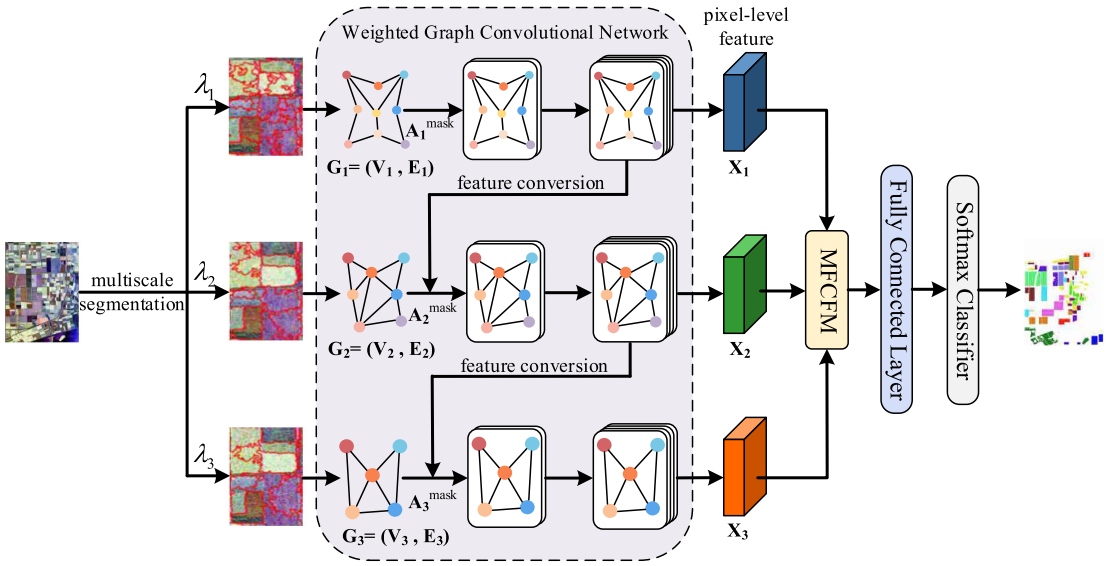

Fig. 1. Proposed MSGWGCN.

for all objects and cannot make full use of spatial features. Thus, multiscale superpixel segmentation is utilized to solve the problem.

In this work, the PolSAR image denoted as $\mathbf{Z} \in \mathbb{R}^{I_h \times I_w \times 6}$ is divided into a series of superpixels by simple linear iterative clustering method [46], where I_h and I_w stand for the length and width of the image, respectively. The parameter λ is used to control the scale of superpixel segmentation, which can make the PolSAR image become N superpixels denoted as $S = \{S_1, \ldots, S_N\}$, in which $N = (I_h \times I_w)/\lambda$. In the process of superpixel segmentation, a pixel-superpixel conversion matrix $\mathbf{C} \in \mathbb{R}^{(I_h I_w) \times N}$ can be obtained as follows:

$$C_{i,j} = \begin{cases} 1, & if\ p_i \in S_j \\ 0, & if\ p_i \notin S_j \end{cases} \tag{9}$$

where $C_{i,j}$ denotes the belonging relationship between the ith pixel p_i and the jth superpixel S_j of a PolSAR image. Three different scales of superpixel segmentation are used in the method. Therefore, three pixel–superpixel conversion matrices denoted as $\mathbf{C}_1 \in \mathbb{R}^{(I_h I_w) \times N_1}$, $\mathbf{C}_2 \in \mathbb{R}^{(I_h I_w) \times N_2}$ and $\mathbf{C}_3 \in \mathbb{R}^{(I_h I_w) \times N_3}$ can be obtained, in which N_1, N_2, and N_3 represent the numbers of superpixels at different segmentation scale.

After that, polarimetric features of all the pixels within a superpixel are averaged as the initial node feature in a graph, which is described as follows:

$$\bar{\mathbf{h}}_i = \frac{1}{M} \sum_{j=1}^{M} \mathbf{p}_j \tag{10}$$

where $\bar{\mathbf{h}}_i$ represents the ith superpixel feature, $\mathbf{p}_j$ stands for the feature of the jth pixel within the ith superpixel, M denotes the pixel number in the ith superpixel, $i = 1, \ldots, N$. Furthermore, after segmentation in three different scales, the superpixel-level node feature matrices at different segmentation scales $\mathbf{H}_1$, $\mathbf{H}_2$, and $\mathbf{H}_3$ can be obtained, in which $\mathbf{H}_i (i \in \{1, 2, 3\})$ is composed of superpixel features corresponding to different scales.

B. Weighted Graph Convolutional Network

Graph construction plays a vital role in the feature extraction of GCN. Different segmentation scales can result in different numbers of generated superpixels, so the corresponding graph structures denoted $G_i = (V_i, E_i), i \in \{1, 2, 3\}$ are also different, in which V_i and E_i represent node and edge sets, respectively. In the proposed MSGWGCN, the construction of the edges in a graph is determined by polarimetric feature correlation between superpixels. Since each GCN layer can update the node feature, a adjacency matrix with learnable weight is constructed for better feature representations, which can be expressed as follows:

$$\mathbf{A}^{\text{graph}} = \text{softmax}(\mathbf{H}\mathbf{W}^Q \times (\mathbf{H}\mathbf{W}^K)^\top) \cdot \mathbf{A}^{\text{mask}} \tag{11}$$

where $\mathbf{A}^{\text{graph}}$ represents the adjacency matrix, $\mathbf{H}$ denotes the node feature matrix, $\mathbf{W}^Q$ and $\mathbf{W}^K$ are trainable parameters, softmax$(\cdot)$ is the softmax function, $\cdot$ stands for the hadamard product operation, $\mathbf{A}^{\text{mask}}$ means the adjacency mask matrix, which is used to select the edges between nodes in each graph. The adjacency mask matrix $\mathbf{A}^{\text{mask}}$ is determined by the initial polarimetric feature correlation between superpixels, which is written as follows:

$$\mathbf{A}^{\text{mask}}(i, j) = \begin{cases} 1, & if\ \bar{\mathbf{h}}_i \in \text{KNN}(\bar{\mathbf{h}}_j) \\ 0, & \text{otherwise} \end{cases} \tag{12}$$

where $\mathbf{A}^{\text{mask}}(i, j)$ means the value on the location (i, j) of the adjacency mask matrix, $\bar{\mathbf{h}}_i$ and $\bar{\mathbf{h}}_j$ are employed to denote the polarimetric features of superpixels S_i and S_j, respectively. KNN$(\cdot)$ means the KNN algorithm [47], which is utilized to select K neighbors with the closest polarimetric features.

After that, the adjacency matrix and initial node features are imported into the graph convolutional network to extract higher level features, which can be described as follows:

$$\mathbf{H}_i^{(l+1)} = \sigma(\tilde{\mathbf{D}}_i^{-\frac{1}{2}} \tilde{\mathbf{A}}_i^{\text{graph}} \tilde{\mathbf{D}}_i^{-\frac{1}{2}} \mathbf{H}_i^{(l)} \mathbf{W}^{(l)}) \tag{13}$$

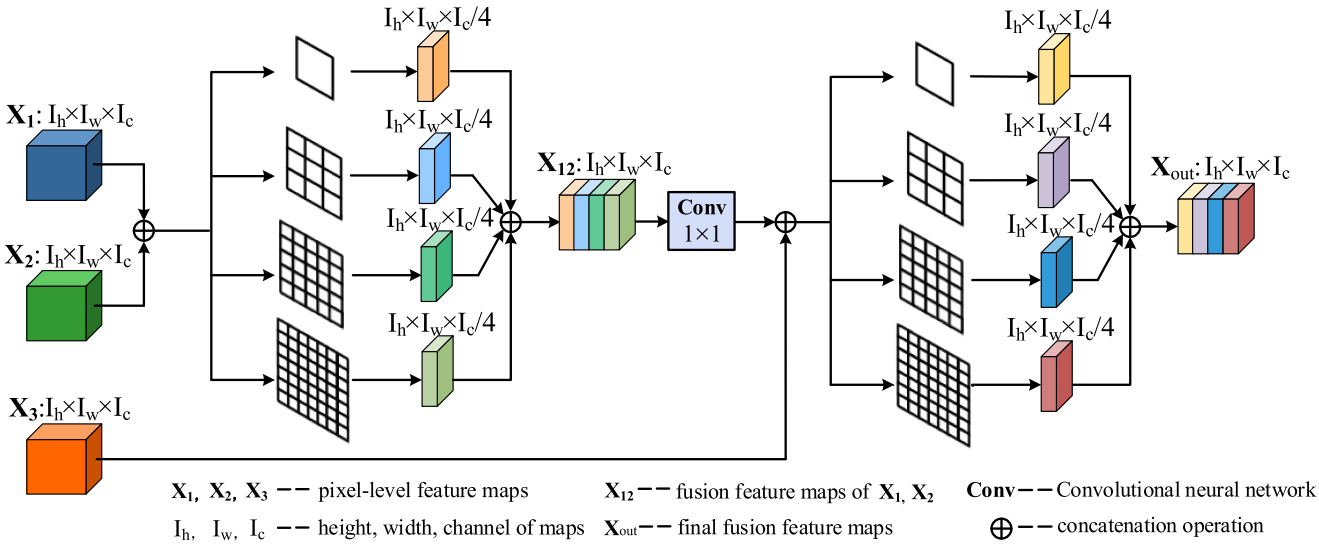

Fig. 2. Multiscale feature cascade fusion module (MFCFM).

where $\mathbf{H}_i^{(l)}$ and $\mathbf{H}_i^{(l+1)}$ stand for the input and output of the lth GCN layer, $\tilde{\mathbf{A}}_i^{\text{graph}} = \mathbf{A}_i^{\text{graph}} + \mathbf{I}$ represents the adjacency matrix $\mathbf{A}^{\text{graph}}$ with self-loop, the degree $\tilde{\mathbf{D}}_{\text{graph}} = \sum_j \tilde{\mathbf{A}}^{\text{graph}}$ is used to regularize $\tilde{\mathbf{A}}^{\text{graph}}$, $\mathbf{W}^{(l)}$ is the trainable parameters, $\sigma(\cdot)$ denotes the ReLU activation function, $i \in \{1, 2, 3\}$ means the segmentation scales.

For each scale of superpixels, two WGCN layers are utilized to extract higher level feature represents in the proposed method. It is worth noting that the input and output of these six GCN layers are connected. The initial input feature of GCN layers is small-scale superpixel embedding, and scale feature conversion operation is carried out through GCN layers of different scales, which can be obtained as follows:

$$\mathbf{H}_2^{(0)} = \overline{\mathbf{C}_2^\top \mathbf{C}_1} \mathbf{H}_1^{(2)} \tag{14}$$

$$\mathbf{H}_3^{(0)} = \overline{\mathbf{C}_3^\top \mathbf{C}_2} \mathbf{H}_2^{(2)} \tag{15}$$

where $\mathbf{H}_2^{(0)}$ and $\mathbf{H}_3^{(0)}$ denote the middle-scale and large-scale superpixel input features of weighted graph convolutional network, $\mathbf{H}_1^{(2)}$ and $\mathbf{H}_2^{(2)}$ denotes the small-scale and middle-scale output features of weighted graph convolutional network, $\overline{\mathbf{C}_2^\top \mathbf{C}_1} = (\mathbf{C}_2^\top \mathbf{C}_1)/\sum_i (\mathbf{C}_2^\top \mathbf{C}_1)$ and $\overline{\mathbf{C}_3^\top \mathbf{C}_2} = (\mathbf{C}_3^\top \mathbf{C}_2)/\sum_i (\mathbf{C}_3^\top \mathbf{C}_2)$ stand for the normalized small-middle scale conversion matrix and middle-large scale conversion matrix, respectively.

C. Multiscale Feature Cascade Fusion Module

After obtaining superpixel-level features of different segmentation scales, the MFCFM shown in Fig. 2 is employed to get the final features for classification. Let $\mathbf{H}_1^{(2)}$, $\mathbf{H}_2^{(2)}$, and $\mathbf{H}_3^{(2)}$ represent the output features of superpixels generated from three segmentation scales. By using the pixel–superpixel conversion matrices $\mathbf{C}_1$, $\mathbf{C}_2$, and $\mathbf{C}_3$, pixel-level features $\mathbf{X}_1 = \mathbf{C}_1 \mathbf{H}_1^{(2)}$, $\mathbf{X}_2 = \mathbf{C}_2 \mathbf{H}_2^{(2)}$, and $\mathbf{X}_3 = \mathbf{C}_3 \mathbf{H}_3^{(2)}$ can be obtained. The pixel-level features are fused step by step, where $\mathbf{X}_1$ and $\mathbf{X}_2$ are fused first, and then they are fused with $\mathbf{X}_3$.

Pixel-level features of different segmentation scales contain diverse scale information of land cover targets. To make good use of features at different scales and obtain better classification details, a parallel multikernel CNN fusion operation is utilized to fuse pairwise features. In this operation, convolutional kernels with different sizes are contained, where the receptive fields of convolutional kernels with different sizes are also different. Therefore, the parallel multikernel CNN fusion operation cannot only utilize the contextual information obtained by the large receptive field, but also focus on the local information obtained by the small receptive field, which makes MFCFM useful for optimizing the details of classification. The convolutional kernels are set to 1×1, 3×3, 5×5, and 7×7 in this module, and the output channel number of each convolutional kernel is reduced to 1/4 of the initial input channel. After that, output features are concatenated by channel dimension, which can be written as follows:

$$\mathbf{X}_{\text{fuse}} = \text{Concat}(\mathbf{M}_1, \mathbf{M}_2, \mathbf{M}_3, \mathbf{M}_4) \tag{16}$$

where $\mathbf{X}_{\text{fuse}}$ represents the fused features, $\mathbf{M}_1$, $\mathbf{M}_2$, $\mathbf{M}_3$, and $\mathbf{M}_4$ stand for the output feature maps through four different convolutional kernels, respectively.

The overall MFCFM can be expressed as

$$\mathbf{X}_{12} = f_p(\mathbf{X}_1 \oplus \mathbf{X}_2) \tag{17}$$

$$\mathbf{X}_{\text{out}} = f_p(\text{Conv}(\mathbf{X}_{12}) \oplus \mathbf{X}_3) \tag{18}$$

where $\mathbf{X}_{12}$ stands for the fusion feature of $\mathbf{X}_1$ and $\mathbf{X}_2$, $\mathbf{X}_{\text{out}}$ represents the final output fusion feature, $\oplus$ denotes the concatenation operation in channel dimension, $f_p(\cdot)$ stands for the parallel multikernel CNN fusion operation, and Conv means the 1×1 convolution.

Algorithm 1: Detailed Implementation of MSGWGCN.

Input: The whole PolSAR image $Z \in \mathbb{R}^{I_h \times I_w \times 6}$, number of neighbors K, segmentation scales $\{Scale1, Scale2, Scale3\}$.

Output: Category labels of the pxiels in the image.

1: Multiscale superpixel segmentation:
Get multiscale superpixel sets S^1, S^2, S^3;
Generate pixel-superpixel conversion matrices $\mathbf{C}_1$, $\mathbf{C}_2$ and $\mathbf{C}_3$ according to (9).
2: Initialize the multiscale superpixel feature representations according to (10);
Get initial superpixel-level feature matrices $\mathbf{H}_1$, $\mathbf{H}_2$ and $\mathbf{H}_3$.
3: Generate adjacency mask matrices according to (12);
Get adjacency mask matrices for the graph structures at different scales $\mathbf{A}_1^{mask}$, $\mathbf{A}_2^{mask}$, $\mathbf{A}_3^{mask}$.
4: Import $\mathbf{H}_i$, $\mathbf{A}_i^{mask}$, $\mathbf{C}_i$ into WGCN;
Get weighted adjacency matrices $\mathbf{A}_i^{graph}$ for each graph structure according to (11);
for i in range(3):
Function GCN ($\mathbf{A}_i^{graph}$, $\mathbf{H}$, σ)
$\mathbf{H}_i^{(l+1)} = \sigma(normal(\mathbf{A}_i^{graph})\mathbf{H}^{(l)}\mathbf{W}^l)$
Return superpixel-level features $\mathbf{H}_i^{l+1}$
End function
5: Generate pixel-level feature maps $\mathbf{X}_i = \mathbf{C}_i\mathbf{H}_i^{l+1}$.
6: Import multiscale superpixel feature maps $\mathbf{X}_i$ into MFCFM;
Fuse the multiscale pixel-level feature maps step by step.
7: $\mathbf{P} \leftarrow Softmax(\mathbf{X}_{out})$.
8: Category labels $\leftarrow$ argmax($\mathbf{P}$).

D. Loss Function

The cross entropy loss on the training samples is calculated to train the proposed MSGWGCN. A softmax classifier is employed to yield the predicted probability matrix of samples, which is described as follows:

$$\mathbf{P}_{ij} = \text{Softmax}(\mathbf{X}_{\text{out}}). \tag{19}$$

Then, the expression of the cross entropy loss L is illustrated as follows:

$$L = -\sum_{i=1}^{M}\sum_{j=1}^{C}\mathbf{Y}_{ij}\log(\mathbf{P}_{ij}) \tag{20}$$

where C stands for the number of classes, M means the training sample number, $\mathbf{Y}_{ij}$ and $\mathbf{P}_{ij}$ represent the ground-truth and the prediction probability of the samples, respectively.

IV. Experiments

In this section, the validity of the proposed network is verified by conducting experiments on three PolSAR datasets. First, details of three different PolSAR datasets are introduced. Then, classification results of MSGWGCN and some other comparative methods are analyzed.

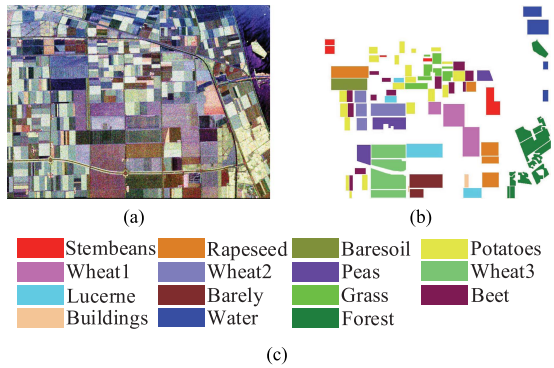

Fig. 3. Flevoland dataset. (a) Pauli RGB image. (b) Ground truth. (c) Color map of categories.

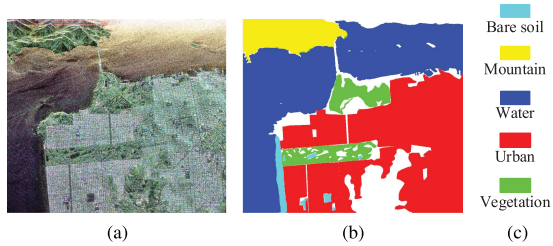

Fig. 4. San Francisco dataset. (a) Pauli RGB image. (b) Ground truth. (c) Color map of categories.

A. Datasets

1) Flevoland Dataset: The Flevoland dataset was produced in August 1989, and the data were acquired from the AIR-SAR sensor [48]. It is a L-band image with ground resolution of 6.6 m × 12.1 m. As shown in Fig. 3, there is 15 categories in this image of size 750 × 1024. In the experiments, 2% of labeled pixels in each category are selected as training samples randomly, 1%, 97% of the remaining labels are used for validation and testing.

2) San Francisco Dataset: San Francisco dataset reflects the land use of San Francisco area, and has a spatial resolution of 10 m. It was sensed by AIR-SAR in 1989. As displayed in Fig. 4, 900 × 1024 pixels and five categories are contained in this L-band image. In the experiments, 5% of labeled pixels per category are randomly selected [49], which are used for training the model. 1% and 94% of the remaining labels are used for validation and testing.

3) ESAR Oberpfaffenhofen Dataset: ESAR Oberpfaffenhofen dataset was produced by the ESAR airborne platform, which reflects the land use of Oberpfaffenhofen village. As displayed in Fig. 5, 1200 × 1300 pixels belonging to three categories are contained in this image [50]. In a similar way, the labeled pixels are randomly divided into 5%, 1%, and 94%, which are used for training, validation, and testing, respectively.

B. Experimental Details and Comparisons

The proposed model is implemented on the Pytorch framework. The network is trained for 400 epoches with the learning rate of 0.005, where Adam is utilized for optimization. In the proposed model, the output representation dimension of each

Fig. 5. ESAR Oberpfaffenhofen dataset. (a) Pauli RGB image. (b) Ground truth. (c) Color map of categories.

TABLE I
SETTING OF PARAMETER λ USED TO CONTROL SEGMENTATION SCALES

Parameter	Flevoland	San Franciso	ESAR Oberpfaffenhofen
λ_1	800	1000	2800
λ_2	1200	1400	3200
λ_3	1400	1800	3600

layer of weighted graph convolutional network is set to 64. More detailed parameter settings of MSGWGCN are reported in Tables II and III. The parameter setting of the WGCN module is given in Table II, which contains the network structure composition, the trainable parameters $\mathbf{W}^Q$, $\mathbf{W}^K$ used for weighted adjacency matrices under different scales, and the trainable parameters $\mathbf{W}^{(l)}$ of the GCN layers. Table III shows the parameter settings of the MFCFM module, which includes the specific network structures and parameters of the parallel multikernel CNN fusion under different fusion steps, and the parameters of the cascading structure. The parameter K used to select the edges of graphs corresponding to the three segmentation scales is equal to the same value 15. The settings of parameters λ_1, λ_2, and λ_3, which are employed to control different segmentation scales of different datasets are listed in Table I.

In order to quantify the performance of MSGWGCN and other methods for comparisons, the comparison experiments employ four metrics methods: per-class accuracy, overall accuracy (OA), average accuracy (AA), and Kappa coefficient. In addition, the visual results of the classification maps are also compared.

Several advanced methods are selected to compare with MSGWGCN model, including SVM [51], 2-D-CNN [35], 3-D-CNN [52], AFS-CNN [45], DSNet [53], GCN [38], CEGCN [54], and ViT [24]. The detailed experimental settings of these compared methods are introduced as follows.

1) SVM: The RBF kernel is utilized in SVM, and the parameter σ used to control the local scope of the RBF kernel is set to 0.001.
2) 2-D-CNN: This model has two 2-D-CNN layers and two fully connected layers, of which the convolutional kernel sizes are 2×2 and 3×3, respectively. The input patch size is 9×9.
3) 3-D-CNN: In this model, three 3-D-CNN layers with the kernel size of $2 \times 3 \times 3$ are employed to obtain classification features, and two fully connected layers are used for dimension transformation. Similarly, the size of input patch is 9×9.
4) AFS-CNN: This model contains a polarimetric feature attention mechanism, six 2-D-CNN layers with the kernel size of 3×3 and two fully connected layers. The size of input patch is as the same as 2-D-CNN.
5) DSNet: This model uses three depthwise separable convolution layers to extract features, in which the kernel sizes are 6×6 and 1×1 in the first layer, 3×3 and 1×1 in the second and last layer. Dense connection is used for strengthening the feature representations. The size of input patch is 15×15.
6) GCN: We build two layers of graph convolutional network to obtain the feature representation, in which the feature representation dimensions for each GCN layer are equal to 64 and 128, respectively.
7) CEGCN: In this model, two layers of graph convolutional network are built to obtain global features, four 2-D-CNN layers are employed to get local features, in which the kernel sizes is 1×1, 5×5, 1×1 and 5×5, respectively.
8) ViT: The input patch size is 15×15, and each pixel in the patch is treated as a token. The self-attention head number is set to 4, and the representation dimension of each head is set to 8. The depth of self-attention layer is equal to 8. The output dimension of MLP layer is 128.

C. Results on the Flevoland Dataset

In Table IV, the experiment results on the Flevoland dataset are reported, including the per-class accuracy, OA, AA, and Kappa coefficient. It can be seen that compared with the methods using deep neural networks, traditional machine learning method SVM obtains the lowest accuracy. Among the CNN-based methods, AFS-CNN performs better than 2-D-CNN, 3-D-CNN, and DSNet in classification results, because it has the ability to mine the useful polarimetric features for classification. By combining local and global features, CEGCN achieves higher classification accuracy than GCN. Comparing the accuracy of per class, it is clearly that the proposed method performs the best in all categories except for bare soil. Compared to the CNN-based methods, the proposed MSGWGCN can maintain a high classification accuracy for per category. The reason is that MSGWGCN is able to use the global information more effectively to obtain more accurate representation of the objects. Compared to the GCN-based methods, MSGWGCN has the best performance in "grass" with large sizes and "buildings" with small size, while GCN and CEGCN do not. It shows that the use of multiscale superpixel features is beneficial to PolSAR image classification.

In Fig. 6, classification maps obtained by different methods on the Flevoland dataset are depicted. Due to the low accuracy obtained by SVM, a large number of misclassified pixels exist in its visual map. Classification maps of 2-D-CNN, 3-D-CNN, AFS-CNN, DSNet, and ViT all have some scattered misclassification pixels, because these models use independent patch features to classify the central pixel. Among them, misclassification pixels of ViT mostly appear near the boundary of objects. However, there are clusters of misclassified pixels in the classification

TABLE II
PARAMETER SETTINGS OF WEIGHTED GRAPH CONVOLUTIONAL NETWORK

Graph structure	WGCN-layer1			WGCN-layer2		
	$\mathbf{W}_1^Q$	$\mathbf{W}_1^K$	$\mathbf{W}^{(l)}$	$\mathbf{W}_2^Q$	$\mathbf{W}_2^K$	$\mathbf{W}^{(l+1)}$
scale1	BatchNorm1d Linear(6,64) LeakyReLU	BatchNorm1d Linear(6,64) LeakyReLU	Linear(6,64) LeakyReLU	BatchNorm1d Linear(64,64) LeakyReLU	BatchNorm1d Linear(64,64) LeakyReLU	Linear(64,64) LeakyReLU
scale2	BatchNorm1d Linear(64,64) LeakyReLU	BatchNorm1d Linear(64,64) LeakyReLU	Linear(64,64) LeakyReLU	BatchNorm1d Linear(64,64) LeakyReLU	BatchNorm1d Linear(64,64) LeakyReLU	Linear(64,64) LeakyReLU
scale3	BatchNorm1d Linear(64,64) LeakyReLU	BatchNorm1d Linear(64,64) LeakyReLU	Linear(64,64) LeakyReLU	BatchNorm1d Linear(64,64) LeakyReLU	BatchNorm1d Linear(64,64) LeakyReLU	Linear(64,64) LeakyReLU

TABLE III
PARAMETER SETTINGS OF MFCFM

Composition	Parameter	Input size	Output size
Parallel multikernel CNN fusion operation	conv2d(1,1,16)+BatchNorm2d+ReLU	128	16
	conv2d(3,3,16)+BatchNorm2d+ReLU	128	16
	conv2d(5,5,16)+BatchNorm2d+ReLU	128	16
	conv2d(7,7,16)+BatchNorm2d+ReLU	128	16
Cascading structure	conv2d(1,1,64)+BatchNorm2d+ReLU	64	64
Parallel multikernel CNN fusion operation	conv2d(1,1,16)+BatchNorm2d+ReLU	128	16
	conv2d(3,3,16)+BatchNorm2d+ReLU	128	16
	conv2d(5,5,16)+BatchNorm2d+ReLU	128	16
	conv2d(7,7,16)+BatchNorm2d+ReLU	128	16

TABLE IV
EXPERIMENTAL RESULTS OBTAINED BY DIFFERENT METHODS ON FLEVOLAND DATASET

Class	**train**	**test**	**SVM** [51]	**2DCNN** [35]	**3DCNN** [52]	**AFS-CNN** [45]	**DSNet** [53]	**GCN** [38]	**CEGCN** [54]	**ViT** [24]	**MSGWGCN**
Steam beans	140	6762	89.30	96.88	95.24	98.49	96.47	97.25	96.98	98.43	**99.83**
Rapeseed	338	16366	73.99	94.78	87.23	96.89	92.53	96.60	96.52	96.60	**99.80**
Bare soil	126	6120	85.17	96.89	92.66	**99.42**	95.33	94.88	99.22	99.23	98.90
Potatoes	388	18821	89.85	98.20	95.18	98.54	98.06	94.38	98.53	98.48	**98.93**
Wheat1	339	16411	77.24	90.06	91.90	97.87	95.70	94.43	97.39	97.04	**99.87**
Wheat2	251	12144	64.60	88.47	78.53	93.44	87.82	95.55	96.94	96.91	**99.94**
Peas	226	10915	84.12	94.28	95.32	98.48	96.25	92.78	97.21	98.43	**99.98**
Wheat3	486	23522	88.28	95.35	93.88	99.13	95.55	96.39	98.01	99.50	**99.92**
Lucerne	238	11544	86.11	94.84	92.06	98.15	96.05	96.86	95.57	98.38	**99.54**
Barley	169	8149	92.99	96.97	95.60	99.24	95.20	96.42	99.48	99.47	**99.56**
Grasses	166	8037	52.81	85.07	77.59	95.84	90.74	89.92	96.74	98.34	**99.95**
Beets	233	11281	79.13	95.85	92.20	97.80	95.75	94.39	96.26	97.75	**99.42**
Buildings	19	902	68.00	82.81	72.65	93.02	90.81	89.33	93.47	91.54	**98.80**
Water	179	8670	76.00	92.03	85.83	99.91	96.17	89.37	99.74	99.84	**100.00**
Forest	422	20416	86.29	95.10	94.81	98.19	94.59	98.73	99.19	96.48	**99.91**
OA			81.22	94.06	91.03	97.93	94.81	95.24	97.69	98.02	**99.69**
AA			79.59	93.17	89.37	97.63	94.47	94.49	97.42	97.76	**99.62**
Kappa			79.51	93.52	90.22	97.74	94.34	94.81	97.48	97.84	**99.67**

maps of GCN, this is due to the fact that the basic processing unit of GCN is superpixel. As displayed in Fig. 6(i), MSGWGCN performs the best visual result. Compared to the classification maps obtained by SVM, 2-D-CNN, 3-D-CNN, AFSCNN, DSNet, and ViT, the proposed MSGWGCN has few scattered misclassification pixels and more accurate classification on the pixels at the boundary, which attributes to multiscale superpixels in MSGWGCN. Due to various scales of objects in the Flevoland dataset, single segmentation used in GCN and CEGCN is difficult to adapt to all objects, so that the classification maps are not satisfactory. In conclusion, the proposed MSGWGCN can effectively combine multiscale superpixel features to obtain better boundary visual classification performance.

D. Results on the San Francisco Dataset

Table V shows the classification results obtained by different models on the San Francisco dataset. Unlike the Flevoland dataset, this dataset contains fewer target categories. SVM obtains the relatively poor classification results, which indicates that original polarimetric features do not have good discrimination. It can be observed that the models 2-D-CNN,

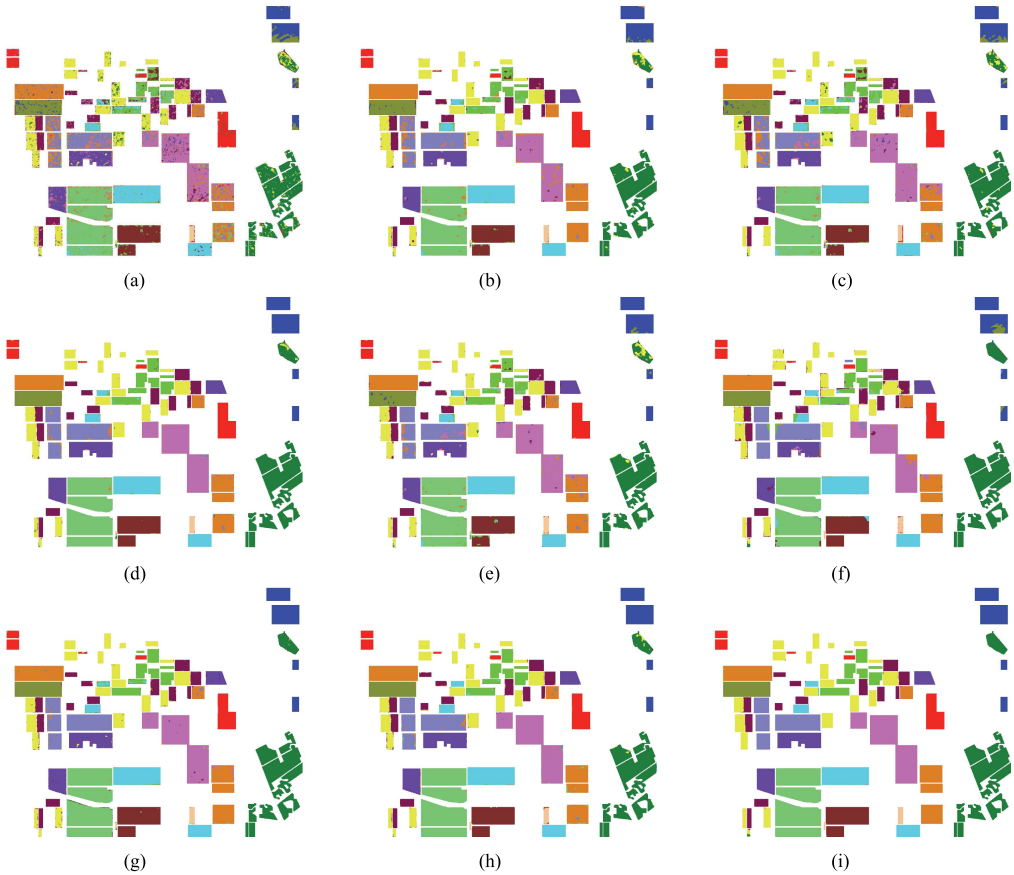

Fig. 6. Classification maps obtained by different methods on the Flevoland dataset. (a) SVM. (b) 2-D-CNN. (c) 3-D-CNN. (d) AFSCNN. (e) DSNet. (f) GCN. (g) CEGCN. (h) ViT. (i) MSGWGCN.

TABLE V
EXPERIMENT RESULTS OBTIANED BY DIFFERENT METHODS ON SAN FRANCISCO DATASET

Class	train	test	SVM [51]	2DCNN [35]	3DCNN [52]	AFS-CNN [45]	DSNet [53]	GCN [38]	CEGCN [54]	ViT [24]	MSGWGCN
Bare soil	686	12869	3.29	35.87	63.26	**91.41**	77.18	24.32	84.25	85.41	91.37
Mountain	3137	58927	55.80	87.23	89.20	97.25	93.00	87.13	98.80	96.80	**99.30**
Water	16479	309773	98.19	98.88	98.67	99.60	98.92	97.87	99.14	99.50	**99.67**
urban	17140	322257	93.19	97.12	97.40	99.11	97.46	97.86	99.36	97.87	**99.78**
Vegetation	2676	50335	37.30	78.58	72.86	73.83	80.07	73.83	94.47	90.88	**98.16**
OA			87.06	94.78	95.06	98.06	96.21	94.17	98.64	97.78	**99.44**
AA			57.55	79.54	84.28	92.24	89.33	76.20	95.20	94.09	**97.66**
Kappa			79.15	91.76	92.21	96.96	94.04	90.75	97.87	96.53	**99.12**

3-D-CNN, GCN do not perform well on this dataset, especially in the category of bare soil. In contrast, the methods AFS-CNN, DSNet, CEGCN, and ViT are more suitable for the complex polarimetric features and yield superior performance in bare soil. MSGWGCN not only achieves the highest OA of 99.44%, AA of 97.66%, and Kappa coefficient of 99.12%, but also exceeds other compared methods. The low classification accuracy in the categories "bare soil" and 'vegetation" are obtained by 2-D-CNN, 3-D-CNN, and ViT, but this issue does not happen in the classification results obtained by MSGWGCN, which is due to the better feature representation ability of MSGWGCN. As for the "bare soil" with smaller size in the San Francisco dataset, GCN and CEGCN get lower classification accuracy, but MSGWGCN performs well. It proves that the proposed method contributes to the identification of objects with small size. Therefore, it can be illustrated that MSGWGCN has great advantages in PolSAR classification on the San Francisco dataset.

Fig. 7 shows the classification visualizations obtained by different models on the San Francisco dataset. It is obvious that the visual effect of MSGWGCN is superior to other comparison methods. Compared with SVM, 2-D-CNN, 3-D-CNN, AFS-CNN, DSNet, and ViT, the visual result of the proposed MSGWGCN does not contain apparent misclassification pixels. Although classification maps of GCN-based methods are acceptable, they still have some shortcomings in the boundary of objects and the preservation of integrity. In the visual maps of MSGWGCN shown in the Fig. 7(i), it can be observed that all the categories have the optimal visual performance. "Urban" occupies the largest proportion in this PolSAR data, in which there are scattered or clustered misclassification pixels in this

Fig. 7. Classification maps obtained by different methods on San Francisco dataset. (a) SVM. (b) 2-D-CNN. (c) 3-D-CNN. (d) AFSCNN. (e) DSNet. (f) GCN. (g) CEGCN. (h) ViT. (i) MSGWGCN.

TABLE VI
EXPERIMENT RESULTS OBTAINED BY DIFFERENT METHODS ON ESAR OBERPFAFFENHOFEN DATASET

Class	train	test	SVM [51]	2DCNN [35]	3DCNN [52]	AFS-CNN [45]	DSNet [53]	GCN [38]	CEGCN [54]	ViT [24]	MSGWGCN
Bulit-up areas	15 892	298 700	76.18	93.32	91.99	95.18	95.18	88.08	94.91	91.61	**98.40**
Open areas	37 382	702 849	96.71	97.35	97.85	98.67	98.61	96.35	98.30	96.66	**99.47**
Wood land	13 581	255 304	91.53	89.87	96.02	97.70	95.18	94.99	96.60	94.23	**98.84**
OA			90.78	94.87	96.08	97.65	97.10	94.11	97.15	94.97	**99.09**
AA			88.14	93.51	95.29	97.19	96.33	93.14	96.60	94.17	**98.91**
Kappa			84.22	91.31	93.36	96.00	95.08	89.99	95.16	91.48	**98.46**

category. The compared methods have shortcomings in maintaining the integrity of the objects, but the proposed MSGWGCN performs much better. Therefore, the proposed MSGWGCN has advantages in the classification of objects with large scales. The visual results show that MSGWGCN is suitable for preserving the classification details and integrity.

E. Results on the ESAR Oberpfaffenhofen Dataset

Experimental results obtained by different methods on the ESAR Oberpfaffenhofen dataset are shown in Table VI. Specially, the highest accuracy is achieved by MSGWGCN. Compared with the advanced CNN-based model AFS-CNN and GCN-based model CEGCN, the proposed MSGWGCN increases the OA by 1.44% and 1.94%, the AA by 1.72% and 2.31%, and the Kappa by 2.46% and 3.3%, respectively. Besides, the highest accuracy in each category is also achieved by the proposed method. The ESAR Oberpfaffenhofen dataset has a large size, in which polarimetric scattering features of each category are complex. Therefore, classification of the pixels in some areas tends to be more difficult, such as the category "built-up areas." Compared to other methods, the proposed MSGWGCN achieves superior results in the "built-up areas." In conclusion, the proposed MSGWGCN has excellent feature representation ability to enhance the performance of PolSAR image classification.

Visual classification results obtained by different methods on the ESAR Oberpfaffenhofen dataset are depicted in Fig. 8. Like previous two datasets, there are some scattered misclassified pixels in the visual maps obtained by the models with

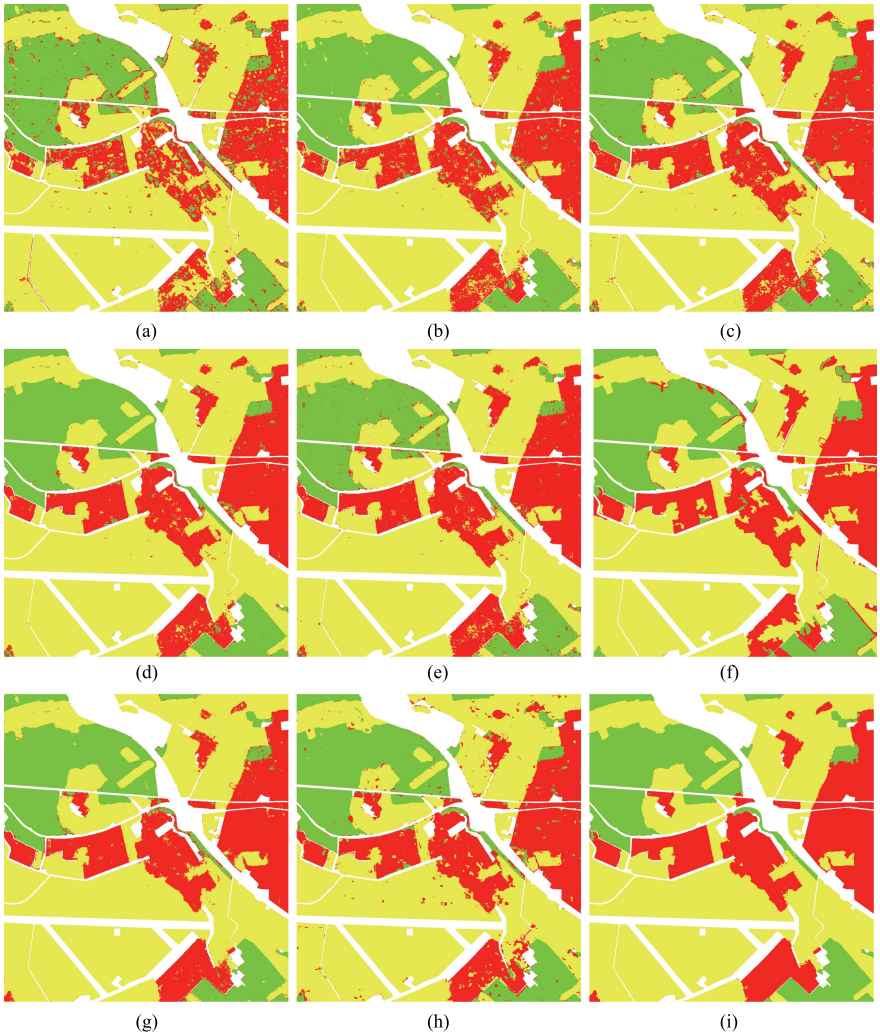

Fig. 8. Classification results obtained by different methods on the ESAR Oberpfaffenhofen dataset. (a) SVM. (b) 2-D-CNN. (c) 3-D-CNN. (d) AFSCNN. (e) DSNet. (f) GCN. (g) CEGCN. (h) ViT. (i) MSGWGCN.

sample patches as input. The basic process unit of the proposed MSGWGCN is superpixels, so that the scattered misclassified pixels rarely appear in the visual maps of MSGWGCN. The object boundary of the ESAR Oberpfaffenhofen dataset is more variable. It can be observed that GCN and CEGCN are prone to misclassify at the boundary of land cover objects. For example, pixels at the edge of "wood land" are misclassified into "built-up areas." In contrast, this misclassification does not appear in the visual results obtained by MSGWGCN, because of its superior superpixel feature representation capability. In summary, the classification effect provided by MSGWGCN is highly competitive with other methods.

F. Ablation Experiments

In order to demonstrate the contributions of the proposed modules to the final classification, a series of ablation experiments are conducted on the Flevoland, San Francisco, and ESAR Oberpfaffenhofen datasets. OA is utilized to evaluate the classification performance of the models. Classification results of the ablation experiments are shown in the Table VII. In the table, "without multiscale" means that multiscale inputs are not used in the MSGWGCN, and only single-scale superpixels are used as inputs. "without WGCN" denotes that the WGCN is not utilized in the MSGWGCN, but is replaced with a regular GCN layer, which is used to analyze the influence of the WGCN module. "Without MFCFM" represents that the MFCFM module is removed from MSGWGCN, so that the validity of MFCFM can be verified.

TABLE VII
ABLATION EXPERIMENTS ON THREE POLSAR DATASETS

Dataset	Flevoland	San Francisco	ESAR Oberpfaffenhofen
without Multiscale	98.40	97.73	98.20
without WGCN	99.45	99.14	98.98
without MFCFM	98.80	98.34	97.88
MSGWGCN	**99.69**	**99.44**	**99.09**

From Table VII, it can be observed that multiscale, WGCN and MFCFM all have positive impacts on the classification performance. For the Flevoland and San Francisco datasets, multiscale input makes a greater contribution to the classification.

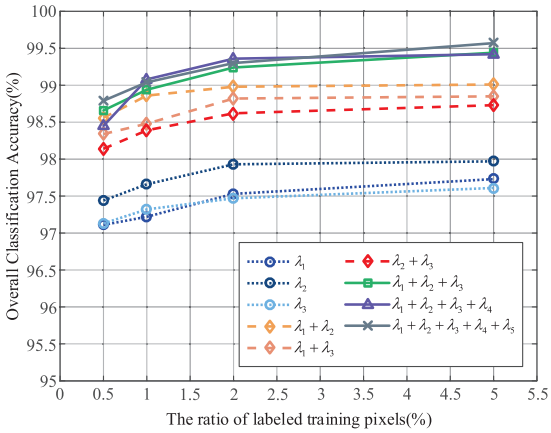

Fig. 9. OA with different numbers of segmentation scales on the San Francisco dataset.

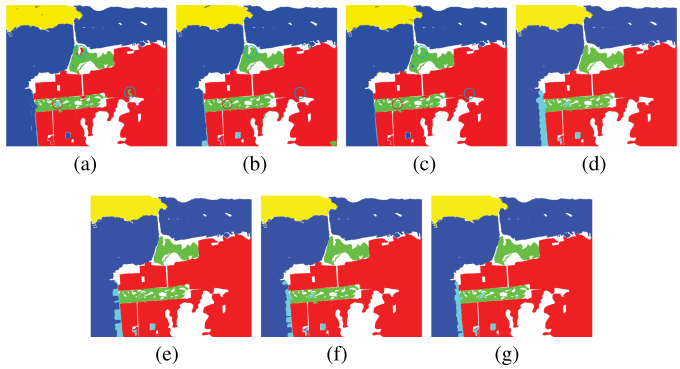

Fig. 10. Classification visual maps with different numbers of segmentation scale on the San Francisco dataset. (a) λ_1. (b) λ_2. (c) λ_3. (d) $\lambda_1+\lambda_2$. (e) $\lambda_2+\lambda_3$. (f) $\lambda_1+\lambda_3$. (g) $\lambda_1+\lambda_2+\lambda_3$.

TABLE VIII
OA WITH DIFFERENT SIZES OF PARALLEL CONVOLUTIONAL KERNEL IN MFCFM

Dataset	$\oplus$	k_1=1	k_1=1 k_2=3	k_1=1 k_2=3 k_3=5	k_1=1 k_2=3 k_3=5 k_4=7
Flevoland	98.80	99.07	99.41	99.54	99.69
San Francisco	98.34	98.77	99.12	99.29	99.44
ESAR Oberpfaffenhofen	97.82	98.11	98.61	98.93	99.09

For the ESAR Oberpfaffenhofen dataset, MFCFM plays a more important role in the classification.

G. Discussions

1) Effect of Multiscale Superpixels: To further verify the effectiveness of multiscale superpixels, the effect of different segmentation scale numbers on the San Francisco dataset are analyzed, and the performance on using different training sample numbers (2%, 1%, 0.5%, 0.1%) are compared in the experiment. Note that the final classification results of single segmentation scale λ_1, λ_2, λ_3 are also optimized by using the proposed MFCFM to ensure fairness. OA of classification performance with different numbers of segmentation scales is displayed in Fig. 9. Obviously, OA increases as training samples increases in all cases. In addition, it is clear that compared with a single segmentation scale, multiscale superpixels can achieve higher OA with different training samples. The compared results prove that multiscale superpixels have strength in using spatial information of PolSAR image. As the number of the used segmentation scales increased, the OA on the San Francisco dataset increases. However, it can be observed that OA increases slowly when the number of used scales reaches 3. Therefore, in the previous comparison experiments, the number of the segmentation scales is set to 3.

The visual results with different numbers of segmentation scale on San Francisco dataset are shown in Fig. 10. It is found that there are more misclassified pixels in the classification maps obtained by using superpixels of single scale. Especially, from Fig. 10(a)–(c), it is observed that some regions in an intact object are always misclassified with a single segmentation of small size λ_1, while this issue rarely occurs in the classification maps of larger scale superpixels λ_2 and λ_3. For example, the regions marked in blue circles. However, small scale superpixels have advantages in classification of boundary regions similar to red circle marks. In contrast, as shown in the Fig. 10(d)–(g), classification maps of multiscale superpixels have preferable classification performance in keeping the integrity and boundaries of objects, where using superpixels of three scales $\lambda_1+\lambda_2+\lambda_3$ contributes the best classification map. Therefore, effective combination of spatial features guided by different scale superpixels is beneficial to PolSAR classification.

2) Effect of MFCFM: To yield the final classification results, MFCFM is proposed to fuse the classification maps of three scales, in which a parallel multikernel CNN fusion operation is designed for optimization. Several experiments are conducted on three datasets to verify the validity of MFCFM and explore the effect of parallel convolutional kernel on classification accuracy. The experimental results are listed in Table VIII. $\oplus$ stands for the direct concatenation of features at different scales. It can be observed that OA obtained by using parallel multikernel CNN is higher than that of direct concatenation on three datasets, which indicates that the proposed MFCFM has advantages in the fusion of features at different scales. Besides, OA of three datasets increases with the size of parallel convolutional kernels, which means that convolutional kernels of different sizes enable to utilize multiscale contextual information to improve the PolSAR image classification.

3) Effect of Neighbor Number K: In MSGWGCN, the KNN algorithm is employed to select K nodes to construct the graphs of different scales. Therefore, a series of experiments on the San Francisco dataset are carried out to analyze K is analyzed. Fig. 11 depicts the classification accuracy with different numbers of the nearest neighbors, where K ranges from 1 to 50. Obviously, OA reaches the highest when K is equal to 15. Finally, the value of the neighbor number K in MSGWGCN is set to 15.

4) Effect of Training Sample Ratios: To analyze the classification performance of the proposed MSGWGCN with different training ratios, a series of experiments are conducted on the Flevoland, San Francisco, and ESAR Oberpfaffenhofen

TABLE IX
EXPERIMENTAL RESULTS OF THE GENERALIZATION PERFORMANCE OBTAINED BY DIFFERENT METHODS

Methods	Fle → San	Fle → ESAR	San → Fle	San → ESAR	ESAR → Fle	ESAR → San
AFS-CNN	83.79	86.65	88.55	83.64	94.56	86.12
CEGCN	86.87	79.94	90.45	85.58	88.42	88.25
ViT	88.74	82.62	91.67	86.62	88.53	74.87
MSGWGCN	**90.29**	**92.01**	**95.62**	**86.77**	**95.97**	**88.90**

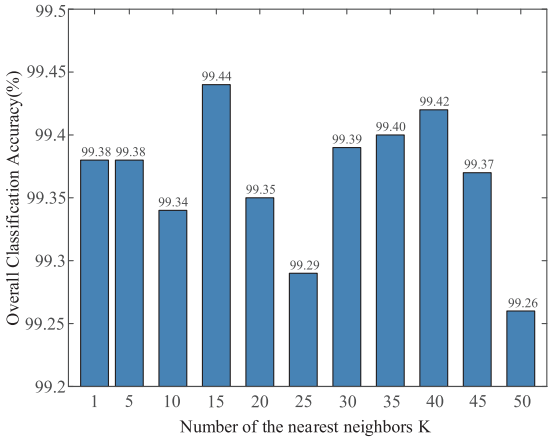

Fig. 11. OA with different numbers of the nearest neighbor K on the San Francisco dataset.

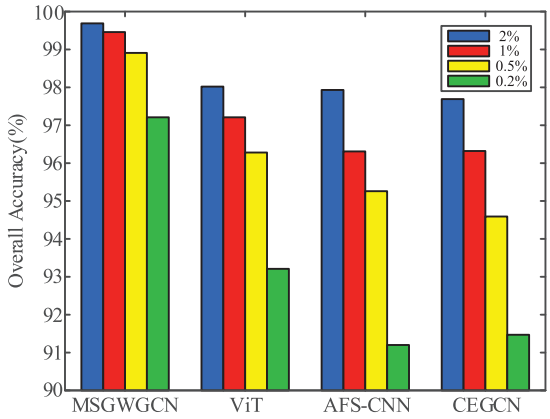

Fig. 13. Classification results obtained by different methods under different training sample ratios on the Flevoland dataset.

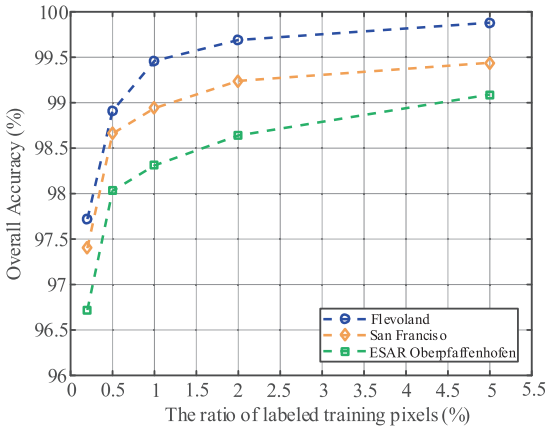

Fig. 12. Classification results by using different ratios of labeled training pixels.

datasets. Experimental results are depicted in the Fig. 12. From the figure, it can be seen that classification accuracy of the three datasets decreases as the ratio of training samples decreases. When the ratio is 0.2%, the OA obtained on the three datasets are the lowest, but still remain above 96%. When the ratio of training samples is higher than 0.5%, the classification accuracy decreases slowly with the decrease of the sampling ratio. Moreover, in order to further evaluate the classification performance of the proposed MSGWGCN under lower training sample ratios, three representative compared methods with the best classification performance, including ViT, AFS-CNN, and CEGCN, are selected to compare with MSGWGCN on the Flevoland dataset. In the comparison experiments, different training sample ratios (2%, 1%, 0.5%, 0.2%) are set. Experimental results are depicted in the Fig. 13. From the Fig. 13, it can be obviously found that under different training ratios, classification performance achieved by the proposed MSGWGCN is better than other compared methods. Especially, under lower training sample ratio of 0.2%, OA obtained by MSGWGCN on the Flevoland dataset can be higher than 97%, while OAs achieved by other compared methods are lower than 94%. Experimental results prove that the proposed MSGWGCN can still have excellent classification performance under lower training ratio.

5) Generalization Performance: In order to validate the generalization performance of the proposed classifier, some comparison studies are designed. In the novel designed experiments, one of the three datasets is used as the training data, and the predictions are conducted on the remaining two datasets to test the generalization performance. Before prediction, only a small number of samples (20 samples per category) from the test datasets are used to fine-tune. In this way, generalization performances of different methods including AFS-CNN, CEGCN, ViT, and MSGWGCN are compared, where the classification performances are shown in the Table IX. Fle, San, and ESAR stand for the Flevoland dataset, San Francisco dataset, and ESAR Oberpfaffenhofen dataset, respectively. → stands for the transfer direction. For example, Fle → San means that Flevoland dataset is used as the training dataset and San Francisco dataset is utilized as the test dataset. OA is used to evaluate the generalization performance of classification.

From Table IX, it can be observed that the proposed MSGWGCN achieves the highest classification results on all transfer experiments. When the transfer experiments are conducted from Flevoland dataset to ESAR Oberpfaffenhofen dataset, as well as

from San Francisco dataset to Flevoland dataset, OA obtained by MSGWGCN is 92.01% and 95.62%, respectively, which is much higher than other methods. Therefore, comparative experimental results show that the proposed method has superior generalization performance.

V. Conclusion

In this article, an MSGWGCN is proposed to classify the PolSAR image. The proposed MSGWGCN can combine the advantages of superpixels with different scales and perform obvious effect on the boundary pixel classification. The weighted graph convolutional network is employed in the proposed framework to obtain multiscale superpixel features, which enables to take full advantage of land cover object spatial information in PolSAR images. The MFCFM is developed to fuse the multiscale features at pixel level, which can preserve multiscale information to make the classification results more accurate. Experimental results on the the three datasets prove the effectiveness of MSGWGCN.

The proposed MSGWGCN obtains satisfactory classification results by training with a relatively large number of labeled samples. Due to the strong feature propagation ability of graph convolutional network, samples without labels can still obtain high confidence in category prediction, which can be added as pseudolabeled samples in the iteration training. Therefore, in the future, the combination of superpixels and deep neural networks in semisupervised learning for PolSAR image classification will be explored.

References

[1] S.-W. Chen, X.-C. Cui, X.-S. Wang, and S.-P. Xiao, "Speckle-free SAR image ship detection," *IEEE Trans. Image Process.*, vol. 30, pp. 5969–5983, 2021.

[2] X. Ma, J. Xu, P. Wu, and P. Kong, "Oil spill detection based on deep convolutional neural networks using polarimetric scattering information from sentinel-1 SAR images," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 4204713.

[3] S. Chen, H. Wang, F. Xu, and Y.-Q. Jin, "Target classification using the deep convolutional networks for SAR images," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 8, pp. 4806–4817, Aug. 2016.

[4] J. Geng, X. Deng, X. Ma, and W. Jiang, "Transfer learning for SAR image classification via deep joint distribution adaptation networks," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 8, pp. 5377–5392, Aug. 2020.

[5] S.-W. Chen, "SAR image speckle filtering with context covariance matrix formulation and similarity test," *IEEE Trans. Image Process.*, vol. 29, pp. 6641–6654, 2020.

[6] F. Shang, T. Saito, S. Ohi, and N. Kishi, "Coniferous and broad-leaved forest distinguishing using L-band polarimetric SAR data," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 9, pp. 7487–7499, Sep. 2021.

[7] C.-A. Liu, H. Gong, Y. Shao, Z. Yang, L. Liu, and Y. Geng, "Recognition of salt crust types by means of PolSAR to reflect the fluctuation processes of an ancient lake in Lop Nur," *Remote Sens. Environ.*, vol. 175, pp. 148–157, 2016.

[8] Z. Yang, L. Fang, B. Shen, and T. Liu, "PolSAR ship detection based on azimuth sublook polarimetric covariance matrix," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 15, pp. 8506–8518, 2022.

[9] E. Krogager, "New decomposition of the radar target scattering matrix," *Electron. Lett.*, vol. 18, no. 26, pp. 1525–1527, 1990.

[10] A. Freeman and S. L. Durden, "A three-component scattering model for polarimetric SAR data," *IEEE Trans. Geosci. Remote Sens.*, vol. 36, no. 3, pp. 963–973, May 1998.

[11] Y. Yamaguchi, T. Moriyama, M. Ishido, and H. Yamada, "Four-component scattering model for polarimetric SAR image decomposition," *IEEE Trans. Geosci. Remote Sens.*, vol. 43, no. 8, pp. 1699–1706, Aug. 2005.

[12] S. R. Cloude and E. Pottier, "A review of target decomposition theorems in radar polarimetry," *IEEE Trans. Geosci. Remote Sens.*, vol. 34, no. 2, pp. 498–518, Mar. 1996.

[13] Q. Wu, B. Hou, Z. Wen, and L. Jiao, "Variational learning of mixture wishart model for PolSAR image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 1, pp. 141–154, Jan. 2019.

[14] H. Bi, J. Sun, and Z. Xu, "Unsupervised PolSAR image classification using discriminative clustering," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 6, pp. 3531–3544, Jun. 2017.

[15] J. Qin, Z. Liu, L. Ran, R. Xie, J. Tang, and Z. Guo, "A target SAR image expansion method based on conditional wasserstein deep convolutional GAN for automatic target recognition," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 15, pp. 7153–7170, 2022.

[16] Z. Qi, A. G. O. Yeh, X. Li, and Z. Lin, "A novel algorithm for land use and land cover classification using RADARSAT-2 polarimetric SAR data," *Remote Sens. Environ.*, vol. 118, pp. 21–39, 2012.

[17] W. Miao, J. Geng, and W. Jiang, "Semi-supervised remote-sensing image scene classification using representation consistency siamese network," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5616614.

[18] W. Chen, S. Ouyang, W. Tong, X. Li, X. Zheng, and L. Wang, "GCSANet: A global context spatial attention deep learning network for remote sensing scene classification," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 15, pp. 1150–1162, 2022.

[19] Y. Tai, Y. Tan, S. Xiong, Z. Sun, and J. Tian, "Few-shot transfer learning for SAR image classification without extra SAR samples," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 15, pp. 2240–2253, 2022.

[20] J. Geng, R. Wang, and W. Jiang, "Polarimetric SAR image classification based on feature enhanced superpixel hypergraph neural network," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5237812.

[21] Y. Zhou, H. Wang, F. Xu, and Y.-Q. Jin, "Polarimetric SAR image classification using deep convolutional neural networks," *IEEE Geosci. Remote Sens. Lett.*, vol. 13, no. 12, pp. 1935–1939, Dec. 2016.

[22] S.-W. Chen and C.-S. Tao, "PolSAR image classification using polarimetric-feature-driven deep convolutional neural network," *IEEE Geosci. Remote Sens. Lett.*, vol. 15, no. 4, pp. 627–631, Apr. 2018.

[23] Y. Wang, J. Cheng, Y. Zhou, F. Zhang, and Q. Yin, "A multichannel fusion convolutional neural network based on scattering mechanism for PolSAR image classification," *IEEE Geosci. Remote Sens. Lett.*, vol. 19, 2022, Art. no. 4007805.

[24] H. Dong, L. Zhang, and B. Zou, "Exploring vision transformers for polarimetric SAR image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5219715.

[25] Z. Zhang, H. Wang, F. Xu, and Y.-Q. Jin, "Complex-valued convolutional neural network and its application in polarimetric SAR image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 12, pp. 7177–7188, Dec. 2017.

[26] X. Tan, M. Li, P. Zhang, Y. Wu, and W. Song, "Deep triplet complex-valued network for PolSAR image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 12, pp. 10179–10196, Dec. 2021.

[27] B. Ren, Y. Zhao, B. Hou, J. Chanussot, and L. Jiao, "A mutual information-based self-supervised learning model for PolSAR land cover classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 11, pp. 9224–9237, Nov. 2021.

[28] L. Zhang, S. Zhang, B. Zou, and H. Dong, "Unsupervised deep representation learning and few-shot classification of PolSAR images," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5100316.

[29] C. Wu et al., "Fuzzy SLIC: Fuzzy simple linear iterative clustering," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 31, no. 6, pp. 2114–2124, Jun. 2021.

[30] Y. Guo, L. Jiao, S. Wang, S. Wang, F. Liu, and W. Hua, "Fuzzy superpixels for polarimetric SAR images classification," *IEEE Trans. Affect. Comput.*, vol. 26, no. 5, pp. 2846–2860, Oct. 2018.

[31] Y. Guo et al., "Adaptive fuzzy learning superpixel representation for PolSAR image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5217818.

[32] T. Gadhiya and A. K. Roy, "Superpixel-driven optimized wishart network for fast PolSAR image classification using global k-means algorithm," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 1, pp. 97–109, Jan. 2020.

[33] J. Geng, X. Ma, J. Fan, and H. Wang, "Semisupervised classification of polarimetric SAR image via superpixel restrained deep neural network," *IEEE Geosci. Remote Sens. Lett.*, vol. 15, no. 1, pp. 122–126, Jan. 2018.

[34] Y. Yu, J. Li, Q. Yuan, Q. Shi, H. Shen, and L. Zhang, "Coupling dual graph convolution network and residual network for local climate zone mapping," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 15, pp. 1221–1234, 2022.

[35] Z. Zhou, C. Fu, and R. Weibel, "Move and remove: Multi-task learning for building simplification in vector maps with a graph convolutional neural network," *ISPRS J. Photogrammetry Remote Sens.*, vol. 202, pp. 205–218, 2023.
[36] S. Ding, F. Feng, X. He, Y. Liao, J. Shi, and Y. Zhang, "Causal incremental graph convolution for recommender system retraining," *IEEE Trans. Neural Netw. Learn. Syst.*, early access, 2022, doi: 10.1109/TNNLS.2022.3156066.
[37] F. M. Bianchi, D. Grattarola, L. Livi, and C. Alippi, "Graph neural networks with convolutional ARMA filters," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 7, pp. 3496–3507, Jul. 2022.
[38] J. Cheng, F. Zhang, D. Xiang, Q. Yin, and Y. Zhou, "PolSAR image classification with multiscale superpixel-based graph convolutional network," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5209314.
[39] F. Liu, J. Wang, X. Tang, J. Liu, X. Zhang, and L. Xiao, "Adaptive graph convolutional network for PolSAR image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5208114.
[40] S. Ren and F. Zhou, "Semi-supervised classification for PolSAR data with multi-scale evolving weighted graph convolutional network," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 14, pp. 2911–2927, 2021.
[41] D. A. Spielman, "Spectral graph theory and its applications," in *Proc. 48th Annu. IEEE Symp. Found. Comput. Sci.*, 2007, pp. 29–38.
[42] D. K. Hammond, P. Vandergheynst, and R. Gribonval, "Wavelets on graphs via spectral graph theory," *Appl. Comput. Harmon. Anal.*, vol. 30, no. 2, pp. 129–150, 2011.
[43] G. Sinclair, "The transmission and reception of elliptically polarized waves," *Proc. IRE*, vol. 38, no. 2, pp. 148–151, Feb. 1950.
[44] S. H. Yueh, S. V. Nghiem, and R. Kwok, "Symmetrization of cross-polarized responses in polarimetric radar images using reciprocity," *IEEE Trans. Geosci. Remote Sens.*, vol. 31, no. 6, pp. 1180–1185, Nov. 1993.
[45] H. Dong, L. Zhang, D. Lu, and B. Zou, "Attention-based polarimetric feature selection convolutional network for PolSAR image classification," *IEEE Geosci. Remote Sens. Lett.*, vol. 19, 2022, Art. no. 4001705.
[46] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Süsstrunk, "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.
[47] P. Hart, "The condensed nearest neighbor rule (Corresp.)," *IEEE Trans. Inf. Theory*, vol. 14, no. 3, pp. 515–516, May 1968.
[48] J. Wang, B. Hou, L. Jiao, and S. Wang, "POL-SAR image classification based on modified stacked autoencoder network and data distribution," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 3, pp. 1678–1695, Mar. 2020.
[49] X. Liu, L. Jiao, F. Liu, D. Zhang, and X. Tang, "PolSF: PolSAR image datasets on San Francisco," in *Proc. Int. Conf. Intell. Sci.*, 2022, pp. 214–219.
[50] Y. Chen, L. Jiao, Y. Li, and J. Zhao, "Multilayer projective dictionary pair learning and sparse autoencoder for PolSAR image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 12, pp. 6683–6694, Dec. 2017.
[51] C. Lardeux et al., "Support vector machine for multifrequency SAR polarimetric data classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 47, no. 12, pp. 4143–4152, Dec. 2009.
[52] X. Tan, M. Li, P. Zhang, Y. Wu, and W. Song, "Complex-valued 3-D convolutional neural network for PolSAR image classification," *IEEE Geosci. Remote Sens. Lett.*, vol. 17, no. 6, pp. 1022–1026, Jun. 2020.
[53] R. Shang, J. He, J. Wang, K. Xu, L. Jiao, and R. Stolkin, "Dense connection and depthwise separable convolution based CNN for polarimetric SAR image classification," *Knowl. Based Syst.*, vol. 194, 2020, Art. no. 105542.
[54] Q. Liu, L. Xiao, J. Yang, and Z. Wei, "CNN-enhanced graph convolutional network with pixel-and superpixel-level feature fusion for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 10, pp. 8657–8671, Oct. 2021.

Ru Wang received the B.S. degree in electronic information engineering from Northwestern Polytechnical University, Xi'an, China, in 2021. She is currently working toward the M.S. degree in electronic information with the School of Electronics and Information, Northwestern Polytechnical University, Xi'an, China.

Her research interests include PolSAR image classification and deep learning.

Yinju Nie received the B.S. degree in automation from Anhui University, Hefei, China, in 2023. He is currently working toward the M.S. degree in artificial intelligence with the School of Electronics and Information, Northwestern Polytechnical University, Xi'an, China.

His research interests include PolSAR object detection and deep learning.

Jie Geng (Member, IEEE) received the B.S. degree in electronic and information engineering from the Dalian University of Technology, Dalian, China, in 2013, and the Ph.D. degree in signal and information processing from the School of Information and Communication Engineering, Dalian University of Technology, in 2018.

He is currently an Associate Professor with the School of Electronics and Information, Northwestern Polytechnical University, Xi'an, China. His research interests include SAR image processing, deep learning, and few-shot learning.