# Multiscale Gaussian Process-Driven Graph Convolutional Neural Network for Polarimetric SAR Image Classification

Ze-Chen Li, Heng-Chao Li, *Senior Member, IEEE*, Jing-Hua Yang, *Member, IEEE*, Fan Zhang, *Senior Member, IEEE*, and Jie Pan

***Abstract*—By focusing on the structure exploration and information propagation from non-Euclidean data space, graph convolutional neural network (GCN), which can extract abundant and discriminative features, has been a valuable topic in polarimetric synthetic aperture radar (PolSAR) image field. However, the existing GCN-based PolSAR classification methods have high computational cost, may easily be prone to over-smoothing or over-fitting, and inadequately learn the polarimetric property. To address these issues, we propose a polarimetric rotation-based multiscale Gaussian process-driven GCN (MGPGCN) for semi-supervised PolSAR image classification. First, for addressing the over-smoothing and over-fitting problems, the Gaussian process (GP) is introduced into GCN framework, which can fit the underlying feature distribution rather than calculating specific values of weights in conventional GCN. Second, we extend the multiscale layer architecture and design the multiscale kernel for improving the representation capability and fully leveraging neighborhood information of GCN. Third, to mitigate the effect caused by noise or imaging angle, a superpixel-level polarimetric rotation-based feature enhancement strategy is designed. With this strategy, the characteristic of each terrain type is more salient, and the representation capability of GP kernel can be further improved. Comprehensive experiments on four PolSAR datasets firmly demonstrate that the proposed MGPGCN can achieve better performance compared with some widely-used GCN-based classification methods.**

***Index Terms*—Gaussian process (GP)-driven graph convolutional network (GCN), polarimetric rotation (PR) domain, polarimetric synthetic aperture radar (PolSAR), semi-supervised classification.**

Received 21 March 2025; revised 15 June 2025 and 19 July 2025; accepted 2 August 2025. Date of publication 11 August 2025; date of current version 28 August 2025. This work was supported in part by the National Natural Science Foundation of China under Grant 62271418 and Grant 12401605, in part by the Natural Science Foundation of Sichuan Province under Grant 2023NSFSC0030 and Grant 2024NSFSC1467, in part by the Postdoctoral Fellowship Program of CPSF under Grant GZC20232198, in part by the China Postdoctoral Science Foundation under Grant 2024M752661, and in part by the Fundamental Research Funds for the Central Universities under Grant 2682024CX017. *(Corresponding author: Jing-Hua Yang.)*

Ze-Chen Li, Heng-Chao Li, and Jing-Hua Yang are with the School of Information Science and Technology, Southwest Jiaotong University, Chengdu 611756, China (e-mail: hcli@home.swjtu.edu.cn; yangjinghua110@126.com).

Fan Zhang is with the College of Information Science and Technology, Beijing University of Chemical Technology, Bejing 100029, China.

Jie Pan is with the Aerospace Information Research Institute, Chinese Academy of Sciences, Bejing 100094, China (e-mail: panjie@aircas.ac.cn).

Digital Object Identifier 10.1109/JSTARS.2025.3597776

## I. Introduction

As one of the most advanced remote sensing technologies, polarimetric synthetic aperture radar (PolSAR) has many advantages, such as long-range, strong penetrating ability, and working in all-weather and multiclimate conditions. By acquiring specific scattering echoes, PolSAR has ability to effectively describe the physical properties and scattering mechanism of diverse land covers. Meanwhile, instead of utilizing only a single mode by SAR, the matrix-variate PolSAR data points are generally acquired in combination with different polarization modes, thus yielding more polarimetric information to be included. Hence, the observed PolSAR data has provided wide applications for both military and civilian fields, such as ship detection [1], land cover classification [2], [3], [4], [5], disaster management [6], and vegetation monitoring [7].

As an important role in PolSAR image interpretation, PolSAR image classification not only assigns various category labels to pixels according to the backscattering information, but also provides the discriminative information for many downstream tasks [8]. For example, high-resolution classification maps can facilitate precise target detection, and temporal classification results support the change detection task for disaster assessment and other tasks. Moreover, category-specific scattering features can enhance parameter inversion accuracy in environmental monitoring. Hence, PolSAR image classification methods have been intensively developed in recent years, which can be classified as the unsupervised, supervised, and semi-supervised methods.

The unsupervised methods, which can be regarded as the cornerstone throughout the whole development of PolSAR image classification, primarily encompass polarimetric decomposition-based methods, and statistical modeling-based methods. Concretely, the polarimetric decomposition aims to factorize the polarimetric coherence or covariance matrices into fundamental components representing distinct geometric structures and scattering mechanisms, typically expressed as linear combinations [9], [10], [11], [12]. Based on phase stability requirements, they are categorized into: coherent (e.g., Pauli, Krogager, and Cameron decompositions) or incoherent (e.g., Freeman, Yamaguchi, and Cloude-Pottier decompositions) techniques [13], [14], [15]. For the statistical modeling-based methods, they leverage the unique statistical properties to model

PolSAR backscattering data, and assume each data point is sampled from a certain distribution (e.g., complex Wishart or complex Gaussian). After estimating the distribution's parameters, the classification is fulfilled by evaluating the probability of a data point belonging to a specific cluster [16]). Although manual annotation of samples is not required, these unsupervised methods cannot obtain very accurate results, and are difficult to deploy on the large-scale dataset.

Inspired by the success of supervised convolutional neural network (CNN) in optical image field to automatically extract abstract and discriminative features, many efforts have been made to develop CNNs in the PolSAR field and subsequently design various modules for adapting to data format or capturing more scattering properties, such as 2D-CNN [17], 3D-CVNN [18], and residual network (ResNet) [19]. Furthermore, some studies also consider to integrate more characteristics of data, e.g., PolCNN [20], manifold CNN [21], and graph-based CNN [22], which leverage multiregion windows, manifold representation, and graph representation for exploring the local/global spatial and structural information. By combining the above model-driven or data-driven strategies, many supervised methods are proposed to break the feature representation bottlenecks, thus improving the classification accuracy and deploying for large-scale PolSAR datasets [23], [24], [25], [26], [27], [28]. However, numerous manually annotated samples are needed in above methods and the collection of labels is a complicated process, thus limiting their application potentials. To address the problem of annotation-throughput tradeoff, the semi-supervised methods represented by graph neural network [29] and graph convolutional network (GCN) [30], have obtained rapid development and improved applicability in recent years. Hence, we focus on the GCN-based semi-supervised classification.

Specifically, since the convolution operations in CNN and its variants are only imposed on the fixed-size and regular square patches, the surrounding pixels are implicitly assumed to have the same labels of center pixel, yielding the boundary information of PolSAR image not being considered. Hence, the CNN-based methods mentioned above may fail to guarantee the classification performance on heterogeneous areas and result in poor generalization ability. To improve classification performance on the heterogeneous area, Chen et al. [31] introduced the GCN in PolSAR field, with which the pixel-level classification is developed to the superpixel-level for leveraging the boundary information in heterogeneous areas. Simultaneously, since GCN can flexibly convolve over regions with arbitrary shapes, i.e., capturing the graph inductive bias, the spatial topology of PolSAR land covers can be further explored and the correlations between long-range data points with the same scattering characteristics can be further leveraged. Benefiting from these advantages, many researchers designed various GCNs to classify PolSAR images. For example, Liu et al. [32] proposed an adaptive GCN, in which each obtained superpixel is corrected by a constructed subgraph, such that the classification performance can be improved by the better superpixel nodes. Similarly, the fuzzy GCNs developed by [33] and [34] also focused on improving the accuracy of superpixel segmentation results, such that the physical property or distance measurement of those undetermined superpixels is utilized to construct a fuzzy graph for extracting more intuitive features. On the other side, some researchers were expected to make GCN applicative on various scenes with limited sample information or data form, e.g., the authors in [35] and [36] designed tensor-GCN and meta-GCN, respectively. By introducing more structural prior information, the PolSAR observations can be further learned. Besides, there are also some studies devoted to improving the representation ability of constructed adjacency matrix in GCN, such as weighted GCN [37] and dual-branch GCN [38].

Although the above GCNs and their extended versions achieve good PolSAR classification performance by improving the representation ability of constructed superpixel/graph or by leveraging more priors, there are still some issues. For instance, since the graph convolutional operations of these GCNs and their extended visions have the same form as that of traditional GCNs, their graph convolutional layers are easily to be over-smoothing in view of utilizing the same nodes in different connection subgraphs. In such case, the latent representation of each node will be imposed to the same value (i.e., the same location in the feature space), and the classification performance may decay with an increasing number of iterations. Meanwhile, considering the influence of imaging angle and environment, the same terrain types in a PolSAR image may show different scattering characteristics (or different terrain types show very similar scattering characteristics). However, these GCN-based methods do not design the corresponding modules for improving the identification accuracy of these "variant" targets.

In fact, some works have illustrated that integrating the finite mixture models, stochastic processes, or other Bayesian probability techniques into deep networks is an effective way to mitigate the over-smoothness and over-fitting phenomenons [39], [40], [41]. Inspired by them, we introduce the Gaussian Process (GP) into GCN for constructing and extending the GP-driven GCN (GPGCN), which is suitable for dealing with over-smoothness and over-fitting problems. By modeling the GCN layer and its variants with GPs, the latent units of the network layers can be assumed to be infinite (i.e., the layer widths tend to infinity), and accordingly, under the central limit theorem, the GP over Gaussian distributions is appropriate for fitting the infinite neural units [42], [43], [44]. Therefore, the calculation of graph convolutional operation in each GCN layer can be transmitted to the estimation of the GP kernel, and the network will fit the underlying feature distribution rather than learning specific values of weights like conventional GCNs. With this strategy, the effectiveness of nodes/subgraphs can be guaranteed and the layer architecture becomes programmable and scalable. However, due to the limitation of the GP kernel, GPGCN is difficult to directly deal with the large-scale dataset. Hence, we employ the superpixel-level classification strategy to ensure the effectiveness on large-scale datasets. Simultaneously, for processing the above "variant problem," a superpixel-level polarimetric rotation (PR)-based strategy is designed to enhance the input feature of the network. By measuring the polarimetric similarity under different angles, the enhanced features could include more polarimetric information and rotation invariant information, thus yielding the variant targets being correctly

recognized. The experimental results also validate the effectiveness of introducing GP and designing the superpixel-level feature enhancement strategy. In conclusion, the main contributions of this article are three-fold as follows.

1) A novel PR-based multiscale GPGCN (MGPGCN) is developed for semi-supervised PolSAR classification. Compared with the existing GCNs-based PolSAR classification methods, MGPGCN models graph convolutional layers by GPs for obtaining the effective kernel, such that the latent feature distribution is learned rather than the specific values of weights. As such, the proposed method can consider the uncertainty, address the over-smoothness, and improve the representation ability.
2) To further explore the contextual information of PolSAR image, we design a multiscale layer architecture to improve the learning capability and robustness of MGPGCN. Meanwhile, for better applying the proposed method on large-scale datasets, the superpixel technique is also introduced to significantly reduce the computational burden of GP kernel and constructed graph.
3) For improving the feature discriminant of MGPGCN layer and mitigating the effect of noise, a PR-based feature enhancement strategy is proposed. In this way, the superpixel-level samples are mapped into the PR domain, and the similarity degrees of the input sample and its projection are determined by leveraging various scattering measurement techniques. With the enhanced features, the characteristic of each terrain type is more salient, and the representation capability of the GP kernel can be further improved, thus obtaining better classification results.

The rest of this article is organized as follows. Section II presents the preliminaries of PolSAR data. In addition, GCN and GP are briefly introduced. In Sections III, the proposed method with PR-based feature enhancement strategy and multiscale strategy will be described in detail. Comprehensive quantitative analysis and evaluation of the proposed methods are implemented in Section IV. Finally, Section V concludes this article.

## II. Preliminaries

### A. PolSAR Data Representation

PolSAR system measures the back scattering signal of land covers by different combinations of polarimetric modes, generally including horizontal polarization ($h$) and vertical polarization ($v$). For the full polarization observation, an arbitrary pixel in PolSAR image can be represented as a polarimetric coherency matrix $\mathbf{T}$, which is given as

$$\mathbf{T} = \left\langle \mathbf{k}_L \cdot \mathbf{k}_L^H \right\rangle = \begin{bmatrix} T_{11} & T_{12} & T_{13} \\ T_{21} & T_{22} & T_{23} \\ T_{31} & T_{32} & T_{33} \end{bmatrix} \tag{1}$$

where $\mathbf{k}_L = [S_{hh}, \sqrt{2}S_{hv}, S_{vv}]^H$ is the polarimetric target vector under the assumption of monostatic reciprocity, and $S_{ab}$ contains the complex-valued backscattering coefficient of the terrain target transmitted (received) in $a(b)$ polarization. $\langle\cdot\rangle$ represents the multilook averaging, and the superscript $H$ is the conjugate transpose operation. Polarimetric coherency matrix $\mathbf{T}$, which is a Hermitian symmetric positive semidefinite matrix, contains the real-valued diagonal elements and complex-valued nondiagonal elements. As a general choice in the most deep-learning methods, the upper triangular elements of $\mathbf{T}$ are selected as the input feature vector, i.e., $x = [T_{11}, \mathcal{R}_{12}, \mathcal{F}_{12}, \mathcal{R}_{13}, \mathcal{F}_{13}, T_{22}, \mathcal{R}_{23}, \mathcal{F}_{23}, T_{33}]$, where $\mathcal{I}_*$ and $\mathcal{R}_*$ are the imaginary and real parts of $T_*$, respectively. Moreover, due to the effect of polarimetric orientation angles, same terrain types may show different scattering characteristics. For fully extracting features of these variant targets, some efforts in [45], [46], [47] have been made to map the polarimetric coherency matrix $\mathbf{T}$ into the PR domain for capturing the PR-invariant features, which is defined as follows:

$$\mathbf{T}^{(R)} = \mathbf{R}\mathbf{T}\mathbf{R}^H \tag{2}$$

where rotation matrix $\mathbf{R}$ with the rotation angle $\alpha \in [-\pi/2, \pi/2)$ has the form as follows:

$$\mathbf{R} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(2\alpha) & \sin(2\alpha) \\ 0 & -\sin(2\alpha) & \cos(2\alpha) \end{bmatrix}. \tag{3}$$

Hence, by selecting various rotation angle $\alpha$, the corresponding $\mathbf{T}^{(R)}$ including abundant polarimetric information can be obtained, which can effectively help to distinguish these variant targets.

### B. Graph Convolutional Network

GCN is a powerful semi-supervised learning method, which is developed from the CNN and can leverage the graph-structured data. The input of GCN is an undirected graph, which is denoted as $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{A})$, where $\mathcal{V}$ and $\mathcal{E}$ are the sets of nodes and edges with $N = |\mathcal{V}|$ and $M = |\mathcal{E}|$, respectively. $A \in \mathbb{R}^{N\times N}$ is the adjacency matrix of $\mathcal{G}$. The spectral convolution on graphs can be represented as the multiplication of the data $x$ with a filter $g_\theta = \text{diag}(\theta)$, i.e.,

$$g_\theta * x = U g_\theta U^T x. \tag{4}$$

Here, eigenvector matrix $U$ is derived from the normalized graph Laplacian $L = \mathbf{I} - D^{-1/2}AD^{-1/2} = U\Lambda U^T$, $D$ is the diagonal matrix $D_{ii} = \sum_j A_{ij}$, $\mathbf{I}$ represents the identity matrix, diagonal matrix $\Lambda$ includes the eigenvalues of $L$, and $U^T x$ is the graph Fourier transform. Based on the above spectral convolution, the architecture of a GCN layer is presented as

$$X^{(l)} = f(AX^{(l-1)}W^{(l)} + b^{(l)}) \tag{5}$$

where $l$ denotes the $l$th layer in the GCN, and $X^{(l-1)}$ and $X^{(l)}$ are the input and output features, respectively. $W^{(l)}$ and $b^{(l)}$ are the weight matrix and bias vector, respectively. $f$ represents the nonlinear activation function, such as ReLU. The representation of each node is given as

$$x_i^{(l)} = \sum_{j\in\mathcal{N}_i\cup\{i\}} \frac{1}{\sqrt{\deg(i)}\cdot\sqrt{\deg(j)}} (\Theta^{(l)} \cdot x_j^{l-1}) \tag{6}$$

where $\deg(i)$ is the degree of $i$th node, and $\Theta$ is set of the trainable weights. By constructing the PolSAR feature to the graph-structured data, the correlations of long-range pixels that may have similar scattering mechanisms in the image can be further learned. Hence, developing the GCN-based PolSAR classification methods have become a valuable topic.

### C. Gaussian Process

GP is defined as an infinite collection of random variables, whose arbitrary finite subset is jointly Gaussian distributed. By determining the mean function $m(x)$ and covariance function (kernel) $K = k_\Lambda(x, x')$, the GP can be completely specified, where $x$ and $x'$ are two possible inputs, and $\Lambda$ is the hyperparameter set for parameterizing the kernel. Hence, the GP is denoted as follows:

$$f_{\text{gp}}(x) \sim \text{GP}(m(x), K). \tag{7}$$

Due to the posterior consistency, tractable posterior and wide support, GP is one of the most widely used Bayesian statistical modeling techniques. By applying GP into GCN, the over smoothness problem can be effectively mitigated. Generally, we let mean function $m(x) = 0$ for reducing the complexity. Hence, after obtaining the appropriate kernel, the classification label of a test point can be calculated with the known data and the joint probability density according to Bayesian principle.

## III. Methodology

In this section, we elaborate our MGPGCN model and derive the corresponding multiscale GP kernel. And then, the PR-based feature enhancement strategy will be discuss in detail, which can effectively extract more discriminative information to fulfill the classification task.

### A. Multiscale Gaussian Process-Driven GCN

As a powerful semi-supervised method, GCN can effectively learn the graph-structured data and capture the graph inductive bias, which has been developed in recent years. However, it also has some disadvantages. For example, the labels of few training samples under semi-supervised strategy cannot be fully transmitted on the whole graph, and particularly, within a deep structure, the adjacent nodes will be much similar to the center node due to the convolutional operation, which is the over smoothing phenomenon. Meanwhile, for the large-scale dataset, its training process has a high time overhead.

To address the above issues, some researchers introduce GP as well as other statistical-modeling techniques to deal with these problems. By modeling the GCN layer from statistical perspective, GPGCN learns the distribution of weights rather than specific values, and the uncertainty can be considered. Inspired by them, we construct the PR-based MGPGCN. Instead of directly modeling GCN layers, we introduce the GCN via Initial residual and Identity mapping (GCNII) [48] architecture for improving the representation capability, and design corresponding multiscale kernel for exploring the contextual information. In such cases, the model becomes more robust and more neighborhood correlations can be captured.

Specifically, instead of using traditional GCN layer, GCNII architecture is introduced to reserve more polarimetric characteristics of original input, whose layer is shown as follows:

$$\mathbf{X}^{(l)} = f\left(\left[(1-\gamma_l)\mathbf{A}\mathbf{X}^{(l-1)} + \gamma_l\mathbf{X}^{(0)}\right]\left[(1-\delta_l)\mathbf{I}_n + \delta_l\mathbf{W}^{(l)}\right]\right) \tag{8}$$

where $\gamma_l$ and $\delta_l$ are two hyperparameters. $\mathbf{I}_n$ and $A$ are the $n \times n$ identity matrix and adjacency matrix, respectively. $\mathbf{X}^{(0)}$ is the original input of network. It can be observed that there are two terms on the right-hand side of (8). The first term is an initial residual connection like the skip connection in ResNet, which allows the features in each layer to preserve some input information for avoiding the over-fitting. By empirically fixing its hyperparameter $\gamma_l$ (generally let $\gamma_l = 0.1$ or 0.2), each node's final representation can retain at least a fraction of $\gamma_l$ from the input feature even if many layers are stacked, thus keeping the balance of model generalization and feature fusion. The second term denotes the identity mapping, which ensures that GCNII obtains at least the same performance as GCN. Its hyperparameter $\delta_l$ imposes the regularization on weight matrix $\mathbf{W}$. By choosing GCNII rather than the layer architecture of GCN, the network could obtain better extensibility and feature extraction ability.

For fully exploring the local spatial context, we model (8) by GP and extend it to the multi-scale version, where the abundant neighborhood information can be incorporated and the over-smoothing problem can be further avoided. By observing (5) and (8), since the update rule in GCNII layer can be regarded as a linear transformation of that in GCN layer, the corresponding kernel can also be calculated by linear transformation. Hence, we first calculate the kernel of GPGCN, based on which we calculate the kernel of MGPGCN.

*1) Constructing GPGCN:* When modeling GCN by GP, the convolutional architecture in (5) is first rewritten as the following element-wise form:

$$y_i^{(l)}(x) = \sum_{j=1}^{d_l-1} W_{ji}^{(l)} x_j^{(l-1)}(x)$$
$$z_i^{(l)}(x) = b_i^{(l)} + \sum_{v\in\mathcal{V}} A_{xv} y_i^{(l)}(v), \quad x_i^{(l)}(x) = f(z_i^{(l)}(x)). \tag{9}$$

Here, $x_i^{(l)}$ (or $z_i^{(l)}$) represents the postactivation (or preactivation) feature corresponding to $i$th index in the $l$th GCN layer, where $x^{(0)}(x)$ (or $z^{(L)}(x)$) denotes the original input vector (or final vector). Accordingly, the matrix notation is given as

$$Y_{N\times d_l}^{(l)} = X_{N\times d_{l-1}}^{(l-1)} W_{d_{l-1}\times d_l}^{(l)}$$
$$Z_{N\times d_l}^{(l)} = \mathbf{1}_{N\times 1} b_{1\times d_l}^{(l)} + A_{N\times N} Y_{N\times d_l}^{(l)}, \quad X_{N\times d_l}^{(l)} = \phi(Z_{N\times d_l}^{(l)}) \tag{10}$$

where $d_{(l-1)}$ and $d_l$ denote the widths of $(l-1)$th and $l$th layers, respectively. When the layers are assumed to be infinitely wide,

the output of GCN (i.e., $z^{(L)}$) can be treated as the multioutput GP. This is because we could let the weight $w_{ij}^{(l)}$ and bias $b_i^{(l)}$ follow the zero mean-Gaussian distributions with variances $\sigma_w^2/d_{l-1}$ and $\sigma_b^2$, since columns of $Z^{(l-1)}$ are independent identically distributed (i.i.d.), each column of the postactivation feature $X^{(l-1)}$ is also i.i.d. In such case, for $(x_1, \cdots, x_N)$, $[y_i^{(l)}(x_1), \cdots, y_i^{(l)}(x_N)]$ obey the joint multivariate Gaussian distribution, and keep identically distributed for different $i$. Hence, $y_i^{(l)}(x_1), \cdots, y_i^{(l)}(x_N)$ can be regarded as a GP.

*2) Calculating the kernel of GPGCN:* For different $i$ and $\tilde{i}$, though $y_i^{(l)}(x)$ and $y_{\tilde{i}}^{(l)}(\tilde{x})$ may share the same input $X^{(l-1)}$, their jointly distribution has the zero covariance, i.e.,

$$\begin{aligned}&\mathrm{Cov}(y_i^{(l)}(x), y_{\tilde{i}}^{(l)}(\tilde{x}))\\&= \sum_{j=1}^{d_{l-1}}\sum_{\tilde{j}=1}^{d_{l-1}} Cov\left(W_{ji}^{(l)}x_j^{(l-1)}(x), W_{\tilde{j}\tilde{i}}x_{\tilde{j}}^{(l-1)}(\tilde{x})\right)\\&= 0\end{aligned} \tag{11}$$

which means $y_i^{(l)}(x)$ and $y_{\tilde{i}}^{(l)}(\tilde{x})$ are independent. Thus, due to $Y^{(l)}$ has the i.i.d. GP columns, each column in $Z^{(l)}$ follows the i.i.d. GP with the form of $\mathcal{N}(0, K^{(l)})$.

Following the decoupling algorithm in [49], since each $z_i^{(l)}$ obeys multivariate Gaussian $\mathcal{N}(0, K^{(l)})$ and are independent, for $x$ and $\tilde{x}$ (which are from the same channel and different nodes), their preactivations $z_j(x)$ and $z_j(\tilde{x})$ are different activated values and still follow the $\mathcal{N}(0, K^{(l)}(x, \tilde{x}))$, and then, according to the laws of large numbers, when the layer widths tend to infinity (i.e., $n^{(l)} \to \infty$), the mean of nodes converge to the expectation, which has the form of $K^{(l)} \to \mathbb{E}[\phi(z_j(x))\phi(z_j(\tilde{x}))]$. After we stack the bias in convolutional operations, the GP kernel $K^{(l)}$ can be computed recursively according to the covariance of $Y^{(l)}$ as follows:

$$\begin{aligned}&\mathbb{E}[y_i^{(l)}(x)y_i^{(l)}(\tilde{x})] = \sigma_w^2 C^{(l-1)}(x, \tilde{x})\\&C^{(l-1)}(x, \tilde{x}) = \mathbb{E}_{z\sim\mathcal{N}(0, K^{(l-1)})}[\phi(z_j(x))\phi(z_j(\tilde{x}))]\end{aligned} \tag{12}$$

where $\mathbb{E}[\cdot]$ denotes the expectation of $[\cdot]$. By selecting various forms for $C^{(0)}$, the correlations between features can be captured, such as the inner product $C^{(0)}(x, x') = xx'/d_0$ or the squared exponential $C^{(0)}(x, x') = \exp(-1/2\sum_{j=1}^{d_0}(x_j - x'_j/l_j))^2$. According to the correlations between $Y^{(l)}$ and $Z^{(l)}$ in (10), since weight $W^{(l+1)}$ is independent and follows $\mathcal{N}(0, \sigma_w^2/d_l)$, the preactivation covariance is factorized as

$$K^{(l)} = \sigma_b^2 \mathbf{1}_{N\times N} + A\ \mathrm{Cov}(Y_i^{(l)})A^T \tag{13}$$

where $\mathrm{Cov}(Y_i^{(l)})$ is denoted as the following form, which is rewritten with the matrix form, i.e.,

$$\mathrm{Cov}(Y_j^{(l)}) = \sigma_w^2 \mathbb{E}[X_j^{(l-1)} X_j^{(l-1)^T}] = \sigma_w^2 C^{(l-1)}. \tag{14}$$

By substituting the (14) into (13), we can determine the recursive formula, and the kernel $K^{(l)}$ can be estimated by

$$K^{(l)} = \sigma_b^2 \mathbf{1}_{N\times N} + \sigma_w^2 A C^{(l-1)} A^T. \tag{15}$$

Fig. 1(a) shows the principle of modeling GCN layes by GP. Compared with the conventional graph convolutional operation in GCN, the input features in GP are assumed to follow the Gaussian distributions, and their corresponding output features are assumed to follow the joint multivariate Gaussian distribution. In such case, the mapping between input and output can be described by GP, which is determined by the kernel function. Hence, GP is regarded as a function over all Gaussian distributions. With this strategy, the learning of specific values of weights in conventional graph convolutional operation is replaced by the estimation of Gaussian kernel in GP [i.e., Fig. 1(b)]. With the determined kernel, the statistical characteristic of graph structure input can be explored [44].

*3) Extending the Kernel to Multiscale Version:* As elaborated before, the processing of GP kernel on MGPGCN can be also obtained from GPGCN by the corresponding transformation according to (12)–(15). In particular, the multiscale information is also integrated into kernel for constructing multiscale kernel. Thus, compared with the GCN layer in (5) and the GCNII layer in (8), the corresponding transformation from (5) to (8) is imposed on the kernel. With changing the form of (13) or (15), we could acquire the final transformation on kernel. Consequently, the corresponding transformation steps are shown as follows, i.e.,

GCNII: $X \leftarrow ((1-\gamma)Af(X) + \gamma X^{(0)})((1-\delta)\mathbf{I} + \delta W)$.

1) The nonlinear activation function $f$ on $X$ corresponds to the transformation on GP kernel $f(\tilde{K}) = C^{(l)} = \mathbb{E}_{z_i^{(l)}\sim\mathcal{N}(0,\tilde{K}^{(l)})}[\phi(z_i^{(l)})\phi(z_i^{(l)})^T]$.
2) The left multiplication by $Af(X)$ corresponds to the operation $ACA^T$ on kernel in view of the expectation in (12).
3) Similarly, multiplication coefficient $\gamma X^{(0)}$ corresponds to $\gamma^2\tilde{K}^{(0)}$, and $(1-\delta)I$ corresponds to $(1-\delta)^2$.
4) For the weight term, $\beta W$ corresponds to $\delta^2\sigma_w^2$, where hyperparameter $\sigma_w$ is the variance of network weight.

In conclusion, when modeling GCNII by GP, the graph convolutional rule in (8) is converted into the operation on GP kernel, and the update rule is given as

$$\tilde{K} \leftarrow \left((1-\gamma)^2 ACA^T + \gamma^2\tilde{K}^{(0)}\right)\left((1-\delta)^2 + \delta^2\sigma_w^2\right). \tag{16}$$

According to the above pattern, it could be found that when the architecture of layer is changed, the corresponding operation on kernel is also determined. For instance, the convolutional operations $AX$ corresponds to the kernel operation $AKA^T$, $Af(X)$ corresponds to $ACA^T$, $X + b$ corresponds to $K + \sigma_b^2\mathbf{1}_{N\times N}$, and so on. Therefore, modeling GCN-based layer by GP is programmable and extended, which is similar to that in GP-based CNN.

Here, considering that only utilizing 1-step adjacent nodes may not extract information sufficiently, the multiscale kernel $\tilde{K}$ in (17) is shown as Fig. 2, where $\eta_p$ is the concentration parameter, $K_p$ denotes the kernel derived from $p$-step adjacent nodes, and $p$ is usually set to 3. With this strategy, the multiscale kernel in Fig. 2 can further explore the neighborhood information. By fusing the kernels learned from different adjacent nodes,

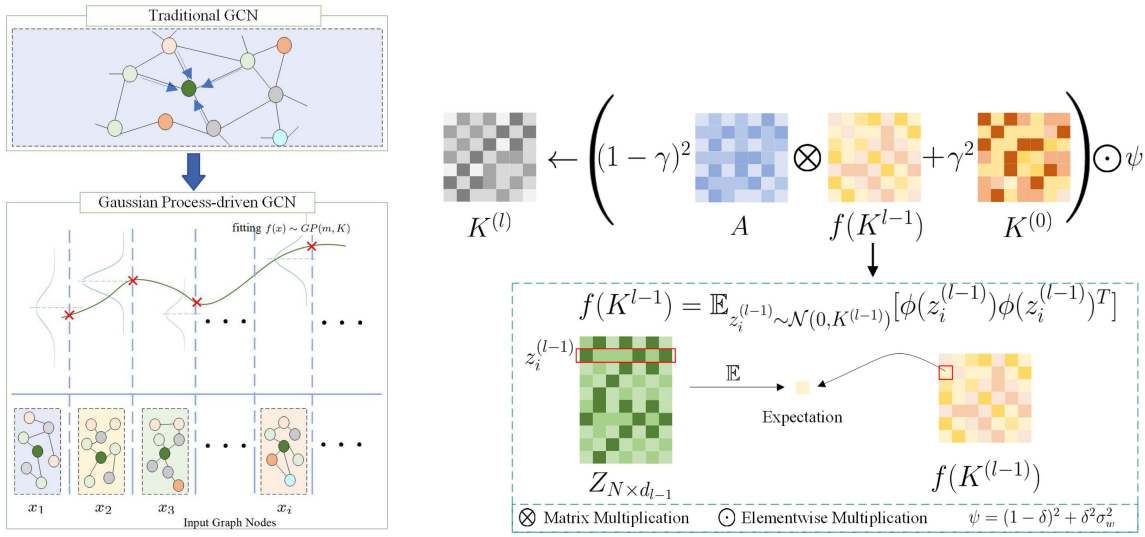



Fig. 1. Illustrations of (a) the GPGCN strategy and (b) the iterative process for kernel.

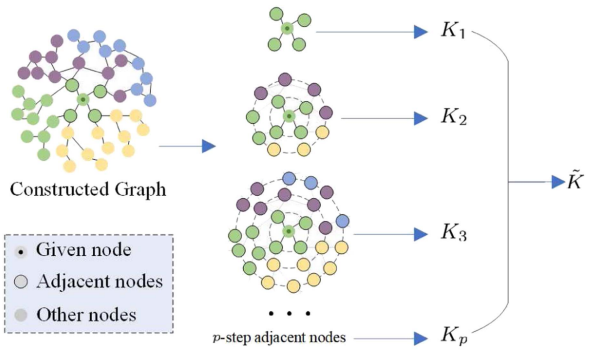



Fig. 2. Illustrations of the multi-scale strategy. By constructing the multiscale kernel, several GP kernels are learned under different scales, and thereby the context information is further leveraged.

the spatial context is further integrated as follows:

$$\tilde{K} = \eta_1 K_1 + \eta_2 K_2 + \cdots + \eta_p K_p. \tag{17}$$

Finally, by the obtained multiscale GP kernel $\tilde{K}$, the classification results $y^*$ of a test point $x^*$ can be calculated by (18) according to [50]

$$y^* = \tilde{K}_{*b}^{(L)} (\tilde{K}_{bb}^{(L)} + \epsilon I)^{-1} y_b. \tag{18}$$

The computational complexity of the proposed MGPGCN mainly focuses on the calculation of the multiscale kernel, which is $\mathcal{O}(Lp(N_* N + N^2))$. Here, $L$ is the number of layer, $p$ is the number of multiscale kernels, and $N_*$ is the number of nonzero elements in adjacent matrix. Specifically, the update rule of each kernel in fusion of multiscale kernels is shown in (16). The computational complexity of $ACA^T$ is $\mathcal{O}(N_* N + N^2)$, and the complexity of $\gamma^2 \tilde{K}^{(0)}$ is $\mathcal{O}(N^2)$ since $\tilde{K}^{(0)}$ is not sparse matrix. Accordingly, the complexity of two terms in (16) are $\mathcal{O}(N_* N + N^2)$ and $\mathcal{O}(N^2)$, respectively.

Different from the existing GCN-based PolSAR classification methods, the proposed method combines the advantages of GCN and GP, such as: 1) MGPGCN not only considers more graph-structure information than GCN but also learns the distribution of features for mitigating over-smoothing; 2) Due to the fact that the workload of GCNs fall on the training phase, they generally have the time-consuming training process, while MGPGCN does not require a costly training phase and is flexible on test time; 3) Since the convolution operation of each layer is transformed to the update of kernel, the quality of obtained kernel will intensively influence the classification performance. Hence, the integration of multiscale information can improve the representation ability for acquiring a better kernel. Hence, as mentioned above, MGPGCN could obtain a shorter training process, effectively deal with the oversmoothness problem, and incorporate the contextual information and uncertainty.

### B. PR-Based Feature Enhancement

By designing MGPGCN, more contextual information over constructed graph is captured and less original information will be lost during feature extraction process. Furthermore, in order to expand the application scenarios of the proposed MGPGCN, it is also necessary to consider the effects of orientation angle, which may generate the following problems about GP.

*1) Lack of Robustness to Noise:* This problem is exhibited as "data length scale" that limits the classification accuracy of obtained GP kernel. Specifically, GP may not be stable for the large length-scale and large spatial-scale dataset. In hyperspectral or optical image field, the element number of each input feature commonly ranges from tens to hundreds, and these few noise elements may generate low effect during feature

extraction. When processing the PolSAR image by GCNs, we generally utilize SLIC or other superpixel techniques to reduce the spatial scale. However, the upper triangular elements of superpixel-level coherency matrix are selected to construct the nine-elements input vector, which still has a low length-scale. Accordingly, the low length scale may lead the speckle noise to generating a serious impact to GP.

*2) Interpretation Ambiguity Caused by Variant Targets:* Due to the influence of imaging angles, the same terrain types may show different scattering characteristics (or various terrain types may show similar scattering characteristics). This is caused by the polarization distortion due to radar side-looking geometry, and these variant targets are easily confused by the low-quality GP kernel.

Hence, for dealing with these two problems, the PR-based feature enhancement strategy is developed in this section, where the diverse scattering similarity measurements are integrated into the PR field to learning the scattering properties, and then, the measurement results of similarity degree are regraded as the new features concatenated with the original data for improving the discriminant and data's length scale, thus making the speckle noise and variant targets be solved.

Specifically, since the PolSAR image is too large to directly construct the graph adjacency matrix, the SLIC algorithm in [31] and [51] is first employed to generate superpixels. By averaging all $\mathbf{T}$ among a superpixel, the mean coherency matrix can be obtained (denoted by $\tilde{\mathbf{T}}$), which has the same form as that in (1). Subsequently, the rotation-based scattering similarity measurement is introduced as follows:

$$D_s = \frac{\mathrm{tr}(\tilde{\mathbf{T}}^{(R)} S_c^H)}{\mathrm{tr}[\tilde{\mathbf{T}}^{(R)}]\mathrm{tr}[S_c]} \tag{19}$$

where $S_c$ represents the typical scattering models. $\mathrm{tr}(\cdot)$ is the trace of matrix, and $(\cdot)^H$ indicates the Hermitian transpose. $\tilde{\mathbf{T}}^{(R)}$ is calculated according to (2) with rotation angle $\alpha$. Compared with the commonly-used polarimetric target decomposition techniques in [52] and [53], which involve in high computation overhead and limited application scenarios, the scattering similarity measurement developed by [54], [55], and [56] has the simple form and unified scale, and can efficiently obtain the scattering components of target via calculating the similarity between each typical scattering and target. More importantly, due to integrating the PR strategy into (19) by $\tilde{\mathbf{T}}^{(R)}$, the influence of rotation angle can be also considered in the scattering similarity measurement, thus fully extracting the variant features.

Meanwhile, since a single scattering model is not enough for the complex scenes, several typical scattering models (like volume, surface, double, helix) are introduced as follows, i.e.,

$$\text{Volume}: \quad S_{c1} = \frac{1}{30}\begin{bmatrix} 15 & 5 & 0 \\ 5 & 7 & 0 \\ 0 & 0 & 8 \end{bmatrix}, S_{c2} = \frac{1}{4}\begin{bmatrix} 2 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$S_{c3} = \frac{1}{15}\begin{bmatrix} 0 & 0 & 0 \\ 0 & 7 & 0 \\ 0 & 0 & 8 \end{bmatrix}, S_{c4} = \frac{1}{30}\begin{bmatrix} 15 & -5 & 0 \\ -5 & 7 & 0 \\ 0 & 0 & 5 \end{bmatrix}$$

$$\text{Surface}: \quad S_{c5} = \frac{1}{1+|\lambda_1|^2}\begin{bmatrix} 1 & \lambda_1^* & 0 \\ \lambda_1 & |\lambda_1|^2 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

$$\text{Double}: \quad S_{c6} = \frac{1}{1+|\lambda_2|^2}\begin{bmatrix} |\lambda_2|^2 & \lambda_2 & 0 \\ \lambda_1^* & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

and

$$\text{Helix}: \quad S_{c7} = \frac{1}{2}\begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & -j \\ 0 & j & 1 \end{bmatrix}, S_{c8} = S_{c7}^H \tag{20}$$

where $S_{c1}$–$S_{c4}$ are volume scattering models, $S_{c5}$ is surface scattering model, $S_{c6}$ is double scattering model, and $S_{c7}$–$S_{c8}$ are Helix scattering models, respectively [45]. $\lambda_1$ and $\lambda_2$ are two parameters. By substituting $\tilde{\mathbf{T}}^{(R)}$ and $S_{c1}$–$S_{c8}$ into (19), eight responsibility weights are calculated. For example, the responsibility weight $D_{s1}(\tilde{\mathbf{T}}^{(R)}, S_{c1})$ is determined as (21), where $b_1$ is the bias corresponding to $D_{s1}$, and $\mathcal{C}_{\mathcal{O}}$ is a normalized factor that can be omitted during the enhancement process. For simplifying, we use the following vector $S_{E1}$ to denote the coefficient in $D_{s1}$ for each element in $\tilde{\mathbf{T}}$

$$\begin{aligned} D_{s1} = \frac{1}{\mathcal{C}_{\mathcal{O}}}\Big\{ & 15\tilde{\mathbf{T}}_{11} + (10\cos^2\alpha_1 - 10\sin^2\alpha_1)\mathcal{R}(\tilde{\mathbf{T}}_{12}) \\ & + 20\sin(\alpha_1)\cos(\alpha_1)\mathcal{R}(\tilde{\mathbf{T}}_{13}) + \tilde{\mathbf{T}}_{22}[8\sin^2(2\alpha_1) \\ & + 7\cos^2(2\alpha_1)] - \sin(4\alpha_1)\mathcal{R}(\tilde{\mathbf{T}}_{23}) + [8\cos^2(2\alpha_1) \\ & + 7\sin^2(2\alpha_1)]\tilde{\mathbf{T}}_{33}\Big\} + b_1 \end{aligned} \tag{21}$$

$$\begin{aligned} S_{E1} = [ & 15, 10\cos(2\alpha_1), 0, 10\sin(2\alpha_1), 0, \sin^2(2\alpha_1) \\ & + 7, -\sin(4\alpha_1), 0, \cos^2(2\alpha_1) + 7]. \end{aligned} \tag{22}$$

Similarly, the other seven responsibility weights corresponding to $S_{c2}$–$S_{c8}$ are given by (23). Therefore, following the linear transformation in each $S_E$, we could obtain eight weights (i.e., new features), which are concatenated with the original input to generate new input features with 17 elements. Furthermore, we could also employ several group steps $\Delta$ on rotation angles for extending the input to higher dimension, i.e., $\alpha_i^{(E1)} = \alpha_i, \alpha_i^{(E2)} = \alpha_i + \Delta_1, \alpha_i^{(E3)} = \alpha_i + \Delta_2, \cdots$. Whenever a $\Delta$ is introduced, the length scale of the input feature can increase 8. Generally, choosing only one or two groups $\alpha_i^{(E)}$ can effectively avoid the feature redundancy.

The proposed PR-based feature enhancement is shown in Fig. 3. The input data of network is mapped into the PR domain to compute the scattering similarities by several scatter models, and then the learned scattering similarities are concentrated with

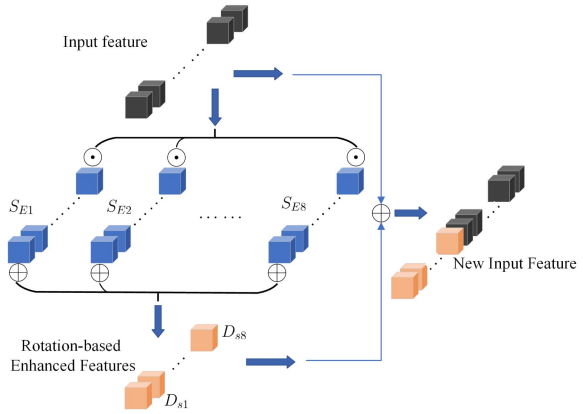



Fig. 3. Illustrations of the rotation-cased feature enhancement strategy. By learning the scattering similarities of each superpixel, the responsibility weights are regarded as the enhanced features.

the original input feature to construct the new input feature. With this strategy, the input feature can contain more polarimetric information and be more discriminative

$$
\begin{aligned}
S_{E2} &= [2,0,0,0,0,1,0,0,1]\\
S_{E3} &= [0,0,0,0,0,7+\sin^2(2\alpha_2),-\sin(4\alpha_2),0,\cos^2(2\alpha_2)\\
&\quad +7]\\
S_{E4} &= [15,-10\cos(2\alpha_3),0,-10\sin(2\alpha_3),0,\sin^2(2\alpha_3)+7,\\
&\quad -\sin(4\alpha_3),0,\cos^2(2\alpha_3)+7]\\
S_{E5} &= [1,2\lambda_1\cos(2\alpha_4),0,2\lambda_1\sin(2\alpha_4),0,\lambda_1^2\sin(4\alpha_4),0,\\
&\quad \lambda_1^2\sin^2(2\alpha_4)]\\
S_{E6} &= [\mathcal{R}^2(\lambda_2)+\mathcal{F}^2(\lambda_2),2\mathcal{R}(\lambda_2)\cos(2\alpha_5),2\mathcal{F}(\lambda_2)\cos(2\alpha_5),\\
&\quad 2\mathcal{R}(\lambda_2)\sin(2\alpha_5),2\mathcal{F}(\lambda_2)\sin(2\alpha_5),\cos^2(2\alpha_5),\sin(4\alpha_5),\\
&\quad 0,\sin^2(2\alpha_5)]\\
S_{E7} &= [0,0,0,0,0,1,0,-2,1]\\
S_{E8} &= [0,0,0,0,0,1,0,2,1]. \qquad (23)
\end{aligned}
$$

Consequently, with this superpixel-level feature enhancement strategy in Fig. 3, the input feature can include more discriminative information for distinguishing variant targets, and the larger length scale also assists GP to eliminate the noise effect. Compared with the traditional feature augmentation techniques in computer vision (CV) (e.g., image rotation, flipping), the polarization rotation enhancement strategy have the distinct advantages in physical interpretability and information exploitation, specifically: 1) Polarization rotation adjusts the polarization basis while strictly preserving the mathematical integrity of scattering matrices under Maxwell equation constraints. However, the geometric transformations in CV disrupt PolSAR pixel spatial relationships and scattering phase coherence; 2) Polarization rotation can selectively amplify specific scattering responses, which can be regarded as the orientation angle compensation to some extent; 3) Traditional rotation in CV introduces geometric artifacts (like aliasing), while polarization rotation suppresses azimuth-dependent scattering fluctuations by adjusting inter-channel complex correlations. Thus, the enhancement strategy in our MGPGCN is constructed on the polarization rotation perspective rather than that in other approaches formed by traditional CV-based enhancement techniques.

**Algorithm 1:** Proposed MGPGCN Algorithm.

**Input:** PolSAR data $\mathbf{T}$, layer number $L$, step length $\Delta$, neighborhood scale $p$, parameters $\gamma$ and $\delta$.
**Initialize:** Calculate superpixel-level polarimetric coherency matrix $\tilde{\mathbf{T}}$ by SLIC. Randomly initialize parameters $\alpha_i$, $\lambda_1$, and $\lambda_2$.
**Step 1:** Enhance input features:
Obtain $\tilde{\mathbf{T}}^{(R)}$ by mapping $\mathbf{T}$ into polarimetric rotation domain via (2).
Calculate the responsibility weights $D_{s1}\sim D_{s8}$ (i.e., enhanced features) for each group of $\alpha_i^{(E)}$ according to (21)-(23) by GCN.
Obtain the new input feature $\tilde{X}$ by the concatenation of the original input and the enhanced features.
**Step 2:** Train the proposed model:
Current layer number $l \le L$ **repeat**
**Expectation-step**: recursively calculate expectation and covariance in (11) and (12).
**Kernel-step**: update each $p$-step GP kernel $K_p$, and determine the multi-scale kernel $\tilde{K}$ via (16) and (17).
**Step 3:** Predict the classification label $y^*$ according to (18).
**Return**: Label $y^*$

The corresponding graph framework of the proposed MGPGCN is exhibited in Fig. 4. First, the PolSAR image is divided into superpixels by SLIC algorithm, and then, each superpixel-level coherency matrix is mapped into the PR domain for exploring the rotation invariance. According to the projection matrix, various scattering similarity measurements under different rotation angles are introduced to acquiring responsibility weights, thus yielding the enhanced features to fully include diverse scattering information. After that, the new input features will be fed into network, where the graph convolutional operation is replaced via the update of GP kernel. By recursively calculating the expectation and covariance in (11) and (12), each kernel under different scales can be obtained. And finally, following the Bayesian criterion, we can predict the label of unlabel test point by (18), and draw the classification map. The complete classification process of MGPGCN is shown in Algorithm 1.

## IV. Experimental Results

### A. Experimental Data and Settings

For quantitatively and qualitatively discussing the classification performance, four real PolSAR data sets are utilized to evaluate the proposed model, which involve in different sensors and include diverse terrain types (e.g., land agriculture,

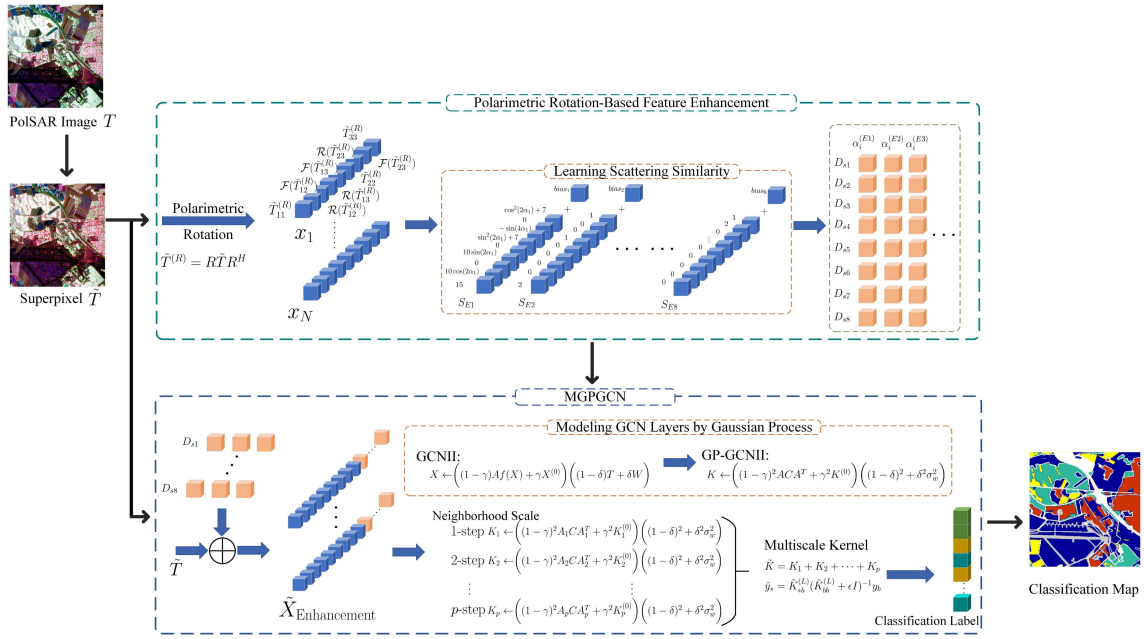
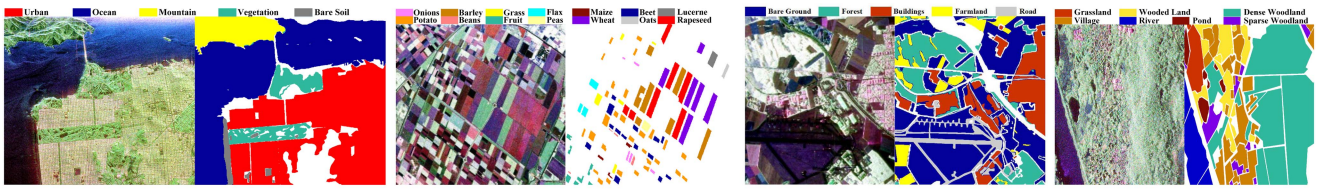



Fig. 4. Framework of the proposed MGPGCN to classify the PolSAR image. First, the observed PolSAR image is segmented into superpixels. Next, the input vectors constructed from these superpixels are mapped into the PR domain to enhance the features. Then, these enhanced features are inputted into the MGPGCN layer to estimate the multiscale GP kernel, which is subsequently used to predict the classification labels.




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

Fig. 5. Pauli RGB images, ground truth images, and color codes. (a) *SanFrancisco* (900×1024). (b) *Flevoland* (920×1024). (c) *Oberpfaffenhofen* (1300×1200). (d) *HaiNan* (1540×1250).

forest, seaside, rivers, and buildings). The Pauli RGB images, color codes and the ground truth maps of these datasets are illustrated in Fig. 5, where the background pixels in the ground truth maps are marked by white. To facilitate and simplify the description in each subsequent experiment, we abbreviate these datasets according to their place names as "*SanFrancisco,*" "*Flevoland,*" "*Oberpfaffenhofen,*" and "*HaiNan,*" and they are all preprocessed by the Lee filter on the "PolSARpro" platform. To eliminate the influence of imbalanced datasets, the number of samples for each class is set to the same value.

1) *SanFrancisco:* This dataset was obtained by NASA/JPL AIRSAR L-band radar in July 1985 over San Francisco Bay, which is 900×1024 in size, as shown in Fig. 5(a). After removing the background in the corresponding color map, there are five different terrain types, i.e., urban, ocean, mountain, vegetation, and bare soil, which include total 802 302 pixels.
2) *Flevoland:* This dataset with the size of 920×1024 was acquired by an airborne system and in June 1991. After removing the background in the ground truth map, there are a total of 135 263 pixels for 14 different terrain types, including potato, beans, fruit, peas, wheat, oats, rapeseed, onions, barley, grass, flax, maize, beet, and lucerne.
3) *Oberpfaffenhofen:* This is an L-band fully PolSAR image with the size of 1300×1200, which was obtained by the E-SAR platform. After removing the white background pixels, this dataset includes 5 terrain types with total 1 311 618 pixels.
4) *HaiNan:* This is an L-band fully PolSAR image with the size of 1540×1250, which was obtained by the airborne platform and was available in 2024. After removing the white background pixels, this dataset includes seven terrain types with total 1 569 911 pixels.

In all subsequent experiments, the hyperparameter $\delta_l$ is set to $\log(0.5/l+1)$, and this configuration ensures that the decay of the weight matrix increases adaptively as more layers are stacked. The number of latent units in each layer is fixed as 256, and dropout rate is set to 0.5. The variances of weight and bias are $\sigma_b = 0$, and $\sigma_w = 1$, respectively. The noise variance $\epsilon$ of training samples is determined as 50 values by a grid search over numerical interval [1e-8, 1e-2]. For the concentration parameter $\eta_p$ in multiscale strategy, since there is significant

TABLE I
PARAMETER ANALYSIS OF SCALE $p$ ON THE *SANFRANCISCO* DATASET

| Scale $p$ | 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|---|
| Bare soil | 94.13 | 88.73 | 91.45 | 83.10 | 70.79 |
| Mountain | 98.75 | 98.93 | 99.22 | 99.64 | 95.09 |
| Ocean | 94.27 | 95.87 | 93.86 | 91.01 | 87.12 |
| Urban | 95.74 | 96.53 | 98.28 | 92.00 | 88.24 |
| Vegetation | 90.93 | 93.89 | 88.16 | 91.91 | 27.65 |
| OA | 95.03 | **96.14** | 95.75 | 92.04 | 74.94 |
| $\kappa$ | 92.37 | **94.04** | 93.43 | 87.96 | 83.99 |
| Time(s) Graph+Inference | **117.33** | 132.67 | 144.02 | 207.63 | 1971.84 |
| $p$-hop neighbor | 71982 | 213832 | 698476 | 2410964 | 8342312 |

TABLE II
PARAMETER ANALYSIS OF ROTATION ANGLE $\alpha$ ON *SANFRANCISCO* DATASET

| | Various $\alpha$ | | | | | |
|---|---|---|---|---|---|---|
| | $-\frac{1}{2}\pi$ | $-\frac{1}{4}\pi$ | 0 | $\frac{1}{4}\pi$ | $\frac{1}{2}\pi$ | learned $\alpha$ (Proposed) |
| OA | 95.90 | 95.87 | 94.83 | 95.35 | 95.46 | **96.14** |
| $\kappa$ | 93.70 | 93.62 | 92.08 | 92.86 | 93.04 | **94.04** |

| | Various $\Delta_i$ | | | |
|---|---|---|---|---|
| | $\Delta_1 = 0$ (Proposed) | single $\Delta$ $\Delta_1{=}1/6\pi$ | single $\Delta$ $\Delta_1{=}1/4\pi$ | $\Delta_1{=}1/6\pi$, $\Delta_2{=}1/4\pi$ |
| OA | **96.14** | 95.90 | 95.64 | 95.28 |
| $\kappa$ | **94.04** | 93.66 | 93.29 | 92.73 |

spatial heterogeneity of terrain targets across diverse scenes, which makes it challenging to determine the relative importance of the information at each scale. Hence, $\eta_p$ is determined as $1/p$ to use the averaging strategy for multiscale information fusion, thus ensuring the model generalizability. In particular, considering too much superpixels will significantly increase the complexity of network and the scale of adjacency matrix, while a few superpixels cannot guarantee the randomness of training process. Thus, following [37], the number of clustering centers, compact factor and iteration number in SLIC algorithm are chosen to 1e4,10, and 20 throughout experiments. Meanwhile, three evaluation indexes [i.e., per-class accuracy, overall accuracy (OA), and kappa coefficient ($\kappa$)] are leveraged to evaluate and analyze these classification results, where higher values indicate better classification performance. Particularly, to ensure the robustness of the results, every experiment is repeated five times, and the average value is taken as the final result. Experimental environment is Windows 10 system and Pytorch 1.11 version with CUDA 11.3, and python 3.8.

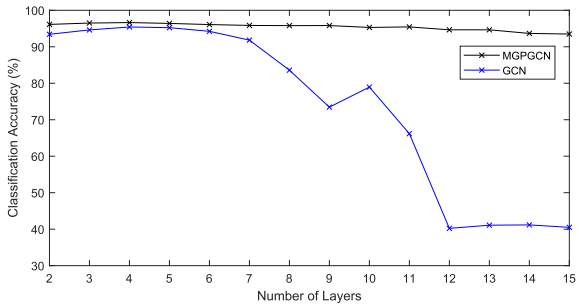
### B. Neighborhood Scale and Rotation Angle

In this section, some experiments are conducted on the *SanFrancisco* dataset to analyze the influence of the hyperparameters $p$ and rotation angle $\alpha$ in the proposed MGPGCN, and the training set is selected as 100 samples per-class.

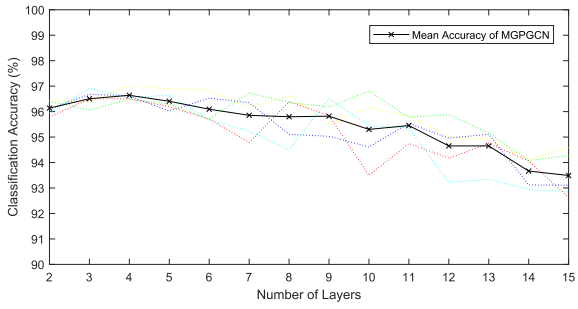
Specifically, various values for neighborhood scale $p$ are selected to be $\{1, 2, 3, 4, 5\}$, while the group number of enhanced features and rotation angle are fixed, and the corresponding results are presented in Table I, where the maximum values of OA and kappa coefficients are highlighted in bold for emphasis. When select a small value of neighborhood scale as $p = 1$, there are only 71 982 1-hop neighbors and the multiscale information is not considered, such that the learning time (i.e., the sum of constructing graph and inference process) only takes 117.33 s. However, although a small scale neighborhood can obtain a fast inference process, small receptive field leads to the inadequate learning of contextual information, thus obtaining an unsatisfied classification performance (i.e., 95.03% on OA). As the neighborhood scale increases (such as $p = 2$ or $p = 3$), more neighborhood information is incorporated in the graph adjacency matrix, and then is transmitted into the GP kernel. Accordingly, more spatial context information over constructed graph is considered and the classification accuracy can be improved. Compared with $p = 2$ and $p = 3$, it can be found that setting $p = 2$ has 213 832 $p$-hop neighbors (which is 1/3.26 than that of $p = 3$) and takes less time for learning, which obtain the better performance. In contrast, if a relatively large value of $p$ is determined as 5, too many $p$-hop neighbors are considered (i.e., 8 342 312). In such case, the learning time significantly increases to 1971.84s, and accordingly, due to focusing on too much neighborhood information, the resulting label is too smooth, thus yielding a very low classification accuracy. Hence, to the *SanFrancisco* dataset, the best value of $p$ is 2, which is an appropriate value to balance the learning of local correlation and the complexity.

Subsequently, the classification results involved in various PR angles and enhanced features are presented in Table II with setting neighborhood scale $p = 2$. By fixing each rotation angle $\alpha_i$ as $-\frac{1}{2}\pi, -\frac{1}{4}\pi, 0, \frac{1}{4}\pi$, and $\frac{1}{2}\pi$, respectively, the corresponding classification accuracies are given in the first four rows of Table II. It can be observed that choosing different rotation angles, the classification performance of the proposed MGPGCN varies slightly. However, the proposed method with the learned angles can gain the largest OA and $\kappa$ values. Meanwhile, when select angle $\alpha_i = 0$, there is a significant decrease in OA and $\kappa$ values, indicating that the rotation-based feature enhancement strategy could effectively learn more representative scattering information from the PR field. Furthermore, we utilize two different $\Delta_i$ (i.e., $\frac{1}{6}\pi$ and $\frac{1}{4}\pi$) to show the influence of enhanced features. When set $\Delta_1 = \frac{1}{6}\pi$ or $\Delta_1 = \frac{1}{4}\pi$ to extend the length scale of enhanced features as 16, the classification accuracy is lower than that obtained by a single group of rotation angles (i.e., $\Delta_i = 0$), which indicates that two groups of rotation angles may extract a few redundant features in some extent. In particular, when utilize more groups of rotation angles (i.e., both utilizing $\frac{1}{6}\pi$ and $\frac{1}{4}\pi$), model will be limited by the step length $\Delta$ to determine the inaccurate rotation angles. Thus, to the *SanFrancisco* dataset, the best value of $\Delta$ is 0.

Moreover, Fig. 6(a) shows the average classification performances of the proposed method and original GCN under different depthes $L$ with five random runs, respectively. The specific classification performance of our MGPGCN is exhibited in Fig. 6(b), where each dotted line represents accuracy of single run and the solid line represents average accuracy. According to Fig. 6(a), when $L$ is larger than 5, the classification accuracy of GCN begins to decline, and when $L$ is larger than 10, the




(a)




(b)

Fig. 6. Performance influence under different depth $L$ on the *SanFrancisco* dataset. (a) Average classification performance of MGPGCN and GCN with different depths. (b) Specific classification performance of our MGPGCN in five random tests (each random test is shown by dotted lines with different color, and their average accuracy is shown as the solid black line).

accuracy of GCN deteriorates significantly. The main reason is that the GCN utilizes the conventional convolutional operation in each layer (where a certain node may be calculated in several subgraphs), and then the network is easily prone to oversmoothing. On the contrary, since the convolutional operation is transformed to the estimation of the kernel in our method, the oversmoothing problem can be effectively addressed. In such case, the proposed MGPGCN can remain stable performance for depth $L$ as large as 10, and exhibit slight decline when $10 < L < 15$. Hence, introducing the GP technique into GCN framework can effectively address the over-smoothness problem.

### C. Analysis of Classification Performance

In this section, four datasets [see Fig. 5] including various complex scenes are considered to validate the effectiveness of the proposed MGPGCN. For accurately evaluating the effectiveness of proposed method, 7 comparative algorithms are selected, i.e., SVM [57], CV-CNN [58], GCN [30], GCNII [48], MDGCN [51], GPGCN, and GPGCNII [44], where SVM (based on H/A-Alpha decomposition), CV-CNN and GCN are the commonly-used baselines, GCNII is the backbone of the proposed method, MDGCN designs the multiscale theory, and both GPGCN and GPGCNII utilize GPs to model the GCN layers. In particular, we also use two ablation models, i.e., the proposed MGPGCN without **p**olarimetric **r**otation-based feature **e**nhancement strategy (denoted as MGPGCN w/o **PRE**) and the proposed MGPGCN without **m**ulti-**s**cale GP **k**ernel (denoted as MGPGCN w/o **MSK**) to validate the effectiveness of these two strategies on all datasets. Considering that these GCN-based methods are all superpixel-level and semisupervised methods, their training rates are only selected as 100, 20, 220, and 80 per class on the *SanFrancisco*, *Flevoland*, *Oberpfaffenhofen* and *HaiNan* datasets, respectively.

### D. Classification on SanFrancisco Dataset

Fig. 7 exhibits the classification experiments on the *SanFrancisco* dataset. According to Fig. 7(a)–(c), we can observe the *SanFrancisco* dataset has five land covers: bare soil, mountain, ocean, urban, and vegetation (corresponding to 13 701, 62 731, 329 566, 342 795, and 53 509 samples, respectively). By observing the Pauli RGB image and its corresponding ground truth map, it can be found that different from the optical image, the same terrain types in the PolSAR image may show very different features, i.e., "variant features." For example, in the east side of Fig. 7(a), there is a triangle area with orientated buildings marked by the black circle, which has very different scattering mechanism from other building areas and is generally identified as the vegetation area. Hence, classifying these "variant features" could indicate how well a model learns polarimetric characteristics sufficiently.

For SVM in Fig. 7(d), its input features are determined by the H/A-Alpha decomposition technique. Since the structure of SVM is shallow, the model cannot extract feature as effectively as deep structures, thus leading to numerous misclassifications among each classes. For the CV-CNN method in Fig. 7(e), many urban pixels in the black circles are misidentified as the vegetation pixels, implying that it lost many polarimetric characteristics during classification process. Besides, due to not introducing superpixel strategy, miscellaneous pixels are identified, such as many ocean pixels in the lower left corner are misclassified as bare soil pixels. In Fig. 7(f), although GCN utilizes the superpixel strategy to consider local information and performs well in the homogeneous areas, i.e., ocean or mountain areas. The urban pixels marked in the lower circle are also recognized as bare soil, indicate that a lot of polarimetric information is still ignored. By designing the residual connection, GCNII could acquire more correct classification results of bare soil and urban categories in Fig. 7(g). However, numerous orientated buildings in the marked triangle area are misclassified either, and the variant features cannot be correctly processed by GCNII.

Particularly, according to Fig. 7(f) and (g) and Table III, when introducing GP to model the network, the classification accuracies of GPGCN and GPGCNII are all lower than GCN and GCNII. The main reason is that their input features have low length scales and large spatial scales. In such cases, GP statistical framework are easily effected by the speckle noise, and cannot capture the latent statistical properties of data. Hence, GPGCN and GPGCNII achieve the lower performance on OA and $\kappa$ values. As for the proposed MGPGCN in Fig. 7(m), since the PR information is incorporated in the enhanced features, those variant targets with similar scattering represents can be distinguished correctly at different rotation angles. Thus, the areas marked by black circles are all identified as the urban areas,

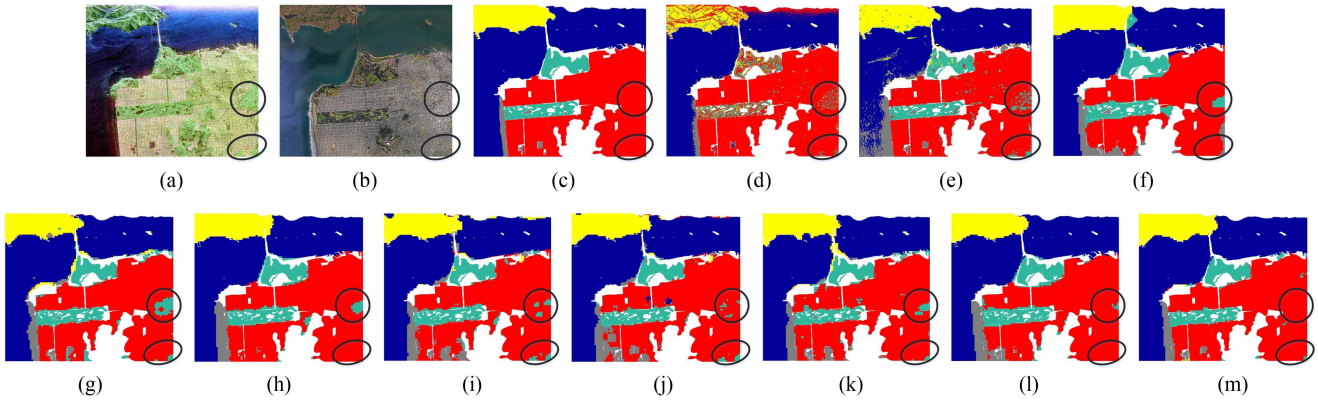



Fig. 7. *SanFrancisco* dataset and its classification maps by different methods. (a) Pauli RGB Image. (b) Optical image from GoogleEarth. (c) Ground-truth map. The resulting classification maps by employing (d) SVM. (e) CV-CNN. (f) GCN. (g) GCNII. (h) MDGCN. (i) GPGCN. (j) GPGCNII. (k) MGPGCN w/o PRE. (l) MGPGCN w/o MSK. (m) Proposed MGPGCN.

TABLE III
ACCURACY OF DIFFERENT METHODS ON *SANFRANCISCO* DATASET

| Class | Producer's Accuracy (%) | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | SVM | CV-CNN | GCN | GCNII | MDGCN | GPGCN | GPGCNII | MGPGCN w/o PRE | MGPGCN w/o MSK | MGPGCN |
| Bare soil | 64.04 | 94.12 | 85.95 | 92.57 | 93.19 | 88.95 | 93.04 | 89.10 | 94.13 | 88.73 |
| Mountain | 62.23 | 97.47 | 97.33 | 95.57 | 97.96 | 96.84 | 99.28 | 97.08 | 98.75 | 98.93 |
| Ocean | 93.95 | 88.19 | 95.56 | 94.43 | 94.69 | 93.74 | 92.88 | 96.27 | 94.27 | 95.87 |
| Urban | 97.06 | 95.29 | 90.47 | 94.10 | 95.18 | 90.65 | 93.35 | 91.64 | 95.74 | 96.53 |
| Vegetation | 37.36 | 91.65 | 96.44 | 91.81 | 92.31 | 92.25 | 93.84 | 95.65 | 90.93 | 93.89 |
| OA | 88.51 | 92.27 | 93.43 | 94.17 | 94.97 | 92.49 | 93.66 | 94.19 | 95.03 | **96.14** |
| $\kappa$ | 86.50 | 88.25 | 89.99 | 91.08 | 92.28 | 88.63 | 90.35 | 91.13 | 92.37 | **94.04** |

verifying the effectiveness of the proposed feature enhancement strategy and indicating this strategy being suitable for the variant problem. Meanwhile, due to leveraging the multiscale strategy to consider the local context information, the proposed method can obtain the smoother classification maps than other methods, especially in the homogeneous mountain and vegetation areas, which also verifies the effectiveness of the multiscale kernel.

Furthermore, for two ablation models (i.e., MGPGCN w/o PRE and MGPGCN w/o MSK), the corresponding classification maps are shown in Fig. 7(k) and (l). Obviously, after omitting the enhanced features, many urban pixels in the black circles are misclassified as vegetation class by MGPGCN w/o PRE. Meanwhile, the smoother result obtained by MGPGCN w/o PRE makes the higher accuracies in homogeneous areas (e.g., Ocean and Vegetation areas). On the contrary, without imposing the multiscale kernel, MGPGCN w/o MSK can effectively identify many variant urban pixels in the black circles, and obtain the better results in the heterogeneous areas (e.g., Bare soil area). However, MGPGCN w/o MSK cannot keep performance on homogeneous areas in view of neglecting the local correlations. Compared with Fig. 7(k) and (l), the proposed MGPGCN in Fig. 7(m) can obtain the good balance between global smoothness and local saliency by utilizing both the multiscale and feature enhancement strategies, thus achieving the better performance than two ablation models. The highest values of $\kappa$ coefficient and OA at the last row of Table III also validate the validity of our MGPGCN.

Table IV exhibits the running time of different methods on the *SanFrancisco* dataset. Compared with the pixel-level CV-CNN, the proposed MGPGCN takes the lower elapsed time and shows the higher efficiency. This is because the superpixel technique is considered in the proposed method to reduce the computation overhead. Meanwhile, since the learning of traditional graph convolutional operation is converted to the estimating of the feature distribution, the proposed MGPGCN as well as other two statistical modeling-based methods (i.e., GPGCN and GPGCNII) can determine the classification results with an efficient process. Compared with the GPGCN and GPGCNII, though MGPGCN spends 82.70s, the better classification results can be obtained by MGPGCN due to more context information and PR information being explored. In summary, by trading a little more time, MGPGCN can obtain higher performance with the proposed multiscale and rotation strategies.

### *E. Classification on* Flevoland, Oberpfaffenhofen, *and* HaiNan *Datasets*

In this section, we discuss the classification experiments on *Flevoland*, *Oberpfaffenhofen,* and *HaiNan* datasets simultaneously. Fig. 5(b)–(d) presents the Pauli RGB images and the ground truth maps of the *Flevoland*, *Oberpfaffenhofen,* and

TABLE IV
RUNNING TIME OF DIFFERENT METHODS ON EACH DATASET

| Time (s) | 1 epoch | 100 epoches | | | inference | | |
|---|---|---|---|---|---|---|---|
| | CV-CNN | GCN | GCNII | MDGCN | GPGCN | GPGCNII | MGPGCN |
| *SanFrancisco_s* | 125.52 | 117.59 | 118.86 | 121.14 | 58.04 | 58.21 | 82.70 |
| *Flevoland_s* | 131.04 | 199.61 | 194.80 | 202.09 | 58.33 | 58.97 | 82.20 |
| *Oberpfaffenhofen* | 152.11 | 335.50 | 335.87 | 339.24 | 84.63 | 84.23 | 116.68 |
| *HaiNan_s* | 145.32 | 224.42 | 225.20 | 233.35 | 95.44 | 96.07 | 127.13 |

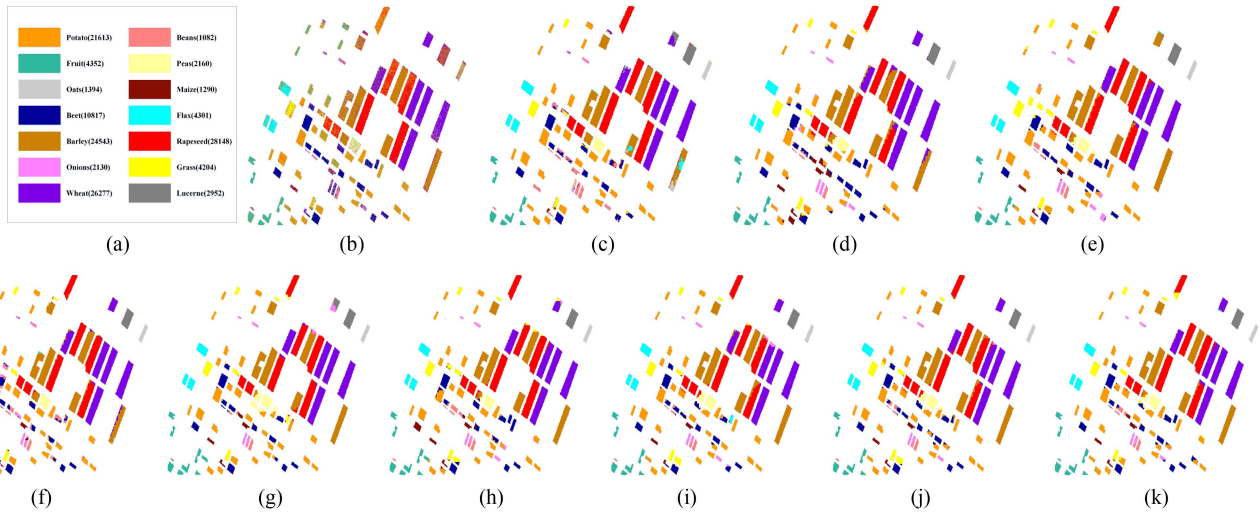



Fig. 8. *Flevoland* dataset and its classification results by employing different methods. (a) Color code map. (b) SVM. (c) CV-CNN. (d) GCN. (e) GCNII. (f) MDGCN. (g) GPGCN. (h) GPGCNII. (i) MGPGCN w/o PRE. (j) MGPGCN w/o MSK. (k) Proposed MGPGCN.

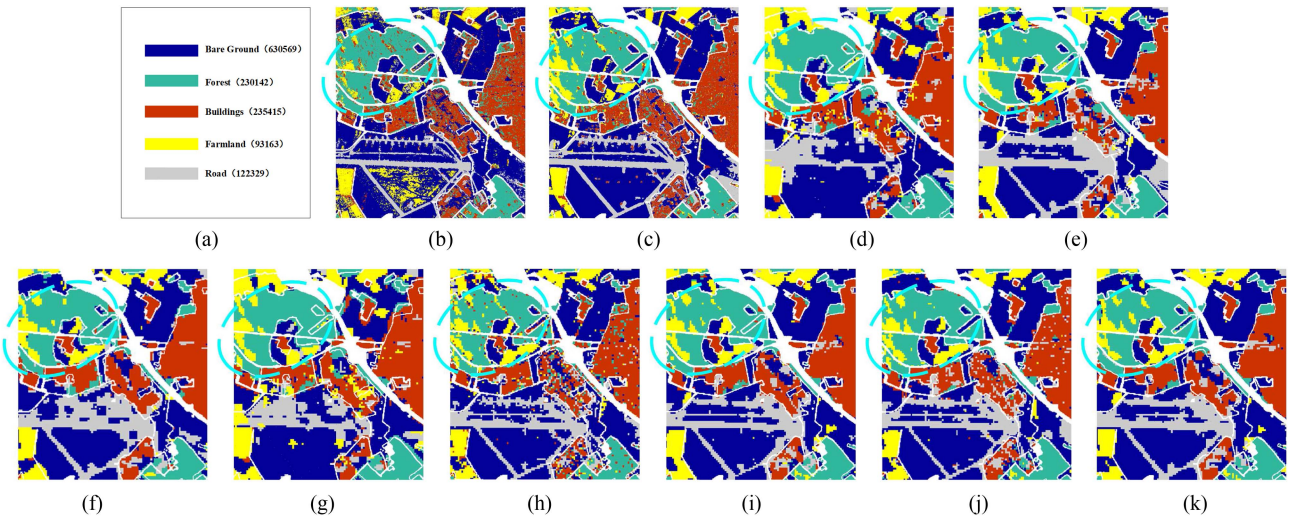



Fig. 9. *Oberpfaffenhofen* dataset and its classification results by employing different methods. (a) Color code map. (b) SVM. (c) CV-CNN. (d) GCN. (e) GCNII. (f) MDGCN. (g) GPGCN. (h) GPGCNII. (i) MGPGCN w/o PRE. (j) MGPGCN w/o MSK. (k) Proposed MGPGCN.

*HaiNan* datasets, respectively, and their corresponding color maps and the number of each terrain types. The classification results generated by the proposed MGPGCN as well as other comparative methods are shown in Figs. 8–10. For the quantitative evaluation, the performance analyses are all summarized in the Table V– VII, including the OA, $\kappa$ coefficient, and producer's accuracy.

Specifically, for the *Flevoland* dataset, due to the fact that the same crop categories are commonly found among a local area, the spatial information is very significant to encourage a certain pixel to learn the same labels with its adjacent pixels. Thus, by observing classification results of SVM, CV-CNN, GCN, and GCNII in Fig. 8(c)–(f) and their performance indicators in Table V, we can find that due to not introducing superpixel

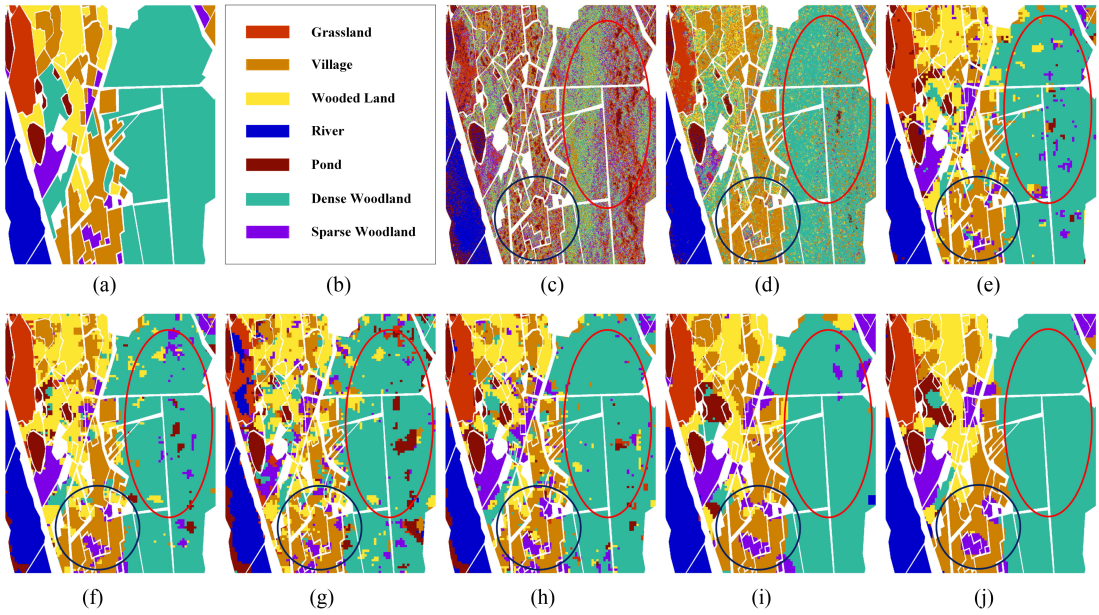



Fig. 10. *HaiNan* dataset and its classification results by employing different methods. (a) Ground truth map. (b) Color code map. (c) SVM. (d) CV-CNN. (e) GCN. (f) GCNII. (g) GPGCN. (h) GPGCNII. (i) MDGCN. (j) MGPGCN.

TABLE V
ACCURACY OF DIFFERENT METHODS ON *FLEVOLAND* DATASET

| Class | Producer's Accuracy (%) | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | SVM | CV-CNN | GCN | GCNII | MDGCN | GPGCN | GPGCNII | MGPGCN w/o PRE | MGPGCN w/o MSK | MGPGCN |
| Potato | 78.71 | 98.62 | 85.39 | 90.20 | 97.60 | 92.51 | 86.72 | 98.42 | 89.97 | 93.80 |
| Fruit | 87.54 | 98.69 | 99.53 | 99.60 | 99.86 | 99.79 | 99.56 | 99.51 | 99.31 | 99.74 |
| Oats | 36.65 | 99.14 | 99.34 | 99.41 | 100.00 | 100.00 | 99.71 | 100.00 | 100.00 | 100.00 |
| Beet | 87.21 | 75.13 | 84.03 | 80.93 | 67.86 | 79.21 | 72.80 | 75.75 | 89.16 | 85.58 |
| Barley | 89.31 | 89.52 | 91.46 | 96.61 | 93.41 | 92.29 | 95.36 | 93.90 | 94.01 | 96.20 |
| Onions | 38.13 | 44.74 | 88.33 | 86.73 | 70.66 | 78.15 | 86.87 | 81.42 | 85.73 | 83.83 |
| Wheat | 91.26 | 94.52 | 97.48 | 90.75 | 98.16 | 93.06 | 95.53 | 92.50 | 96.80 | 97.63 |
| Beans | 44.97 | 91.95 | 80.88 | 96.98 | 82.67 | 81.26 | 99.44 | 83.33 | 97.08 | 85.40 |
| Peas | 78.40 | 94.49 | 94.06 | 99.43 | 99.72 | 99.16 | 99.85 | 99.25 | 99.02 | 99.86 |
| Maize | 43.93 | 50.76 | 99.05 | 97.32 | 88.26 | 97.32 | 96.69 | 93.38 | 97.32 | 99.44 |
| Flax | 83.49 | 99.44 | 99.67 | 99.23 | 99.29 | 99.34 | 99.93 | 100.00 | 100.00 | 100.00 |
| Rapeseed | 73.57 | 95.98 | 91.07 | 93.87 | 94.84 | 96.70 | 98.87 | 95.48 | 93.64 | 93.28 |
| Grass | 63.61 | 81.71 | 99.21 | 99.16 | 95.53 | 99.25 | 98.81 | 98.01 | 99.50 | 99.16 |
| Lucerne | 36.28 | 91.87 | 98.29 | 100.00 | 98.84 | 99.59 | 100.00 | 99.04 | 99.93 | 100.00 |
| OA | 80.04 | 91.63 | 91.96 | 92.90 | 93.52 | 93.11 | 93.45 | 93.71 | 94.17 | **94.90** |
| $\kappa$ | 78.65 | 90.23 | 90.57 | 91.68 | 92.40 | 91.95 | 92.33 | 92.62 | 93.17 | **94.02** |

TABLE VI
ACCURACY OF DIFFERENT METHODS ON *OBERPFAFFENHOFEN* DATASET

| Class | Producer's Accuracy (%) | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | SVM | CV-CNN | GCN | GCNII | MDGCN | GPGCN | GPGCNII | MGPGCN w/o PRE | MGPGCN w/o MSK | MGPGCN |
| Bare ground | 80.71 | 80.60 | 66.36 | 75.83 | 79.81 | 63.38 | 70.51 | 72.75 | 76.07 | 77.22 |
| Forest | 83.95 | 81.05 | 86.67 | 88.58 | 82.11 | 87.99 | 91.81 | 91.91 | 89.17 | 88.10 |
| Buildings | 79.32 | 83.16 | 82.69 | 89.10 | 81.87 | 84.11 | 83.80 | 90.22 | 90.62 | 94.00 |
| Farmland | 61.93 | 75.93 | 76.76 | 90.99 | 86.69 | 79.28 | 90.55 | 90.10 | 91.02 | 92.51 |
| Road | 67.38 | 77.16 | 44.69 | 76.08 | 76.42 | 43.86 | 82.16 | 80.82 | 79.80 | 79.05 |
| OA | 78.45 | 80.49 | 71.57 | 81.13 | 80.76 | 70.73 | 79.14 | 81.23 | 82.39 | **83.39** |
| $\kappa$ | 73.73 | 72.63 | 61.11 | 73.90 | 73.18 | 60.38 | 71.72 | 74.45 | 75.77 | **77.16** |

TABLE VII
ACCURACY OF DIFFERENT METHODS ON *HAINAN* DATASET

| Class | Producer's Accuracy (%) | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | CV-CNN | GCN | GCNII | MDGCN | GPGCN | GPGCNII | GCN +PR | GCNII +PR | MDGPGCN +PR | GPGCN +PR | GPGCNII +PR | MGPGCN |
| Village | 75.27 | 70.49 | 70.61 | 76.52 | 59.72 | 63.30 | 75.07 | 75.35 | 71.34 | 57.96 | 64.18 | 76.48 |
| River | 97.47 | 96.18 | 94.00 | 97.60 | 86.44 | 89.45 | 97.91 | 95.67 | 94.66 | 86.11 | 89.96 | 98.62 |
| Pond | 72.92 | 89.79 | 92.56 | 95.80 | 77.68 | 94.09 | 93.09 | 93.18 | 69.01 | 89.19 | 93.89 | 94.13 |
| Sparse Woodland | 22.79 | 78.64 | 89.23 | 92.53 | 54.39 | 85.27 | 76.64 | 84.71 | 79.28 | 74.69 | 88.40 | 93.68 |
| Woodland | 37.70 | 61.99 | 69.86 | 80.22 | 63.98 | 75.17 | 66.79 | 70.90 | 65.34 | 68.87 | 77.07 | 82.45 |
| Dense Woodland | 75.69 | 84.61 | 88.47 | 90.50 | 75.50 | 85.98 | 84.63 | 88.35 | 89.46 | 79.28 | 87.34 | 90.70 |
| Grassland | 68.73 | 92.21 | 92.07 | 97.14 | 63.64 | 92.51 | 91.03 | 94.18 | 97.04 | 86.60 | 92.85 | 97.38 |
| OA | 71.43 | 81.28 | 84.44 | 83.53 | 71.31 | 81.93 | 82.77 | 85.35 | 84.00 | 75.70 | 83.16 | **88.71** |
| $\kappa$ | 57.44 | 72.93 | 77.36 | 76.04 | 59.71 | 73.80 | 75.12 | 78.61 | 76.42 | 65.56 | 75.53 | **83.49** |

technique and ignoring the multi-scale information, the classification results of SVM and CV-CNN cannot provide the smooth label, and many misclassifications are generated by GCN and GCNII among potato and wheat areas. Although MDGCN incorporates the multiscale information, its performance shows low accuracy to recognize the pixels in a local patch, e.g., the onions class. Moreover, GPGCN and GPGCNII obtain miscellaneous pixels in the beet areas in view of the insufficient learning of polarimetric information. By compared with the proposed model and its ablation models, the classification results obtained by the proposed MGPGCN verify that our superpixel-level PR feature enhancement strategy can effectively assist the GP statistical-modeling techniques to learn the accurate data properties and eliminate the effect of noise.

According to Fig. 5(c), the *Oberpfaffenhofen* dataset includes five classes: bare ground, forest, buildings, farmland, and road (corresponding to 630 569, 230 142, 235 415, 93 163, and 122 329 samples, respectively). Obviously, pixel-based classification methods may offer greater advantages for this dataset due to the coexistence of multiple land-cover categories within localized small pixel blocks (e.g., road areas). For example, in Fig. 9(b) and (c), while SVM and CV-CNN produce significant misclassifications in homogeneous regions, they preserve clearer edge structures than superpixel-based methods. In contrast, the conventional SLIC superpixel segmentation struggles to define precise boundaries in central areas containing diverse targets within local blocks, leading to partial spatial information loss in superpixel-based methods [i.e., Fig. 9(d)–(k)] and consequent misclassifications in heterogeneous zones. Hence, in the future work, we consider to improve the segmentation results of superpixels by incorporating spatial information or utilizing pixel-level information to assist superpixel segmentation [5], thus determining more accurate superpixel edges to address this issue. Compared with GCN, GCNII, MDGCN, GPGCN, and GPGCNII in Fig. 9(d)–(h), the proposed MGPGCN can obtain the better performance by employing the PR strategy, especially for the areas marked by the blue circle. Meanwhile, by applying multiscale strategy, MGPGCN can acquire more smoother labels in the homogeneous areas, such as the lower right corner of the image. The highest OA and $\kappa$ values in Table VI further confirm our method's effectiveness.

As for the *HaiNan* dataset [59], the corresponding classification experiments are exhibited in the Fig. 10. According to Fig. 10(a), we can find that there are seven land covers in the whole image: village, river, pond, sparse woodland, woodland, dense woodland, and grassland (corresponding to 261 358, 131 327, 38 020, 60 897, 153 711, 847 909, 76 689 samples, respectively). Since the scene is very complex, we set 12% samples per class of supervised SVM and CV-CNN for training, and 80 samples per class of other superpixel-level semi-supervised methods for training. By observing Fig. 10(d), though the CV-CNN's classification result maintains the main structure due to the higher train ratio, there are still too many misclassifications in each terrain type, resulting in a miscellaneous labeling map. Meanwhile, for the GCN, GCNII, GPGCN, and GPGCNII in Fig. 10(e)–(h), many homogeneous dense woodland pixels among the red circle are incorrectly recognized as miscellaneous pixels, and the heterogeneous pixels in the black circle or the central area are also difficult to identify. The main reason is that the pixel patches in this dataset are very small and most of them have irregular shapes. Hence, these four methods without considering the multiscale features cannot fully leverage the contextual information from neighboring nodes. On the contrary, with the multiscale strategy, the classification results obtained by the MDGCN and proposed MGPGCN all exhibit the smoother labels. Compared with the result of MDGCN, MGPGCN integrates more detailed information and achieves the better performance the intersection of different land covers in view of employing the PR strategy to enhance the discriminant of features. In particular, Table VIII exhibits the determined rotation angles, it can be observed that each learned $\alpha_i$ is not equal to 0, which means all enhanced features have been effectively rotated. Fig. 11 shows the visual feature maps of each PR-based channel. According to these maps, the discriminant information can be explored from various angles, and thereby the original input of network can be enhanced from diverse scattering model perspectives.

TABLE VIII
FINAL ROTATION ANGLES OF *HAINAN* DATASET

| Learned angles | $\alpha_1$ | $\alpha_2$ | $\alpha_3$ | $\alpha_4$ | $\alpha_5$ |
|---|---|---|---|---|---|
| $\times\pi$ | 0.8308 | 0.9765 | 0.0555 | 0.1440 | 0.7369 |

Furthermore, the PR-based enhancement strategy is also added in the GCN, GCNII, MDGCN, GPGCN, and GPGCNII

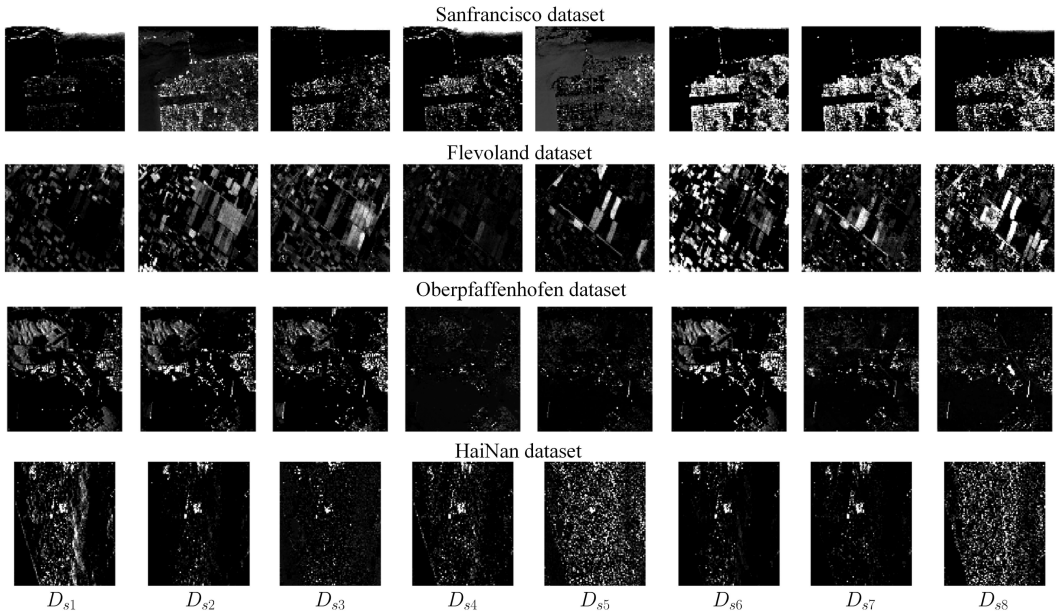



Fig. 11. Visual maps of rotation features on different datasets.

for verify its effectiveness, where the rotation angles are fixed as -$\pi$/4, -$\pi$/6, $\pi$/6, and $\pi$/4 for selecting the best result for comparison. Their corresponding results are also given in the Table VII. According to the table, we can find that the classification performance of each method is improved by leveraging PR technique, thus verifying the effectiveness of our feature enhancement strategy. Meanwhile, the highest classification accuracy of our method also verifies the superiority of the proposed MGPGCN.

In summary, the above experiments demonstrate that the proposed MGPGCN effectively explores the rotational invariance and extracts more discriminative information through the PR-based feature enhancement strategy, while reducing interference from speckle noise. The multi-scale strategy further extracts neighborhood information, suppressing anomalies within local regions to achieve smooth and reliable classification results. In addition, the uncertainty quantification inherent to the GP enhances the model's robustness. Hence, the proposed model can have a certain degree of robustness and generalization for different PolSAR scenes.

## V. Conclusion

In this work, a novel MGPGCN has been proposed for PolSAR image semi-supervised classification. First, considering that deep probability networks can effectively learn the latent feature distribution, GP is introduced into GCN for addressing the oversmoothing problem, improving the learning capability and capturing uncertainty. Instead of directly modeling GCN by GP, we model the GCNII architecture to reserve more polarization information of the original PolSAR image and derive the corresponding multiscale GP kernel function. In such case, the representation capability can be further improved, and the contextual information can be fully learned. In addition, the superpixel technique is utilized to reduce the computation complexity of graph and GP kernel on spatial scale perspective. Meanwhile, the PR-based feature enhancement strategy is also designed to enhance the data length scale, by which the input feature is more discriminative and the noise effect is further mitigated. Finally, according to the learned kernel, the accurate classification results are obtained by the joint probability density. The experimental results on four real-world PolSAR datasets with various scenes show that our MGPGCN has better performance than some widely-used GCN-based classification methods.

In the future, we will consider to construct the graph neural processes by extending GP to neural processes, such that the prior assumption of GP can be more relaxed and the computation overhead will be much lower. Meanwhile, the negative sample selection strategy will be introduced into the graph neural processes. In this case, those adjacent negative nodes that include abundant information can help the positive samples to enlarge the interclass distance, further explore the neighboring correlations, and improve the learning ability. Furthermore, the superpixel segmentation technique will also be improved by polarimetric scattering information for obtaining the more accurate edges.

## Acknowledgment

The authors would like to thank the European Space Agency, Jet Propulsion Laboratory, United States, and Aerospace Information Research Institute, Chinese Academy of Sciences for providing the PolSAR images. The authors would also like to thank the Associate Editor and the Anonymous Reviewers for their valuable comments and suggestions, which are greatly helpful to improve the quality and presentation of this paper.

## REFERENCES


[1] T. Li, Z. Liu, R. Xie, L. Ran, and J. Wang, "Ship detection for polarimetric SAR images based on $\mathcal{G}_p^0$ mixture model," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 12, no. 6, pp. 1812–1824, Jun. 2019.

[2] R. Yang, X. Xu, Z. Xu, H. Dong, R. Gui, and F. Pu, "Dynamic fractal texture analysis for PolSAR land cover classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 8, pp. 5991–6002, Aug. 2019.

[3] 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.

[4] Q. Chen, G. Kuang, J. Li, L. Sui, and D. Li, "Unsupervised land cover/land use classification using PolSAR imagery based on scattering similarity," *IEEE Trans. Geosci. Remote Sens.*, vol. 51, no. 3, pp. 1817–1825, Mar. 2013.

[5] H. Jin et al., "Scale-aware triple semantic deep network for polarimetric SAR image classification," *Appl. Soft. Comput.*, vol. 177, Jun. 2025, Art. no. 113232.

[6] Y. Yamaguchi, "Disaster monitoring by fully polarimetric SAR data acquired with ALOS-PALSAR," *Proc. IEEE*, vol. 100, no. 10, pp. 2851–2860, Oct. 2012.

[7] D. Mandal et al., "Dual polarimetric radar vegetation index for crop growth monitoring using sentinel-1 SAR data," *Remote Sens. Environ.*, vol. 247, no. 15, pp. 111954–1-17, Sep. 2020.

[8] H. Parikh, S. Patel, and V. Patel, "Classification of SAR and PolSAR images using deep learning: A review," *Int. J. Image Data Fusion*, vol. 11, no. 1, pp. 1–32, Jan. 2020.

[9] B. Hou, C. Yang, B. Ren, and L. Jiao, "Decomposition-feature-iterative-clustering-based superpixel segmentation for PolSAR image classification," *IEEE Geosci. Remote Sens. Lett.*, vol. 15, no. 8, pp. 1239–1243, Aug. 2018.

[10] D. Xiang, T. Tang, Y. Ban, Y. Su, and G. Kuang, "Unsupervised polarimetric SAR urban area classification based on model-based decomposition with cross scattering," *ISPRS J. Photogramm. Remote Sens.*, vol. 116, no. 1, pp. 86–100, Mar. 2016.

[11] R. Gui, X. Xu, R. Yang, L. Wang, and F. Pu, "Statistical scattering component-based subspace alignment for unsupervised cross-domain PolSAR image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 7, pp. 5449–5463, Jul. 2021.

[12] B. Zou, D. Lu, L. Zhang, and W. M. Moon, "Eigen-decomposition-based four-component decomposition for PolSAR data," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 9, no. 3, pp. 1286–1296, Mar. 2016.

[13] J. E. Ferguson and G. E. Gunn, "Polarimetric decomposition of microwave-band freshwater ice SAR data: Review, analysis, and future directions," *Remote Sens. Environ.*, vol. 280, pp. 113176–1-13, Aug. 2022.

[14] S.-W. Chen, Y.-Z. Li, X.-S. Wang, S.-P. Xiao, and M. Sato, "Modeling and interpretation of scattering mechanisms in polarimetric synthetic aperture radar: Advances and perspectives," *IEEE Signal Process. Mag.*, vol. 31, no. 4, pp. 79–89, Jul. 2014.

[15] M. Shokr and M. Dabboor, "Polarimetric SAR applications of sea ice: A review," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 16, pp. 6627–6641, 2023.

[16] C. Liu, W. Liao, H.-C. Li, K. Fu, and W. Philips, "Unsupervised classification of multilook polarimetric SAR data using spatially variant Wishart mixture model with double constraints," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 10, pp. 5600–5613, Oct. 2018.

[17] 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.

[18] 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.

[19] H. Shen, L. Lin, J. Li, Q. Yuan, and L. Zhao, "A residual convolutional neural network for polarimetric SAR image super-resolution," *ISPRS J. Photogramm. Remote Sens.*, vol. 161, pp. 90–108, Mar. 2020.

[20] J. Shi, S. Ji, H. Jin, J. Li, M. Gong, and W. Lin, "Content-adaptive multi-region deep network for polarimetric SAR image classification," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 35, no. 1, pp. 617–631, Jan. 2025.

[21] J. Shi, W. Wang, H. Jin, M. Nie, and S. Ji, "A lightweight Riemannian covariance matrix convolutional network for PolSAR image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 62, Jul. 2024, Art. no. 4411017.

[22] H. Bi, J. Sun, and Z. Xu, "A graph-based semisupervised deep learning model for PolSAR image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 4, pp. 2116–2132, Apr. 2019.

[23] X. Liu, L. Jiao, X. Tang, Q. Sun, and D. Zhang, "Polarimetric convolutional network for PolSAR image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 5, pp. 3040–3054, May 2019.

[24] P. Wang, Z. He, B. Huang, M. Dalla Mura, H. Leung, and J. Chanussot, "VOGTNet: Variational optimization-guided two-stage network for multispectral and panchromatic image fusion," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 36, no. 5, pp. 9268–9282, May 2025.

[25] M. Ding, X. Fu, T.-Z. Huang, J. Wang, and X.-L. Zhao, "Hyperspectral super-resolution via interpretable block-term tensor modeling," *SIAM J. Imag. Sci.*, vol. 15, no. 3, pp. 641–656, Apr. 2021.

[26] Y. He, B. Tu, P. Jiang, B. Liu, J. Li, and A. Plaza, "IGroupSS-mamba: Interval group spatial-spectral mamba for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 62, 2024, Art. no. 5538817.

[27] P. Wang, Y. Su, B. Huang, D. Zhu, and W. Liu, "Low-rank tensor completion pansharpening based on haze correction," *IEEE Trans. Geosci. Remote Sens.*, vol. 62, May 2024, Art. no. 5405720.

[28] Z. Wen, Q. Wu, Z. Liu, and Q. Pan, "Polar-spatial feature fusion learning with variational generative-discriminative network for PolSAR classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 11, pp. 8914–8927, Nov. 2019.

[29] J. Zhou et al., "Graph neural networks: A review of methods and applications," *AI Open*, vol. 1, pp. 57–81, Apr. 2020.

[30] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," in *Proc. Int. Conf. Learn. Representations*, Toulon, France, Apr. 2017, pp. 1–14.

[31] J. Chen, 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.

[32] 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.

[33] H. Liu, T. Zhu, F. Shang, Y. Liu, D. Lv, and S. Yang, "Deep fuzzy graph convolutional networks for PolSAR imagery pixelwise classification," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 14, pp. 504–514, 2021.

[34] J. Shi, T. He, S. Ji, M. Nie, and H. Jin, "CNN-improved superpixel-to-pixel fuzzy graph convolution network for PolSAR image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 61, 2023, Art. no. 4410118.

[35] J. Cheng, D. Xiang, Q. Yin, and F. Zhang, "A novel crop classification method based on the tensor-GCN for time-series PolSAR data," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, Oct. 2022, Art. no. 5237614.

[36] S. Yang, R. Li, Z. Li, H. Meng, Z. Feng, and G. He, "Meta-graph representation learning for PolSAR image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 62, 2024, Art. no. 5211012.

[37] 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.

[38] A. Radman, M. Mahdianpari, B. Brisco, B. Salehi, and F. Mohammadimanesh, "Dual-branch fusion of convolutional neural network and graph convolutional network for PolSAR image classification," *Remote Sens.*, vol. 15, no. 1, pp. rs15010075–1-19, Feb. 2023.

[39] L. V. Jospin, H. Laga, F. Boussaid, W. Buntine, and M. Bennamoun, "Hands-on Bayesian neural networks-a tutorial for deep learning users," *IEEE Comput. Intell. Mag.*, vol. 17, no. 2, pp. 29–48, May 2022,

[40] W. Xu, R. T. Q. Chen, X. Li, and D. Duvenaud, "Infinitely deep Bayesian neural networks with stochastic differential equations," in *Proc. Int. Con. Artif. Intell. Stat.*, Valencia, Spain, Mar. 2022, pp. 721–738.

[41] Y. Liu, S. Liu, Y. Wang, F. Lombardi, and J. Han, "A survey of stochastic computing neural networks for machine learning applications," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 32, no. 7, pp. 2809–2824, Jul. 2020.

[42] G. Yang, "Wide feedforward or recurrent neural networks of any architecture are gaussian processes," in *Proc. Adv. Neural Inf. Process. Syst.*, Vancouver, Canada, Dec. 2019, pp. 1–10.

[43] A. Garriga-Alonso, C. E. Rasmussen, and L. Aitchison, "Deep convolutional networks as shallow gaussian processes," in *Proc. Int. Conf. Learn. Representations*, New Orleans, LA, USA, May 2019, pp. 1–16.

[44] Z. Niu, M. Anitescu, and J. Chen, "Graph neural network-inspired kernels for Gaussian processes in semi-supervised learning," in *Proc. Int. Conf. Learn. Representations*, Kigali, Rwanda, May 2023, pp. 1–19.

[45] Q. Zhang, C. He, B. He, and M. Tong, "Learning scattering similarity and texture-based attention with convolutional neural networks for PolSAR image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 61, 2023, Art. no. 5207419.

[46] Y. Cui, F. Liu, L. Jiao, Y. Guo, X. Liang, and L. Li, "Polarimetric multipath convolutional neural network for PolSAR image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5207118.

[47] C. Tao, S. Chen, Y. Li, and S. Xiao, "PolSAR land cover classification based on roll-invariant and selected hidden polarimetric features in the rotation domain," *Remote Sens.*, vol. 9, no. 7, Jul. 2017, Art. no. 660.

[48] M. Chen, Z. Wei, Z. Huang, B. Ding, and Y. Li, "Simple and deep graph convolutional networks," in *Proc. Int. Conf. Mecha. Learn.*, Vienna, Austria, Jul. 2020, pp. 1725–1735.

[49] R. Novak, "Bayesian deep convolutional networks with many channels are Gaussian processes," in *Proc. Int. Conf. Learn. Representations*, New Orleans, LA, USA, May 2019, pp. 1–35.

[50] C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning*, Cambridge, MA, USA: MIT press, 2006.

[51] 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.

[52] C. Yang, B. Hou, B. Ren, Y. Hu, and L. Jiao, "CNN-based polarimetric decomposition feature selection for PolSAR image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 11, pp. 8796–8812, Jul. 2020.

[53] H. Dong, L. Zhang, D. Lu, and B. Zhou, "Attention-based polarimetric feature selection convolutional network for PolSAR image classification," *IEEE Geosci. Remote Sens. Lett.*, vol. 19, 2022, Art. no. 4001705.

[54] D. Li and Y. Zhang, "Random similarity between two mixed scatterers," *IEEE Geosci. Remote Sens. Lett.*, vol. 12, no. 12, pp. 2468–2472, Dec. 2015.

[55] J. Yang, Y.-N. Peng, and S.-M. Lin, "Similarity between two scattering matrices," *Electron. Lett.*, vol. 37, no. 3, pp. 193–194, Feb. 2001.

[56] Q. Chen, Y. Jiang, L. Zhao, and G. Kuang, "Polarimetric scattering similarity between a random scatterer and a canonical scatterer," *IEEE Geosci. Remote Sens. Lett.*, vol. 7, no. 4, pp. 866–869, Oct. 2010.

[57] C.-C. Chang and C.-J. Lin, "LIBSVM: A library for support vector machines," *ACM Trans. Intell. Syst. Technol.*, vol. 2, no. 3, Apr. 2011, Art. no. 1C27.

[58] 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.

[59] Y. Jin et al., "MPOLSAR-1.0: Multidimensional SAR multibandfully polarized fine classification dataset," *J. Radars*, vol. 13, no. 3, pp. 525–538, Mar. 2024.