Neurocomputing 658 (2025) 131662

Contents lists available at ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom

Graph information bottleneck for remote sensing segmentation

Yuntao Shou[a], Tao Meng[a,*], Wei Ai[a], Haiyan Liu[b], Keqin Li[c]

[a] College of Computer and Mathematics, Central South University of Forestry and Technology, Changsha, Hunan 410004, China
[b] College of Information Engineering, Changsha Medical University, Changsha, Hunan 410203, China
[c] Department of Computer Science, State University of New York, New Paltz, NY 12561, USA

ARTICLE INFO

Communicated by J. Han

Keywords:
Remote sensing segmentation
Graph learning
Contrastive learning
Information bottleneck

ABSTRACT

Remote sensing segmentation has a wide range of applications in environmental protection, urban change detection, etc. Despite the success of deep learning-based remote sensing segmentation methods (e.g., CNN and Transformer), they are not flexible enough to model irregular objects. In addition, existing graph contrastive learning methods usually adopt the approach of maximizing mutual information to keep the node representations consistent between different graph views, which may cause the model to learn task-independent redundant information (i.e., information unrelated to the downstream task, including both redundancy and noise.). To tackle the above problems, this paper treats images as graph structures and introduces a novel Graph Information Bottleneck for Remote Sensing Segmentation (GIB-RSS) architecture. Specifically, we construct a node-masking and edge-masking graph view to obtain an optimal graph structure representation, which can adaptively learn whether to mask nodes and edges. Here, the optimal graph structure representation refers to the refined node and edge embeddings derived from the masked graph views under the GIB objective, where task-relevant structural information is preserved while task-irrelevant redundancy and noise are suppressed. Furthermore, this paper innovatively introduces information bottleneck theory into graph contrastive learning to maximize task-related information while minimizing task-independent redundant information. Finally, we replace the convolutional module in UNet with the GIB-RSS module to complete the segmentation and classification tasks of remote sensing images. Extensive experiments on publicly available real datasets demonstrate that our method outperforms state-of-the-art remote sensing image segmentation methods.

1. Introduction

Remote sensing segmentation has been widely developed in a variety of scenarios including, land cover mapping, environmental protection, and road information extraction, which require high-quality feature representations to be learned from irregular objects (e.g., roads, trees, etc.) [1,2]. In recent years, thanks to the powerful modeling ability for image data, convolutional neural networks (CNNs) [3] and Transformer with attention module [4,5] have provided an effective way to extract the underlying visual features and multi-scale features of images and exhibit guaranteed performance in remote sensing segmentation [6].

Although encouraging segmentation performance has been achieved, CNN-based and Transformer-based remote sensing segmentation models suffer from some limitations. Taking Fig. 1 as an example, Fig. 1(a) shows the CNN-based image modeling method, which treats the image as a regular grid structure. Fig. 1(b) shows the Transformer-based image modeling method, which regards the image as a continuous sequence structure. Both of the above methods are unable to model irregular objects [7]. As shown in Fig. 1(c), we argue that both grid and sequence structures are special cases of graph structures and that GNN-based approaches [8–11] are capable of modeling data in non-Euclidean spaces. For instance, the vision GNN proposed by Han et al. [7] extracts low-level information about the image by treating the image as a graph structure. Therefore, we propose a GNN-based remote sensing image modeling method for multi-scale feature extraction of irregular objects. However, the convergence speed and convergence effect of GNN are unsatisfactory [12].

Recent advances in graph contrastive representation learning have demonstrated that it can improve model convergence and enhance model robustness [13]. Nevertheless, the existing methods suffer from two limitations. First, most existing methods perform feature augmentation by randomly masking graph views to obtain better node representations. However, randomly masking nodes and edges may

* Corresponding author.
Email address: mengtao@hnu.edu.cn (T. Meng).

https://doi.org/10.1016/j.neucom.2025.131662
Received 3 June 2025; Received in revised form 24 August 2025; Accepted 23 September 2025
Available online 11 October 2025

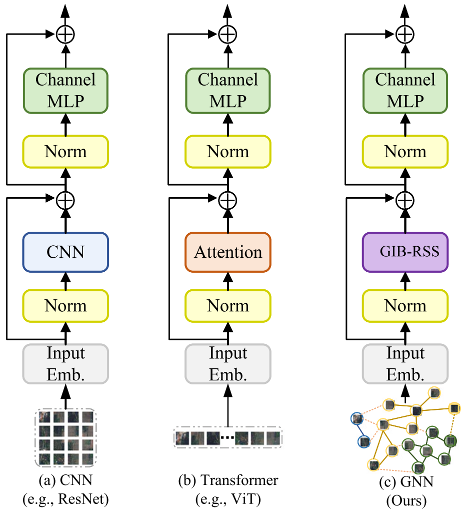

Fig. 1. Illustrative examples of different modeling approaches for an image. (a) CNNs view images as regular grid structures (i.e., squares and rectangles). (b) Transformer treats images as a continuous sequence structure. (c) We believe that both sequence structure and grid structure are special cases of graph structure, and graph structure can flexibly model regular and irregular objects. We thus view images as graph structures.

be too random, which destroys the expressive ability of the semantic information of the original graph. Second, most existing methods generate multiple contrastive views and enforce consistency by maximizing the mutual information (MI) between them (e.g., [13,14]). While this strategy improves representation robustness, it also has a potential drawback: it may lead the model to preserve task-independent or redundant patterns that exist across views but are not semantically informative for downstream tasks. For example, GraphCL [13] applies random structural augmentations (such as node dropping and edge perturbation) to generate graph views, and then maximizes MI between them. This may encourage the model to retain low-level topological patterns that are shared due to augmentation artifacts rather than task-relevant semantics. Similarly, InfoGraph [14] attempts to maximize MI between node-level and graph-level representations, but lacks explicit mechanisms to suppress noise or irrelevant patterns that may be reinforced by global pooling. In contrast, GIB [15] argues that effective representation learning in downstream tasks requires minimizing MI between the original graph and its latent encoding, thus discarding redundant structural cues while preserving task-relevant information. However, existing GIB-based frameworks usually assume random or fixed augmentations, which fail to adapt to structural heterogeneity in graph-structured data. In this paper, we reveal this limitation and propose an adaptive instantiation that extends GIB to node and edge-level views, deepening its theoretical foundation in graph contrastive learning.

To address the aforementioned issue, we propose a novel Graph Information Bottleneck for Remote Sensing Segmentation (GIB-RSS) method, which consists of two key steps, i.e., an adaptive feature augmentation module and a graph contrastive learning via an information bottleneck module.

First, we introduce a learnable graph contrastive view to adaptively learn whether to mask nodes and edges to improve the node representation ability of the original graph, which is optimized together with downstream remote sensing segmentation and classification in an end-to-end learning manner. The intuition behind the adaptive masking strategy is that random masking may discard minority class nodes, which aggravates the data imbalance in the graph structure. However, GCNs aggregate the information of surrounding neighbor nodes through the message-passing mechanism, which makes it easy for GCNs to reconstruct the feature information of popular nodes, but it is difficult to reconstruct the feature information of isolated nodes with low degrees. These adaptively masking-generated graph contrastive views increase the ability against imbalanced learning for remote sensing segmentation.

Second, we propose to integrate different graph-contrastive views into compact representations for downstream remote sensing segmentation tasks, which can further improve the feature representation capabilities of nodes. Recent advances have shown that downstream performance can be improved by fusing complementary semantic information between different views [12]. Therefore, we argue that maximizing the mutual information (MI) between graph contrastive views forces a consistent representation of the graph structure, which leads the model to capture task-independent redundant information. Inspired by the information bottleneck (IB) theory, we use it to minimize the MI between the original graph and the generated contrastive view while preserving task-relevant semantic information. Through the above approach, the model can jointly learn complementary semantic information between different views.

Compared with previous work, the contributions of this paper are summarized as follows.

1. We propose a novel Graph Information Bottleneck for Remote Sensing Segmentation (GIB-RSS) method, which enables flexible modeling of irregular objects.
2. We introduce a novel graph contrastive learning approach to optimize node representations by adaptively masking nodes and edges, which improves the representation ability of graph structure.
3. We innovatively embed the information bottleneck theory into the graph contrastive learning method, which can effectively eliminate redundant information while preserving task-related information.
4. Extensive experiments demonstrate that our method outperforms the state-of-the-art on three publicly available datasets.

2. Related work

2.1. CNN, and transformer for remote sensing segmentation

The early mainstream network architecture for remote sensing segmentation extracts visual features of images by using CNN. The earliest remote sensing image segmentation methods based on CNN are all evolved from FCN ([16,17], etc) and UNet (e.g., [18–20], etc). UNet extracts the context and location information of the image by designing a U-shaped structure based on the encoder and decoder, where both of them are composed of convolutional layers, skip connections, and pooling layers. FCN extracts image features through several convolutional layers and then connects a deconvolutional layer to obtain a feature map of the same size as the raw image, so as to predict the image pixel by pixel. However, both FCN and UNet algorithms need to down-sample to continuously expand the receptive field when extracting image features, which leads to the loss of image position information. To alleviate the problem of information loss caused by the downsampling operation, the DeepLab series [21] uses hole convolution to increase the receptive field to obtain multi-scale feature information. The HRNet proposed by Wang et al. [22] achieves high-resolution semantic segmentation by extracting feature maps of different resolutions and recovering high-resolution feature maps.

Transformer [23,24] is widely used in the image processing field because of its powerful global information processing capabilities. ViT proposed by Dosovitskiy et al. [4] applied the Transformer architecture to CV for the first time, and she used the attention to extract global visual features. Since the complexity of the attention is $O(n^2)$, this leads to a very large number of parameters in the model, and the model is difficult to train. To solve the above problems, Liu et al. [5] proposed Swin-Transformer, which improves the issue of high model complexity through a hierarchical attention mechanism. The Wide-Context Transformer proposed by Ding et al. [1] extracts global context information by introducing a Context Transformer while using CNN to extract features. Zhang et al. [3] extract multi-scale contextual features by combining Swin-Transformer and dilated convolutions and use a U-shaped decoder to achieve image semantic segmentation.

2.2. Graph neural networks

Kipf et al. [25] were the first to propose graph convolutional neural networks. In recent years, spatial-based GCNs and spectral-based GCNs have started to receive widespread attention, and they are applied to graph-structured data (e.g., social networks [26] and citation networks [27], etc.).

In recent years, graph neural networks (GNNs) has received extensive attention from researchers due to its powerful feature extraction capabilities, and it has been widely used in action recognition [28], point cloud analysis [29] and other fields [7]. GNNs can flexibly model irregular objects and extract global location feature information. In the remote sensing segmentation field, Saha et al. [30] use GNNs to aggregate and label unlabeled data to improve the ability of the model to approach the target domain.

2.3. Graph contrastive learning

Graph contrastive learning (GCL) aims to learn compact representations of nodes or subgraphs in graph data, emphasizing similarities within the same graph and differences between different graphs. GCL has been applied in many fields, including social network analysis, drug discovery, image analysis, etc. For example, in social networks, similarities between users can be discovered through GCL, and in drug discovery, potential drug similarities can be mined by contrasting molecular structures.

In recent research, DGI [31] and InfoGraph [14] obtain compact representations of graphs or nodes by maximizing the mutual information (MI) between different augmented views. MVGRL [12] argues that it can achieve optimal feature representation by contrasting first-order neighbor nodes and performing node diffusion to maximize the MI between subgraphs. GraphCL [13] constructs four types of augmented views and maximizes the MI between them. GraphCL enables better generalization performance on downstream tasks. However, GraphCL requires complex manual feature extraction. We argue that a good contrast-augmented view should be structurally heterogeneous while semantically similar, while previous research work maximizes the mutual information between nodes, which may lead to overfitting of the model. To solve the above issues, Wu et al. [15] introduced GIB to regulate redundancy in graph representation. However, their formulation does not consider the heterogeneity of contrastive graph views, nor the challenges of imbalanced graph structures. In contrast, our work extends GIB by integrating adaptive masking strategies and deriving graph-specific variational bounds, thereby deepening the theoretical understanding of GIB in contrastive learning.

3. Approach

In this section, we illustrate the construction of graph-structured data from images and introduce the GCL architecture with the information bottleneck to learn to extract global information locations of images.

3.1. Structure flow

Our main goal is to design an efficient modeling paradigm for global location information extraction of irregular objects, detailed in Fig. 2. For a given remote sensing image ($H \times W \times 3$), we first divide it into M patches. Then we map each image patch to a D-dimensional feature space $x_i \in \mathbb{R}^D$, and obtain a collection of feature vectors for an image X. We consider X to be a node in the graph, i.e., $V = \{v_1, v_2, v_N\}$. For node v_i, we use the KNN algorithm to find its K neighbors $N(v_i) = \{v_i^1, v_i^2, \ldots, v_i^K\}$. For $v_j \in N(v_i)$, we connect an edge e_{ji} from v_j to v_i. Through the above process, we get a directed graph $G = (V, E)$. Following UNet's network architecture design, feature embedding for images uses N encoders for feature encoding. Each stage consists of a GIB Embedding block, a skip connection module and a downsampling layer. GIB Embedding Block utilizes the inherent flexible modeling of non-Euclidean distance in the graph structure, follows the global modeling rules of node aggregation, and customizes the global position information interaction of the image. We downsample the feature maps with a 3×3 kernel. Similarly, the decoder stage consists of the proposed GE block and an upsampling layer to decode and reconstruct features. To ensure the effective utilization of information and the depth of network training, the decoder input of each stage is connected with the output of

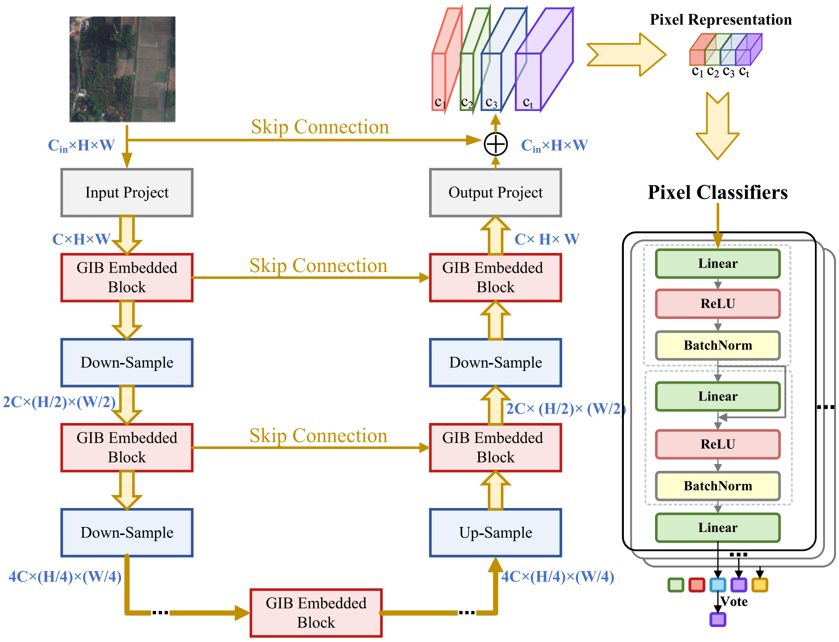

Fig. 2. The architecture of the proposed GIB-RSS method. Specifically, we first divide the image into patches and construct it as a graph. Then we replace the convolutional block in UNet with our GCN Block and use the constructed graph as the input. Finally, we build an MLP to classify pixels.

the encoder of the same stage. Finally, a convolutional layer is applied to generate the segmented image $S \in C_{in} \times H \times W$, which is predicted pixel by pixel.

3.2. GCN embedded block

The advantages of using a graph structure to model images are as follows: (1) The graph can flexibly handle data with non-Euclidean distances. (2) Compared with regular grid or sequence structures, graphs can model irregular objects while eliminating redundant information, and remote sensing images are mostly irregular objects. (3) The graph structure establishes the connection between objects (e.g., roads, trees, etc) through the connection between nodes and edges.

Specifically, for an input image feature X, we first construct a directed graph $G = G(x)$. To obtain the global location information of the image and update node features, we use graph convolution operations to aggregate and update node features. The formula is defined as follows:

$$\begin{aligned} G' &= F(G, \mathcal{W}) \\ &= \text{Update}\left(\text{Aggregate}\ \left(G, W_{\text{agg}}\right), W_{\text{update}}\right) \\ &= \text{LeakyReLU}\left(\sum_{r\in\mathcal{R}}\sum_{j\in\mathcal{N}_i^r}\frac{1}{|\mathcal{N}_i^r|}\left(\omega_{ij}^{(l)}W_{\theta_1}^{(l)}x_j^{(l)} + \omega_{ii}^{(l)}W_{\theta_2}^{(l)}x_i^{(l)}\right)\right) \end{aligned} \tag{1}$$

where W_{agg}, $W_{update}, W_{\theta_1}^{(l)}, W_{\theta_2}^{(l)}$ is learnable weights, w_{ij} is the edge weight between node i and node j, and its formula is defined as follows:

$$\begin{aligned} \omega_{ij}^{(l+1)} &= \text{softmax}\left(W^{(l)}[x_i^{(l)} \oplus x_j^{(l)}]\right) \\ &= \frac{\exp\left[x_i^{(l)} \oplus x_j^{(l)}\right]}{\sum_{\eta\in\mathcal{N}_i}\exp\left[x_i^{(l)} \oplus x_j^{(l)}\right]}, \end{aligned} \tag{2}$$

To capture the location information of key regions in the image, we further introduce a multi-head attention mechanism to update node features. The format is defined as follows:

$$\mathbf{x}_i' = \left[\text{head}^1 W_{\text{update}}^1, \text{head}^2 W_{\text{update}}^2, \ldots, \text{head}^h W_{\text{update}}^h\right] \tag{3}$$

where h represents the number of multi heads, we set $h = 4$.

We introduce the residual idea, and project node features to the same domain through a linear layer, which can help restore structural features and global position information. In addition, we also insert the LeakyReLU non-activation function to improve the nonlinear fitting ability of the model. The formula is expressed as follows:

$$Y = \text{LeakyReLU}\left(\text{GraphConv}\left(XW_{\text{in}}\right)\right)W_{\text{out}} + X \tag{4}$$

To improve the feature transformation ability of nodes and alleviate the over-smoothing phenomenon of GCN, we use feed-forward network (FFN) to perform feature mapping on each node again. The formula for FFN is defined as follows:

$$Y' = \text{LeakyReLU}\left(YW_1\right)W_2 + Y \tag{5}$$

where W_1 and W_2 are the learnable parameters.

3.3. Graph information bottleneck

The principle of graph information bottleneck (GIB) [15] is to introduce information bottleneck (IB) on the basis of GCL to perform contrastive learning between nodes or graphs. It forces the node representation Z_X to minimize the task-independent redundant information $\mathcal{D}$ and maximize the information Y relevant to the downstream tasks.

Specifically, we follow the local dependency assumption for graph-structured data: for a given node v, node v's first-order neighbor node data are related to node v, while the rest of the graph's data are independent and identically distributed with respect to node v. The hypothesis space represented by nodes can be constrained according to local dependency assumptions, which reduces the difficulty of GIB optimization. We assume that $\mathbb{P}(Z_X|\mathcal{D})$ represents modeling the correlation between node features hierarchically. In each iteration l, the representation of each node is optimized by aggregating surrounding neighbor node information and graph structure information $Z_A^{(l)}$. Therefore, the optimization goal of GIB is defined as follows:

$$\min_{\mathbb{P}(Z_X^{(L)}|\mathcal{D})\in\Omega} \text{GIB}_\beta(\mathcal{D}, Y; Z_X^{(L)}) \triangleq \left[-I(Y; Z_X^{(L)}) + \beta I(\mathcal{D}; Z_X^{(L)})\right] \tag{6}$$

where Ω conforms to the representation space of Markov chain probability dependence within a given data set $\mathcal{D}$, $I(,)$ represents mutual information between feature vectors, $Z_X^{(L)}$ represents the feature representations of the nodes, and β is the balance coefficient. In Eq. (6), the model only needs to optimize two distributions, i.e., $\mathbb{P}(Z_A^{(l)}|Z_X^{(l-1)}, A)$, and $\mathbb{P}(Z_X^{(l)}|Z_X^{(l-1)}, Z_A^{(l)})$, where $Z_A^{(l)}$ is the graph structure information.

However, in Eq. (6), calculating the mutual information $I(Y; Z_X^{(L)})$ and $I(\mathcal{D}; Z_X^{(L)})$ is a difficult estimation problem. Therefore, we follow the IB criterion to introduce variational bounds on $I(Y; Z_X^{(L)})$ and $I(\mathcal{D}; Z_X^{(L)})$ to effectively perform parameter optimization. We give the upper and lower bounds of $I(\mathcal{D}; Z_X^{(L)})$ and $I(Y; Z_X^{(L)})$ as shown in Theorems 1 and 2 respectively.

Theorem 1. *For any class distribution given $\mathbb{Q}_1(Y_v|Z_{X,v}^{(L)})$ for $v \in V$ and $\mathbb{Q}_2(Y)$ in a graph, we can obtain a theoretical lower bound for $I(Y; Z_X^{(L)})$:*

$$\begin{aligned} I(Y; Z_X^{(L)}) \geq 1 &+ \mathbb{E}\left[\log\frac{\prod_{v\in V}\mathbb{Q}_1(Y_v|Z_{X,v}^{(L)})}{\mathbb{Q}_2(Y)}\right] \\ &+ \mathbb{E}_{\mathbb{P}(Y)\mathbb{P}(Z_X^{(L)})}\left[\frac{\prod_{v\in V}\mathbb{Q}_1(Y_v|Z_{X,v}^{(L)})}{\mathbb{Q}_2(Y)}\right] \end{aligned} \tag{7}$$

Theorem 2. *For any given node feature distribution $\mathbb{Q}(Z_X^{(l)})$ and graph structure information distribution $\mathbb{Q}(Z_A^{(l)})$, we use Markov chain dependence to derive the upper bound of $I(Y; Z_X^{(L)})$ as follows:*

$$I\left(\mathcal{D}; Z_X^{(L)}\right) \leq I\left(\mathcal{D}; \{Z_X^{(l)}\}_{l\in S_X} \cup \{Z_A^{(l)}\}_{l\in S_A}\right) \leq \sum_{l\in S_A} AIB^{(l)} + \sum_{l\in S_X} XIB^{(l)} \tag{8}$$

where $l \in \{S_X, S_A\}$, and

$$\begin{aligned} AIB^{(l)} &= \mathbb{E}\left[\log\frac{\mathbb{P}(Z_A^{(l)}|A, Z_X^{(l-1)})}{\mathbb{Q}(Z_A^{(l)})}\right], \\ XIB^{(l)} &= \mathbb{E}\left[\log\frac{\mathbb{P}(Z_X^{(l)}|Z_X^{(l-1)}, Z_A^{(l)})}{\mathbb{Q}(Z_X^{(l)})}\right] \end{aligned} \tag{9}$$

where AIB and XIB represent the adjacency matrix features and the node features obtained using the IB criterion, respectively.

We optimize $\mathbb{P}(Z_A^{(l)}|Z_X^{(l-1)}, A)$ and $\mathbb{P}(Z_X^{(l)}|Z_X^{(l-1)}, Z_A^{(l)})$ given a theoretical upper and lower bound. Next, we will specify the optimization goals of GIB.

Objective for training. To update model parameters in GIB, we need to calculate the theoretical boundary of GIB in (6). Specifically, we use a uniform distribution to optimize the classification problem: $Z_A \sim \mathbb{Q}(Z_A), Z_{A,v} = \cup_{t=1}^{\mathcal{T}}\{u \in V_{vt}|u \overset{\text{iid}}{\sim} \text{Cat}(\frac{1}{|V_{vt}|})\}$. Therefore, we can obtain an estimate of AIB$^{(l)}$ as follows:

$$\widehat{\mathrm{AIB}}^{(l)} = \mathbb{E}_{\mathbb{P}(Z_A^{(l)}|A,Z_X^{(l-1)})}\left[\log\frac{\mathbb{P}(Z_A^{(l)}|A,Z_X^{(l-1)})}{\mathbb{Q}(Z_A^{(l)})}\right] \tag{10}$$

AIB$^{(l)}$ can be formally defined as follows:

$$\widehat{\mathrm{AIB}_{\mathrm{C}}}^{(l)} = \sum_{v\in V,t\in[\mathcal{T}]} \mathrm{KL}\left(\mathrm{Cat}(\phi_{vt}^{(l)})||\,\mathrm{Cat}\left(\frac{1}{|V_{vt}|}\right)\right) \tag{11}$$

For the estimation of XIB, we use a learnable Gaussian distribution to set $\mathbb{Q}(Z_X^{(l)})$. Specifically, for a given node v, $Z_X \sim \mathbb{Q}(Z_X)$, we assume $Z_{X,v} \sim \sum_{i=1}^m w_i \mathrm{Gaussian}(\mu_{0,i},\sigma_{0,i}^2)$. Therefore $\widehat{\mathrm{XIB}}^{(l)}$ is formally defined as follows:

$$\begin{aligned}\widehat{\mathrm{XIB}}^{(l)} &= \log\frac{\mathbb{P}(Z_X^{(l)}|Z_X^{(l-1)},Z_A^{(l)})}{\mathbb{Q}(Z_X^{(l)})}\\ &= \sum_{v\in V}\left[\log\Phi(Z_{X,v}^{(l)};\mu_v,\sigma_v^2) - \log\left(\sum_{i=1}^{m} w_i\Phi(Z_{X,v}^{(l)};\mu_{0,i},\sigma_{0,i}^2)\right)\right]\end{aligned} \tag{12}$$

where $\mu_{0,i},\sigma_{0,i},w_i$ are the learnable.

Combining Eqs. (11) and (12), we can estimate $I(\mathcal{D};Z_X^{(L)})$ as follows:

$$I(\mathcal{D};Z_X^{(L)}) \rightarrow \sum_{l\in S_A}\widehat{\mathrm{AIB}}^{(l)} + \sum_{l\in S_X}\widehat{\mathrm{XIB}}^{(l)} \tag{13}$$

We use cross entropy to estimate $I(Y;Z_X^{(L)})$ as follows:

$$I(Y;Z_X^{(L)}) \rightarrow -\sum_{v\in V}\text{Cross-Entropy}(Z_{X,v}^{(L)}W_{\mathrm{out}};Y_v) \tag{14}$$

By combining Eqs. (13) and (14), we can obtain the optimization objective of GIB.

3.4. Instantiating GIB-RSS

After detailing the optimization principles of GIB, we will explain the GIB-RSS architecture we designed as shown in Fig. 3. It is worth noting that this instantiation is not a simple application of GIB. By introducing learnable node- and edge-masking, we uncover a limitation of original GIB—random augmentations may discard rare but task-critical nodes. Our adaptive implementation extends GIB to heterogeneous multi-view contrastive learning, providing a new paradigm where the bottleneck is optimized not only within a single view but across multiple structurally diverse views.

Node-masking view. To improve the feature representation ability of nodes in the learning process, we perform learnable node masking before each information aggregation and feature update of GCN. The formula for the node mask view we created is as follows:

$$\mathcal{G}_{ND}^{(l)} = \left\{\left\{v_i \odot \eta_i^{(l)} \mid v_i \in \mathcal{V}\right\},\mathcal{E},\mathcal{R},\mathcal{W}\right\}, \tag{15}$$

where $\eta_i^{(l)} \in \{0,1\}$ is sampled from a parameterized Bernoulli distribution $Bern(\omega_i^l)$, and $\eta_i^{(l)} = 0$ represents masking node v_i, $\eta_i^{(l)} = 1$ represents keeping node v_i.

Edge-masking view. The goal of the edge-masking view is to generate an optimized graph structure, and the formula is defined as follows:

$$\mathcal{G}_{ED}^{(l)} = \left\{\mathcal{V},\left\{e_{ij}\odot\eta_{ij}^{(l)} \mid e_{ij}\in\mathcal{E},\mathcal{R},\mathcal{W}\right\}\right\}, \tag{16}$$

where $\eta_{ij}^{(l)} \in \{0,1\}$ is also sampled from a parameterized Bernoulli distribution $Bern()$, and $\eta_{ij}^{(l)} = 0$ represents masking edges e_{ij}, $\eta_i^{(l)} = 1$ represents keeping edge e_{ij}.

To enable the model to adaptively learn whether to mask nodes and edges, we introduce learnable parameters for nodes and edges as follows:

$$\hat{\mathbf{e}}_i^{(l)} = \omega_i^{(l)}\left(\mathbf{e}_i^{(l)}\right); \quad \hat{\mathbf{e}}_{ij}^{(l)} = \omega_{ij}^{(l)}\left(\left[\mathbf{e}_i^{(l)};\mathbf{e}_j^{(l)}\right]\right), \tag{17}$$

where $\omega_i^{(l)}$ and $\omega_{ij}^{(l)}$ are the learnable parameters.

To efficiently optimize the multi-view structure learning in an end-to-end manner, we adopt the reparameterization trick [32] and relax the binary mask variable ρ from being sampled directly from a Bernoulli distribution to a deterministic and differentiable function of a learnable parameter ω and an independent random variable ϵ, formulated as:

$$\rho = \sigma\left(\frac{\log\epsilon - \log(1-\epsilon) + \omega}{\tau}\right), \tag{18}$$

where $\epsilon \sim Uniform(0,1)$, $\tau \in \mathbb{R}^+$ is the temperature, and $\sigma(\cdot)$ is the sigmoid function. This relaxation ensures smooth gradients $\frac{\partial\rho}{\partial\omega}$, enabling

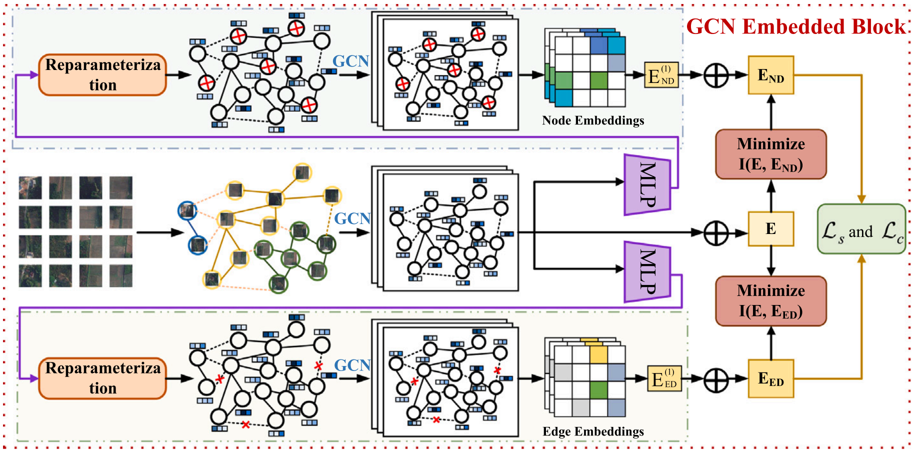

Fig. 3. The overview of the GCN Embedded Block framework. We generate two contrastive graph views through learnable node masking and edge masking mechanisms. Each view is encoded via GCN-based embeddings to produce E_{ND} (node-masked embeddings) and E_{ED} (edge-masked embeddings), respectively. Both are passed through shared MLPs to compute representations E, which are then regularized using the graph information bottleneck objective. The mutual information between the input graph structure and each embedding is minimized to remove redundant task-independent redundant information. The final loss combines the supervised segmentation loss $\mathcal{L}_s$ and the bottleneck regularization loss $\mathcal{L}_c$.

efficient end-to-end optimization of the learnable Node-Masking and Edge-Masking views.

In practice, the logits ω are generated by lightweight neural networks implemented as a multilayer perceptrons (MLPs) with two linear layers. For the node mask network, the input is the node feature vector output from the previous GCN layer, which passes through a MLPs and outputs a scalar logit; applying the sigmoid function yields the node retention probability. For the edge mask network, the input is the concatenated feature vectors of the two endpoint nodes $[\mathbf{e}_i; \mathbf{e}_j]$, which are fed into a similar MLPs to produce the edge retention probability. These networks are lightweight auxiliary modules, independent from but trained jointly with the GCN layers. While their parameters are not shared with the GCN, they are conditioned on the evolving graph embeddings, allowing masking decisions to adapt dynamically to the representation learning process. During training, node or edge dropping is guided by the learned probabilities, while at inference we drop nodes or edges with a probability less than 0.5 to maintain structural consistency.

After obtaining the masked node and edge-masking views, we input them into GCN for feature representation to obtain optimized multi-views. The formula is defined as follows:

$$\begin{aligned} \mathbf{E}_{ND}^{(l)} &= GraphConv\left(\mathbf{E}_{ND}^{(l-1)}, \mathcal{G}_{ND}^{(l)}\right), \\ \mathbf{E}_{ED}^{(l)} &= GraphConv\left(\mathbf{E}_{ED}^{(l-1)}, \mathcal{G}_{ED}^{(l)}\right). \end{aligned} \tag{19}$$

where $GraphConv$ represents the graph convolution operation, and we choose GAT as our graph encoder. $\mathbf{E}_{ND}$ and $\mathbf{E}_{ED}$ represent the node feature representations of node-masking view and edge-masking view respectively, $\mathbf{G}_{ND}$ and $\mathbf{G}_{ED}$ represent node-masking view and edge-masking view respectively.

After obtaining the node mask and edge mask views, we combine Eqs. (13) and (14) to jointly optimize the self-supervised losses $\mathcal{L}_s$ and $\mathcal{L}_c$ as follows:

$$\begin{aligned} \min(\mathcal{L}_s + \mathcal{L}_c) = I(\mathcal{D}^{(ED)}; Z_X^{(ED)}) + I(Y; Z_X^{(ED)}) \\ + I(\mathcal{D}^{(ND)}; Z_X^{(ND)}) + I(Y; Z_X^{(ND)}) \end{aligned} \tag{20}$$

where $\mathcal{D}^{(ED)}$ and $\mathcal{D}^{(ND)}$ represent the graph structure of the node-masking and edge-masking views respectively, and $Z_X^{(ED)}$ and $Z_X^{(ND)}$ represent the node features of the node-masking and edge-masking views respectively. Notably, we do not introduce additional loss terms or regularization for mask selection. The mutual information loss functions (e.g., $XIB^{(l)}$ and $AIB^{(l)}$ in Eqs. (11) and (12) inherently supervise the learning of meaningful and sparse masks by encouraging the embeddings to preserve task-relevant information while discarding task-independent redundancy.

3.5. Model training

All components of the proposed GIB-RSS architecture are trained jointly in an end-to-end fashion. The encoder–decoder structure, composed of multiple GIB Embedded Blocks, is fully differentiable. Each GIB block processes the graph at a different resolution level, following the UNet design, and contributes to the final segmentation output. Importantly, all GIB blocks share a unified training process and are optimized simultaneously, rather than in a stage-wise or separate manner.

During training, we optimize a total loss function that consists of two parts:

- Supervised segmentation loss $\mathcal{L}_{\text{seg}}$, which encourages accurate pixel-level classification on labeled remote sensing images.
- Graph information bottleneck loss on two contrastive views (node-masked and edge-masked), which regularizes the mutual information between node representations and graph structures, denoted as $\mathcal{L}_c$ and $\mathcal{L}_s$, respectively.

The final objective function is defined as:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{seg}} + \alpha \cdot \left(\mathcal{L}_c + \mathcal{L}_c\right) \tag{21}$$

where α is a balancing hyperparameter. The supervised loss $\mathcal{L}_{\text{seg}}$ is the cross-entropy loss defined in Eq. (20). The bottleneck losses $\mathcal{L}_{\text{GIB}}^{\text{ND}}$ and $\mathcal{L}_{\text{GIB}}^{\text{ED}}$ are derived from the upper and lower bounds of mutual information as defined in Eqs. (13) and (14), computed independently for each view.

All model parameters, including the GIB block parameters, multi-head attention weights, Bernoulli masking generators, and pixel classifiers, are updated jointly via backpropagation using the AdamW optimizer. This end-to-end training scheme ensures that the model learns semantically meaningful and task-relevant node representations, while effectively suppressing redundant information.

4. Experiments

In this section, we verify the effectiveness of the proposed GIB-RSS on remote sensing image segmentation tasks.

4.1. Benchmark datasets used

For the GIB-RSS model, we use the widely used datasets UAVid [33], Vaihingen [34] and Potsdam [35] datasets for experimental evaluation. The UAVid dataset comes with two spatial resolutions. Specifically, the UAVid dataset contains a total of 420 images, and each image is cropped to a size of 1024 × 1024. The Vaihingen dataset consists of 33 images with a spatial resolution of 2494 × 2064. Each image is cropped to 1024 × 1024. The Potsdam dataset contains 38 image patches with a spatial resolution of 6000 × 6000, and we crop the original image size to 1024 × 1024. The LoveDA dataset contains 5, 987 high-resolution remote sensing images with size 1024 × 1024, 2522 images are used for training, 1669 images are used for validation, and 1796 images are used for testing. The data information of the dataset is shown in Table 1.

4.2. Experimental settings

GIB-RSS is implemented on NVIDIA A100 GPU with 80 G memory using PyTorch framework. For the hyperparameters in the experiments, the paper utilize the AdamW optimizer for gradient updates. The GIB-RSS's learning rate (LR) is set to 5e-4 and a cosine learning rate decay is utilized to dynamically adjust the LR. During model training, we use a random flip strategy for data augmentation. For the UAVid dataset, we crop the image size to 1024 × 1024. For Vaihinge, Potsdam datasets, we crop images to 512 × 512. When the GIB-RSS is trained, we set epoch to 80, and batch size to 32.

4.3. Evaluation metrics

We used multiple evaluation metrics to evaluate the experimental performance of all models, including Overall Accuracy (OA), meanF1, and mIoU. OA, F1 and mIOU reflect the accuracy of remote sensing image segmentation from different perspectives.

4.4. Baseline models

MSD: The Multi-Scale-Dilation (MSD) method proposed by Lyu et al. [33] achieves image segmentation by using a large-scale pre-trained model to extract multi-scale features of the image.

Table 1
The division of the train set, val set and test set in the benchmark dataset and the resolution information of the images.

Datasets	Resolutions	Train	Test	Val
UAVid	3840 × 2160/4096 × 2160	200	150	70
Vaihingen	2494 × 2064	15	17	1
Potsdam	6000 × 6000	22	14	1
LoveDA	1024 × 1024	2522	1796	1669

CANet: The Context Aggregation Network (CANet) proposed by Yang et al. [36] effectively extracts the spatial information and global information of the image by building a dual-branch CNN and uses an aggregation mechanism to fuse the spatial and global context information.

DANet: The dual attention network (DANet) proposed by Fu et al. [37] achieves the extraction and fusion of global and local semantic information in space and channels.

SwiftNet: SwiftNet proposed by Orsic et al. [38] uses a pyramid structure to perform feature fusion of local information. SwiftNet adds regularization terms to constrain the model during the optimization process.

BiSeNet: The Bilateral Segmentation Network (BiSeNet) proposed by Yu et al. [39] extracts spatial information and high-resolution features by setting small-stride spatial convolution kernels. At the same time, a down-sampling strategy is used to extract contextual information, and a fusion module is designed to achieve effective fusion of information.

MANet: The multi-attention network (MANet) proposed by Li et al. [40] reduces the computational load of the model by constructing a linear attention module to ensure modeling context dependencies.

ABCNet: The Attention Bilateral Context Network (ABCNet) proposed by Li et al. [41] can lightweightly extract spatial information and contextual information of images.

Segmenter: Segmenter proposed by Strudel et al. [42] introduces ViT to realize the modeling of global context information. Unlike CNNs, Segmenter can obtain class labels pixel by pixel.

SegFormer: SegFormer proposed by Xie et al. [43] combines Transformer and MLP to extract multi-scale features of images in a hierarchical manner.

BANet: Wang et al. [44] proposed a bilateral perception network (BANet) to extract texture information and boundary information in images in a fine-grained manner. BANet is based on the Transformer pre-training model to achieve information fusion.

BoTNet: The BoTNet proposed by Srinivas et al. [45] integrates the self-attention mechanism into the ResNet module to extract the global context information of the image.

TransUNet: TransUNet proposed by Chen et al. [46] embeds Transformer's self-attention mechanism into the structure of UNet so that the model can better capture the global relationship of the input image.

ShelfNet: ShelfNet proposed by Zhuang et al. [47] adopts a multi-resolution processing strategy, which processes input images at different levels. Such a design allows the network to better capture local details in the image while retaining the global information of the image.

CoaT: CoaT proposed by Xu et al. [48] adopts a co-scaling mechanism to maintain the integrity of the Transformers encoder branch at different scales and provides rich multi-scale and contextual information.

UNetFormer: UNetFormer proposed by Wang et al. [49] introduces the Transformer mechanism based on UNet. In UNetFormer, Transformer is used to better capture the global contextual information in the image and improve the model's ability to understand the overall structure.

5. Results and discussion

To illustrate the superiority of our proposed method GIB-RSS, we conduct experiments on four benchmark datasets (i.e., UAVid, Vaihingen, LoveDA, and Potsdam). The experimental results are shown in Tables 2–5. GIB-RSS outperforms the existing state-of-the-art comparison algorithms.

Specifically, on the UAViD dataset as shown in Table 2, GIB-RSS's mIoU value is 70.6 %, which is 3 % to 11 % higher than other models. The segmentation accuracy in other categories is also better than other comparison algorithms. For example, the IoU values of segmentation on cluster, road, tree, and vegetation have all reached SOTA, which is significantly better than existing methods. Although the IoU values on building, moving car, and human are not optimal, the difference from the best segmentation results is relatively small. Among other comparison algorithms, UNetFormer's effect is slightly lower than our algorithm, with an mIoU value of 67.8 %. We believe this is due to the fact that the architecture we designed is more suitable for segmenting irregular objects. Except for UNetFormer, the mIoU values of other comparison algorithms are significantly lower than the method GIB-RSS proposed in this paper.

On the Vaihingen dataset as shown in Table 3, GSIB-RSS's mIoU value is 85.3 %, which is 2 % to 6 % higher than other models. OA and meanF1 values are also higher than other methods. Specifically, the segmentation IoU value of our method GIB-RSS in four categories is significantly better than that of other comparison algorithms. It is only lower than some comparison algorithms (e.g., UNetFormer and Segmenter, etc.) in the tree category. The effect of UNetFormer is second, its mIoU value is 67.8 %, which is 1.8 % lower than GIB-RSS. The segmentation effects of other comparison algorithms are significantly lower than GIB-RSS and UNetFormer, even if they use some pre-trained models with better performance.

On the Potsdam dataset as shown in Table 4, GIB-RSS's mIoU value is 87.8 %, which is 1 % to 12 % higher than other models. Our algorithm GIB-RSS is significantly better than other comparison algorithms in the segmentation effects of all categories. Similarly, UNetFormer has the second best segmentation effect on the Potsdam dataset, with an mIoU value of 87.4 %. Other comparison algorithms usually use pre-trained models such as ResNet or ViT as backbones to fine-tune downstream tasks. Although the segmentation effect on the Potsdam dataset is acceptable, it is lower than GIB-RSS.

On the LoveDA dataset as shown in Table 5, GIB-RSS can achieve optimal segmentation results in all categories. In addition, GIB-RSS has

Table 2
Experimental results of our method and SOTA methods on the UAVid dataset. The optimal values in columns are shown in bold.

Methods	Backbone	Clutter	Building	Road	Tree	Vegetation	MovingCar	StaticCar	Human	mIoU
MSD	–	56.8	79.6	73.9	73.9	56.1	63.2	31.8	20.0	56.9
CANet	–	65.8	87.0	61.9	78.8	77.9	48.0	**68.5**	20.0	63.5
DANet	ResNet	65.1	86.2	78.0	77.9	60.9	60.0	47.1	8.9	60.5
SwiftNet	ResNet	63.9	84.9	61.3	78.3	76.4	51.2	62.4	15.8	61.8
BiSeNet	ResNet	64.5	85.8	61.0	78.1	77.1	48.8	63.2	17.4	62.0
MANet	ResNet	64.4	85.1	77.9	77.4	60.5	67.5	53.4	14.6	62.6
ABCNet	ResNet	67.3	86.1	81.5	79.7	63.3	69.2	48.3	13.6	63.6
Segmenter	ViT-Tiny	63.7	85.2	80.1	77.0	58.1	58.4	35.3	13.9	59.0
SegFormer	MiT-B1	67.3	87.3	79.8	80.1	62.7	71.7	52.7	29.3	66.3
BANet	ResT-Lite	65.9	86.0	81.2	79.1	61.9	68.7	52.4	20.5	64.4
BoTNet	ResNet	65.4	85.1	79.1	78.4	61.2	66.3	52.0	23.1	63.8
CoaT	CoaT-Mini	68.9	**89.1**	79.8	80.4	61.7	69.5	60.2	19.1	66.1
UNetFormer	ResNet	67.7	86.4	82.0	81.2	64.1	**74.0**	55.8	**30.9**	67.8
GIB-RSS	–	**71.2**	89.0	**83.0**	**81.9**	**79.7**	70.6	59.3	29.9	**70.6**

Table 3
Experimental results of our method and SOTA lightweight methods on the Vaihingen dataset. The optimal values in columns are shown in bold.

Methods	Backbone	Imp.suf.	Building	Lowveg.	Tree	Car	MeanF1	OA	mIoU
DABNet	–	88.0	89.1	73.9	85.0	59.9	79.2	83.9	69.9
ERFNet	–	88.7	89.8	76.2	86.1	54.0	79.0	86.2	70.3
BiSeNet	ResNet	88.7	90.7	81.0	87.1	72.9	84.1	86.6	76.3
PSPNet	ResNet	88.8	92.9	81.8	88.1	44.2	79.2	88.0	76.1
DANet	ResNet	89.7	94.1	81.9	86.9	44.6	79.4	87.6	69.6
FANet	ResNet	91.2	94.1	83.1	88.7	72.0	85.8	90.0	76.0
EaNet	ResNet	92.1	94.7	82.9	88.8	80.4	87.8	90.0	78.5
ShelfNet	ResNet	92.1	94.8	84.1	88.9	78.0	87.6	90.1	77.9
MARsU-Net	ResNet	91.8	94.8	84.1	89.0	78.2	87.6	89.7	78.9
SwiftNet	ResNet	91.9	95.1	83.6	89.6	80.6	88.2	90.0	79.2
ABCNet	ResNet	93.1	94.8	84.8	90.0	84.7	89.5	91.2	81.0
BoTNet	ResNet	90.0	91.6	82.4	89.1	72.4	85.1	87.8	74.2
BANet	ResT-Lite	91.6	94.8	84.0	90.3	87.2	89.6	90.5	81.4
Segmenter	ViT-Tiny	90.1	92.6	80.7	90.3	68.2	84.4	87.6	73.7
UNetFormer	ResNet	93.1	94.9	85.2	**90.8**	88.2	90.4	90.7	83.2
GIB-RSS	–	**94.7**	**96.8**	**86.8**	89.9	**91.5**	**91.8**	**92.9**	**85.3**

Table 4
Experimental results of our method and SOTA lightweight methods on the Potsdam dataset. The optimal values in columns are shown in bold.

Methods	Backbone	Imp.suf.	Building	Lowveg.	Tree	Car	MeanF1	OA	mIoU
ERFNet	–	89.9	92.6	81.0	76.4	90.8	86.2	84.6	75.9
DABNet	–	90.0	92.7	83.3	81.9	93.1	88.0	87.2	79.4
BiSeNet	ResNet	90.2	94.6	85.5	86.2	92.7	89.8	88.2	81.7
EaNet	ResNet	92.0	95.7	84.3	85.7	95.1	90.6	88.7	83.4
MARsU-Net	ResNet	91.4	95.6	85.8	86.6	93.3	90.5	89.0	83.9
DANet	ResNet	91.0	95.6	86.1	87.6	84.3	88.9	89.1	80.3
SwiftNet	ResNet	91.8	95.9	85.7	86.8	94.5	91.0	89.3	83.8
FANet	ResNet	92.0	96.1	86.0	87.8	94.5	91.3	89.9	84.2
ShelfNet	ResNet	92.5	95.8	86.6	87.1	94.6	91.3	89.9	84.4
ABCNet	ResNet	93.5	96.9	87.9	89.1	95.8	92.7	91.3	86.5
Segmenter	ViT-Tiny	90.9	94.6	84.9	84.7	89.1	88.7	89.3	81.1
BANet	ResT-Lite	92.6	95.8	86.5	88.9	96.2	91.9	91.7	85.7
SwinUperNet	Swin-Tiny	92.7	96.5	88.0	88.4	95.8	91.7	91.2	86.0
UNetFormer	ResNet	93.8	96.9	88.1	89.3	96.8	93.1	91.0	87.4
Mask2Former	IMP	88.4	92.9	83.1	84.0	86.00	-	87.5	86.9
GIB-RSS	–	**94.9**	**97.9**	**88.7**	**90.7**	**97.2**	**93.9**	**93.5**	**87.8**

Table 5
Experimental results of our method and state-of-the-art methods on the LoveDA dataset. The optimal values in columns are shown in bold.

Methods	Backbone	Background	Building	Road	Water	Barren	Forest	Agriculture	mIoU	Complexity	Speed
PSPNet	ResNet50	44.4	52.1	53.5	76.5	9.7	44.1	57.9	48.3	105.7	52.2
DeepLabV3 + +	ResNet50	43.0	50.9	52.0	74.4	10.4	44.2	58.5	47.6	95.8	53.7
SemanticFPN	ResNet50	42.9	51.5	53.4	74.7	11.2	44.6	58.7	48.2	103.3	52.7
FarSeg	ResNet50	43.1	51.5	53.9	76.6	9.8	43.3	58.9	48.2	–	47.8
FactSeg	ResNet50	42.6	53.6	52.8	76.9	16.2	42.9	57.5	48.9	–	46.7
BAnet	ResNet50	43.7	51.5	51.1	76.9	16.6	44.9	62.5	49.6	52.6	11.5
TransUNet	ViT-R50	43.0	56.1	53.7	78.0	9.3	44.9	56.9	48.9	803.4	13.4
Segmenter	ViT-Tiny	38.0	50.7	48.7	77.4	13.3	43.5	58.2	47.1	26.8	14.7
SwinUperNet	Swin-Tiny	43.3	54.3	54.3	78.7	14.9	45.3	59.6	50.0	349.1	19.5
DC-Swin	Swin-Tiny	41.3	54.5	56.2	78.1	14.5	47.2	62.4	50.6	183.8	23.6
UNetFormer	ResNet18	44.7	58.8	54.9	79.6	20.1	46.0	62.5	52.4	46.9	115.3
GIB-RSS	–	**45.8**	**59.6**	**56.4**	**80.4**	**21.2**	**48.2**	**63.7**	**54.1**	**34.2**	**122.1**

a model parameter volume of 34.2 M and an inference speed of 122.1 FPS, which is far superior to other comparison algorithms. Like other comparison algorithms, due to the use of large-scale pre-training models, this results in a relatively large number of model parameters and slow inference speed.

The performance improvement may be attributed to our method's ability to flexibly model irregular objects, and the introduction of the multi-head attention effectively improves the model's capture of key position information in the image. At the same time, we also introduced the information bottleneck theory to perform graph comparison learning. Unlike the previous GCL method, GIB obtains optimal graph structure representation by minimizing the mutual information between nodes. The intuition behind this is that a good augmented multi-view should be structurally heterogeneous but semantically similar. However, the existing methods are all based on CNN or Transformer architecture, and their ability to extract global positional information of irregular objects is worse than GNN.

6. Sensitivity analysis of α

Fig. 4 illustrates the effect of varying α on the segmentation performance across UAVid, Vaihingen, Potsdam, and LoveDA. The results show that the model performance generally improves as α increases from 0.1, reaching the highest accuracy within the range of 0.5–0.8, and then slightly decreases when α becomes too large. For UAVid and Potsdam, the mIoU peaks around $\alpha = 0.7$, achieving approximately 72 % and 90 %

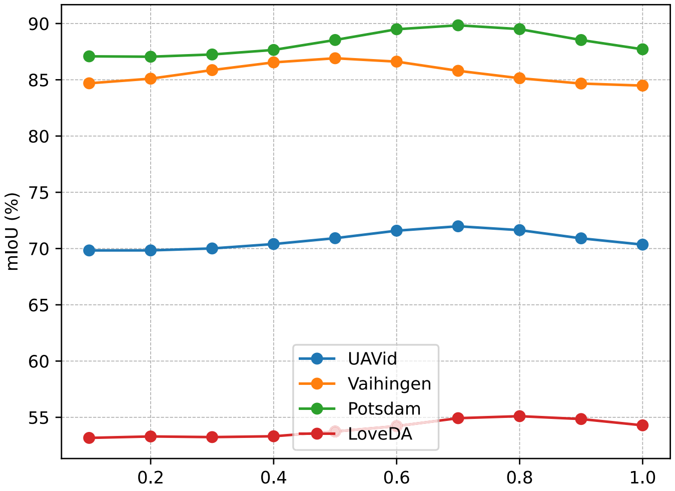

Fig. 4. Sensitivity analysis of the hyperparameter α on four benchmark datasets (UAVid, Vaihingen, Potsdam, and LoveDA).

respectively, while Vaihingen attains its best performance near $\alpha = 0.5$ with an mIoU close to 87 %. LoveDA, despite having lower absolute values, also benefits from stronger bottleneck regularization and shows its highest score at $\alpha = 0.8$. These observations indicate that although α is critical in balancing segmentation loss and bottleneck regularization, the model remains stable in a moderate range, confirming that setting α between 0.5 and 0.8 is a robust choice across different datasets.

7. Complexity and inference speed analysis

To further evaluate the practicality and efficiency of the proposed GIB-RSS model, we compare its computational complexity and inference speed with several representative baseline methods on the LoveDA test set. As shown in Table 6, the comparison includes the number of model parameters (denoted as “Complexity (M)”) and inference speed measured in frames per second (FPS). All measurements are conducted using 1024×1024 input resolution on a single NVIDIA GTX 3090 GPU. Our GIB-RSS achieves a competitive trade-off between performance and efficiency. Specifically, it requires 87.9 million parameters and achieves 61.7 FPS, which is significantly faster than most Transformer-based models such as TransUNet (13.4 FPS), Segmenter (14.7 FPS), and DC-Swin (23.6 FPS), while maintaining a much smaller parameter size compared to models like TransUNet (803.4 M) or SwinUperNet (349.1 M). Although slightly larger than UNetFormer (46.9 M), GIB-RSS outperforms it in segmentation accuracy, as shown in Table 5. This result demonstrates that GIB-RSS not only provides strong segmentation accuracy but also maintains practical runtime efficiency and model size, making it suitable for deployment in real-world remote sensing systems.

Table 6
Quantitative comparison on the LoveDA test set against other networks. Complexity and inference speed are evaluated using 1024 × 1024 inputs with a single NVIDIA GTX 3090 GPU.

Method	Backbone	Complexity (M)	Speed (FPS)
PSPNet	ResNet50	105.7	52.2
DeepLabV3 +	ResNet50	95.8	53.7
SemanticFPN	ResNet50	103.3	52.7
FarSeg	ResNet50	—	47.8
FactSeg	ResNet50	—	46.7
BANet	ResT-Lite	52.6	11.5
TransUNet	ViT-R50	803.4	13.4
Segmenter	ViT-Tiny	26.8	14.7
SwinUperNet	Swin-Tiny	349.1	19.5
DC-Swin	Swin-Tiny	183.8	23.6
UNetFormer	ResNet18	46.9	115.3
GIB-RSS	—	87.9	61.7

8. Visualization of segmentation results

As shown in Figs. 5–7, we also intuitively display the segmentation results of the model. The visualized segmentation results demonstrate the effectiveness of our designed GIB-RSS in dealing with challenging irregular objects.

Specifically, in Fig. 5, we see that GIB-RSS can more accurately segment trees and buildings than other SOTA models, and the cases of incorrect segmentation are relatively small. Other models easily misclassify lowveg. categories as background categories, and they fail to learn better for building category boundaries. In particular, in the first row of images, existing methods cannot segment the tree category well, either identifying it as background or identifying it as other categories. In the second row of pictures, existing comparison methods cannot segment some relatively small categories well, while GIB-RSS can segment small irregular objects better. In the third row of images, GIB-RSS can better segment the boundary areas of two different categories.

As shown in Fig. 6, our proposed model is more clearly distinguishes the difference between trees and lowveg. The experimental results show that GIB-RSS more effectively learn the boundary information between different categories. The class boundary learning ability of other models is significantly worse than GIB-RSS. Specifically, in the first row of

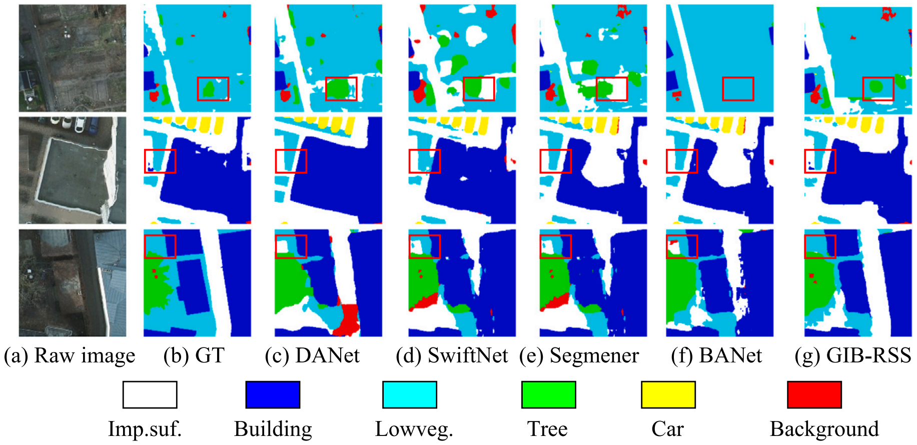
(a) Raw image (b) GT (c) DANet (d) SwiftNet (e) Segmener (f) BANet (g) GIB-RSS

Fig. 5. Visualization of the segmentation results of different models on the Postdam dataset.

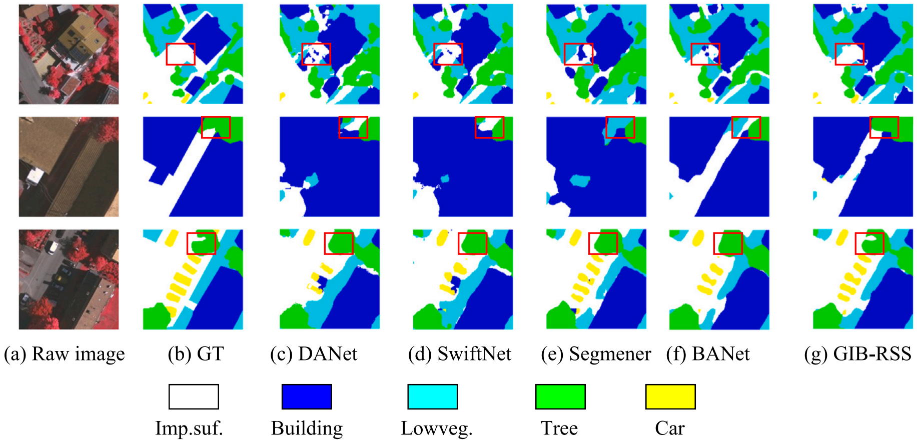

Fig. 6. Visualization of the segmentation results of different models on the Vaihingen dataset.

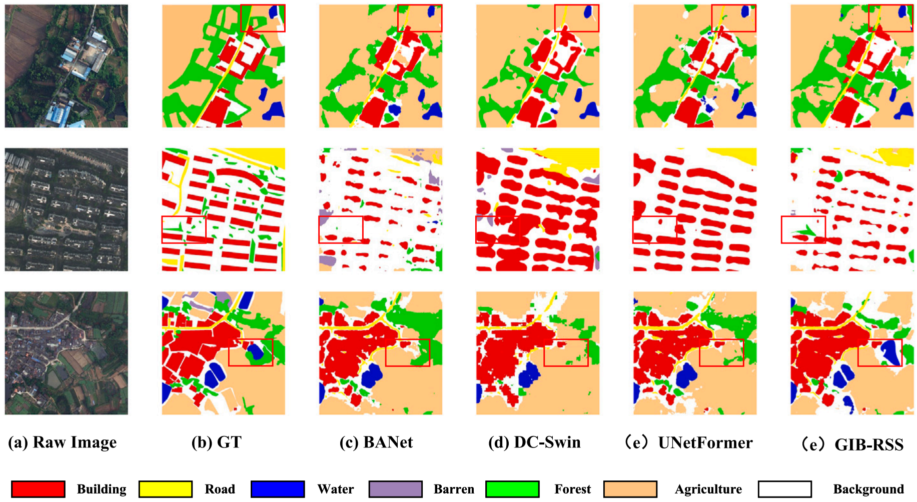

Fig. 7. Visualization of the segmentation results of different models on the LoveDA dataset.

images, our method can better identify the background area, while other comparison methods easily misclassify the background area as a building category. In the second row of pictures, GIB-RSS can segment the tree category relatively completely, while other methods easily identify the tree category as a background category or other categories. In the third row of images, GIB-RSS can sensitively detect the boundary areas of categories, while other methods cannot correctly segment the boundary areas of categories.

As shown in Fig. 7, in the first row of images, existing methods cannot correctly classify the tree category, but incorrectly classify it as the agriculture category. Unlike contrastive methods, GIB-RSS can well distinguish the difference between two categories and achieve better class boundary segmentation. In the second row of pictures, since the segmented objects are relatively small, existing methods cannot perform fine-grained segmentation on them. GIB-RSS can segment small objects at fine granularity while also distinguishing differences between tree and background categories. In the third row of images, none of the existing comparison methods can segment the water category, while GIB-RSS can segment them accurately. Experimental results demonstrate the superior segmentation performance of the GIB method for irregular objects.

9. Ablation study

We conduct ablation studies of our model GIB-RSS on four segmentation datasets to illustrate the effectiveness of our used modules.

Table 7
Experimental results of different types of graph convolutional neural networks on datasets. We choose the mIoU value as our evaluation metric.

GraphConv	UAVid	Vaihingen	Potsdam	LoveDA
EdgeConv	69.5	84.4	86.9	53.6
GIN	68.7	83.6	86.7	53.1
GraphSAGE	68.6	83.4	86.2	52.7
GAT	**70.6**	**85.3**	**87.8**	**54.1**

9.1. Type of graph convolution

In experiments we explore the performance of three different graph convolution variants on segmentation, including EdgeConv, GIN, GraphSAGE, and GAT. As shown in Table 7, GAT achieves the highest accuracy with mIoU values of 79.6 %, 85.3 %, 87.8 % and 54.1 % on the four datasets. The effect of EdgeConv is second, with mIoU values of 69.5 %, 84.4 %, 86.9 % and 53.6 % on the four datasets. The effect of GraSAGE is worst, with mIoU values of 68.6 %, 83.4 %, 86.2 % and 52.7 % on the four datasets. The performance improvement may be attributed to GAT's ability to capture key region information in the image.

9.2. The effects of modules in GIB-RSS

To illustrate that the modules (i.e., node-masking and edge-masking) proposed in this paper can better improve the performance of GNN

Table 8
The effectiveness of the proposed three core modules (i.e., GNN, Node-Masking (NM), Edge-Masking (EM)) is verified by ablation experiments on the dataset. We choose the mIoU value as our evaluation metric.

GNN	NM	EM	UAVid	Vaihingen	Potsdam	LoveDA
✓	✗	✗	67.5	81.8	86.2	53.0
✓	✓	✗	68.0	81.7	85.4	53.6
✓	✗	✓	68.2	82.3	86.5	53.8
✓	✓	✓	**70.6**	**85.3**	**87.8**	**54.1**

Table 9
The influence of different number of neighbor nodes K on the experimental results. We choose the mIoU value as our evaluation metric.

K	UAVid	Vaihingen	Potsdam	LoveDA
3	66.7	82.5	84.4	52.1
6	67.6	82.9	85.8	52.7
9	67.8	83.3	85.9	52.9
12	68.9	84.2	86.6	53.6
15	**70.6**	**85.3**	**87.8**	**54.1**
18	69.3	83.9	87.0	53.6

in the field of image segmentation, we verify the effect of these modules through ablation studies. We use node mask view and edge mask view with information bottleneck criterion to improve the generalization ability of the model. From Table 8 we can see that the performance of image segmentation using graph convolution alone is not competitive. The accuracy of segmentation can be improved by introducing node-masking and edge-masking. Specifically, the model performs best when using both node mask and edge mask views, with mIoU values of 70.6 %, 85.3 %, 87.8 % and 54.1 % respectively. When only using the node mask view, the effect of the model is second, with mIoU values of 68.2 %, 82.3 %, 86.5 % and 53.8 % respectively. The model has the worst performance when the node mask and edge mask views are not used, with mIoU values of 67.5 %, 81.8 %, 86.2 % and 53.0 % respectively. In summary, applying GIB directly on GCN yields limited improvements, confirming that original GIB may over-suppress rare features in imbalanced graphs. By contrast, our adaptive masking instantiation consistently boosts performance, demonstrating the necessity of extending GIB to view-specific regulation. These results empirically reveal limitations of the original GIB and verify our theoretical extension.

9.3. The number of neighbors

The number of neighbor nodes K is a hyperparameter controlling information aggregation. Too few neighbor nodes will lead to low frequency of information exchange, and the global position information cannot be fully extracted, while too many neighbors will lead to over-smoothing of the model. Based on the above analysis, we adjusted the range of K from 3 to 18, and the results are shown in Table 9. When the number of neighbor nodes K is 15, the segmentation effect is better. When the number of nodes K is less than 15, the effect of the model increases as the number of K increases, and when K is greater than 15, the training effect of the model begins to show a downward trend. The above phenomenon is consistent with our analysis.

10. Potential applications in low-shot and zero-shot settings

Although the proposed GIB-RSS framework is primarily designed for fully supervised remote sensing segmentation tasks, its architectural characteristics naturally lend themselves to extension in data-scarce environments such as low-shot or zero-shot learning. We briefly outline the potential directions for applying GIB-RSS in these challenging settings. (1) Self-Supervised Pretraining for Low-Data Segmentation. Inspired by recent work such as Unsupervised Pre-training with Language-Vision Prompts for Low-Data Instance Segmentation [50], GIB-RSS can be adapted as a self-supervised pretraining module. Specifically, our graph contrastive learning mechanism, coupled with the information bottleneck, can learn structure-aware and task-relevant representations from unlabeled remote sensing data. These pretrained representations can then be fine-tuned on downstream segmentation tasks with limited annotations, offering a promising route to enhance label efficiency. (2) Generalization to Unseen Categories via Feature Synthesis. The robust semantic feature modeling capabilities of GIB-RSS also make it potentially applicable to zero-shot segmentation or detection tasks. For example, the memory-based region synthesis strategy employed in M-RRFS [51]: A Memory-based Robust Region Feature Synthesizer for Zero-shot Object Detection can be integrated with our graph-based node representation to transfer knowledge across categories. By aligning graph embeddings with class-level semantic prototypes or external language features, the GIB-RSS framework could enable segmenting unseen object categories in remote sensing images. (3) Structural Robustness under Low-Resource Conditions. The adaptive masking mechanism in our method selectively retains the most informative nodes and edges during contrastive training. This selective learning process may inherently boost robustness under low-shot settings, where overfitting to limited data is a risk. Moreover, the information bottleneck objective discourages memorization of noise or irrelevant features, which further enhances generalization.

11. Conclusions

In this paper, we regard images as graph data and introduce GNN to perform remote sensing image segmentation tasks, which can flexibly model irregular objects. To extract the global contextual location information in the image, we introduce a multi-head attention mechanism for global information extraction. Furthermore, we introduce a feed-forward network for each node to perform feature transformation on node features to encourage information diversity. In addition, in order to accelerate the convergence speed of GNN, we introduce the information bottleneck theory for graph comparison learning. We argue that a good augmented view should be structurally heterogeneous but semantically similar. Experimental results prove the superiority of our model GIB-RSS.

CRediT authorship contribution statement

Yuntao Shou: Writing – original draft, Visualization, Validation, Software, Resources, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **Tao Meng:** Writing – review & editing, Validation, Supervision, Resources, Project administration, Methodology, Funding acquisition. **Wei Ai:** Writing – review & editing, Supervision, Resources, Project administration. **Haiyan Liu:** Writing – review & editing, Validation, Software, Methodology. **Keqin Li:** Writing – review & editing, Supervision, Software, Resources, Methodology.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work is supported by the National Natural Science Foundation of China (Grant No. 62372478), the Research Foundation of Education Bureau of Hunan Province of China (Grant No. 22B0275), the Hunan Provincial Natural Science Foundation General Project (Grant No. 2025JJ50380), and the Hunan Provincial Natural Science Foundation Youth Project (Grant No. 2025JJ60420).

Data availability

Data will be made available on request.

References

[1] L. Ding, D. Lin, S. Lin, J. Zhang, X. Cui, Y. Wang, H. Tang, L. Bruzzone, Looking outside the window: wide-context transformer for the semantic segmentation of high-resolution remote sensing images, IEEE Trans. Geosci. Remote Sens. 60 (2022) 1–13, https://doi.org/10.1109/TGRS.2022.3168697

[2] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, H. Adam, Encoder-decoder with atrous separable convolution for semantic image segmentation, in: Proceedings of the European Conference on Computer Vision (ECCV), 2018, pp. 801–818.

[3] C. Zhang, W. Jiang, Y. Zhang, W. Wang, Q. Zhao, C. Wang, Transformer and CNN hybrid deep neural network for semantic segmentation of very-high-resolution remote sensing imagery, IEEE Trans. Geosci. Remote Sens. 60 (2022) 1–20.

[4] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, et al., An image is worth 16×16 words: transformers for image recognition at scale, in: International Conference on Learning Representations, 2020.

[5] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, B. Guo, Swin transformer: hierarchical vision transformer using shifted windows, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2021, pp. 10012–10022.

[6] Y. Shou, T. Meng, W. Ai, C. Xie, H. Liu, Y. Wang, Object detection in medical images based on hierarchical transformer and mask mechanism, Comput. Intell. Neurosci. 2022 (2022) 1–12.

[7] K. Han, Y. Wang, J. Guo, Y. Tang, E. Wu, Vision GNN: an image is worth graph of nodes, Adv. Neural Inf. Process. Syst. 35 (2022) 8291–8303.

[8] N. Yin, L. Shen, M. Wang, X. Luo, Z. Luo, D. Tao, Omg: towards effective graph classification against label noise, IEEE Trans. Knowl. Data Eng. 35 (12) (2023) 12873–12886, https://doi.org/10.1109/TKDE.2023.3271677

[9] N. Yin, L. Shen, H. Xiong, B. Gu, C. Chen, X. Hua, S. Liu, X. Luo, Messages are never propagated alone: collaborative hypergraph neural network for time-series forecasting, IEEE Trans. Pattern Anal. Mach. Intell. (1) (5555) (2024) 1–15, https://doi.org/10.1109/TPAMI.2023.3331389

[10] N. Yin, L. Shen, B. Li, M. Wang, X. Luo, C. Chen, Z. Luo, X.-S. Hua, Deal: an unsupervised domain adaptive framework for graph-level classification, in: Proceedings of the 30th ACM International Conference on Multimedia, MM '22, Association for Computing Machinery, New York, NY, USA, 2022, pp. 3470–3479, https://doi.org/10.1145/3503161.3548012

[11] Y. Shou, T. Meng, W. Ai, S. Yang, K. Li, Conversational emotion recognition studies based on graph convolutional neural networks and a dependent syntactic analysis, Neurocomputing 501 (2022) 629–639.

[12] K. Hassani, A.H. Khasahmadi, Contrastive multi-view representation learning on graphs, in: International Conference on Machine Learning, PMLR, 2020, pp. 4116–4126.

[13] Y. You, T. Chen, Y. Sui, T. Chen, Z. Wang, Y. Shen, Graph contrastive learning with augmentations, Adv. Neural Inf. Process. Syst. 33 (2020) 5812–5823.

[14] F.-Y. Sun, J. Hoffman, V. Verma, J. Tang, Infograph: unsupervised and semi-supervised graph-level representation learning via mutual information maximization, in: International Conference on Learning Representations, 2019.

[15] T. Wu, H. Ren, P. Li, J. Leskovec, Graph information bottleneck, Adv. Neural Inf. Process. Syst. 33 (2020) 20437–20448.

[16] J. Long, E. Shelhamer, T. Darrell, Fully convolutional networks for semantic segmentation, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 3431–3440.

[17] S. Wu, J. Shi, Z. Chen, Hg-FCN: hierarchical grid fully convolutional network for fast VVC intra coding, IEEE Trans. Circuits Syst. Video Technol. 32 (8) (2022) 5638–5649.

[18] Z. Zhou, M.M. Rahman Siddiquee, N. Tajbakhsh, J. Liang, Unet + +: a nested U-Net architecture for medical image segmentation, in: Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support: 4th International Workshop, DLMIA 2018, and 8th International Workshop, ML-CDS 2018, Held in Conjunction with MICCAI 2018, Granada, Spain, September 20, 2018, Proceedings 4, Springer, 2018, pp. 3–11.

[19] H. Cao, Y. Wang, J. Chen, D. Jiang, X. Zhang, Q. Tian, M. Wang, Swin-UNet: UNet-like pure transformer for medical image segmentation, in: European Conference on Computer Vision, Springer, 2022, pp. 205–218.

[20] S. Liu, Y. Ma, X. Zhang, H. Wang, J. Ji, X. Sun, R. Ji, Rotated multi-scale interaction network for referring remote sensing image segmentation, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2024, pp. 26658–26668.

[21] C. Liang-Chieh, G. Papandreou, I. Kokkinos, K. Murphy, A. Yuille, Semantic image segmentation with deep convolutional nets and fully connected CRFs, in: International Conference on Learning Representations, 2015.

[22] J. Wang, K. Sun, T. Cheng, B. Jiang, C. Deng, Y. Zhao, D. Liu, Y. Mu, M. Tan, X. Wang, et al., Deep high-resolution representation learning for visual recognition, IEEE Trans. Pattern Anal. Mach. Intell. 43 (10) (2020) 3349–3364.

[23] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A.N. Gomez, Ł. Kaiser, I. Polosukhin, Attention is all you need, Adv. Neural Inf. Process. Syst. 30 (2017).

[24] D. Wang, J. Zhang, B. Du, M. Xu, L. Liu, D. Tao, L. Zhang, Samrs: scaling-up remote sensing segmentation dataset with segment anything model, Adv. Neural Inf. Process. Syst. 36 (2023) 8815–8827.

[25] T.N. Kipf, M. Welling, Semi-supervised classification with graph convolutional networks, arXiv preprint arXiv:1609.02907, 2016.

[26] Y. Mo, L. Peng, J. Xu, X. Shi, X. Zhu, Simple unsupervised graph representation learning, in: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 36, 2022, pp. 7797–7805.

[27] Z. Wen, Y. Fang, Trend: temporal event and node dynamics for graph representation learning, in: Proceedings of the ACM Web Conference 2022, 2022, pp. 1159–1169.

[28] X. Hao, J. Li, Y. Guo, T. Jiang, M. Yu, Hypergraph neural network for skeleton-based action recognition, IEEE Trans. Image Process. 30 (2021) 2263–2275.

[29] W. Shi, R. Rajkumar, Point-GNN: graph neural network for 3D object detection in a point cloud, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 1711–1719.

[30] S. Saha, S. Zhao, X.X. Zhu, Multitarget domain adaptation for remote sensing classification using graph neural network, IEEE Geosci. Remote Sens. Lett. 19 (2022) 1–5, https://doi.org/10.1109/LGRS.2022.3149950

[31] P. Veličković, W. Fedus, W.L. Hamilton, P. Liò, Y. Bengio, R.D. Hjelm, Deep graph infomax, in: International Conference on Learning Representations, 2018.

[32] E. Jang, S. Gu, B. Poole, Categorical reparameterization with Gumbel-Softmax, in: International Conference on Learning Representations, 2017.

[33] Y. Lyu, G. Vosselman, G.-S. Xia, A. Yilmaz, M.Y. Yang, Uavid: a semantic segmentation dataset for UAV imagery, ISPRS J. Photogram. Remote Sens. 165 (2020) 108–119.

[34] H. Li, K. Qiu, L. Chen, X. Mei, L. Hong, C. Tao, Scattnet: semantic segmentation network with spatial and channel attention mechanism for high-resolution remote sensing images, IEEE Geosci. Remote Sens. Lett. 18 (5) (2020) 905–909.

[35] A. Boguszewski, D. Batorski, N. Ziemba-Jankowska, T. Dziedzic, A. Zambrzycka, Landcover.AI: dataset for automatic mapping of buildings, woodlands, water and roads from aerial imagery, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021, pp. 1102–1110.

[36] M.Y. Yang, S. Kumaar, Y. Lyu, F. Nex, Real-time semantic segmentation with context aggregation network, ISPRS J. Photogram. Remote Sens. 178 (2021) 124–134.

[37] J. Fu, J. Liu, H. Tian, Y. Li, Y. Bao, Z. Fang, H. Lu, Dual attention network for scene segmentation, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019, pp. 3146–3154.

[38] M. Oršić, S. Šegvić, Efficient semantic segmentation with pyramidal fusion, Pattern Recognit. 110 (2021) 107611.

[39] C. Yu, J. Wang, C. Peng, C. Gao, G. Yu, N. Sang, Bisenet: bilateral segmentation network for real-time semantic segmentation, in: Proceedings of the European Conference on Computer Vision (ECCV), 2018, pp. 325–341.

[40] R. Li, S. Zheng, C. Zhang, C. Duan, J. Su, L. Wang, P.M. Atkinson, Multiattention network for semantic segmentation of fine-resolution remote sensing images, IEEE Trans. Geosci. Remote Sens. 60 (2021) 1–13.

[41] R. Li, S. Zheng, C. Zhang, C. Duan, L. Wang, P.M. Atkinson, Abcnet: attentive bilateral contextual network for efficient semantic segmentation of fine-resolution remotely sensed imagery, ISPRS J. Photogram. Remote Sens. 181 (2021) 84–98.

[42] R. Strudel, R. Garcia, I. Laptev, C. Schmid, Segmenter: transformer for semantic segmentation, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2021, pp. 7262–7272.

[43] E. Xie, W. Wang, Z. Yu, A. Anandkumar, J.M. Alvarez, P. Luo, Segformer: simple and efficient design for semantic segmentation with transformers, Adv. Neural Inf. Process. Syst. 34 (2021) 12077–12090.

[44] L. Wang, R. Li, D. Wang, C. Duan, T. Wang, X. Meng, Transformer meets convolution: a bilateral awareness network for semantic segmentation of very fine resolution urban scene images, Remote Sens. 13 (16) (2021) 3065.

[45] A. Srinivas, T.-Y. Lin, N. Parmar, J. Shlens, P. Abbeel, A. Vaswani, Bottleneck transformers for visual recognition, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021, pp. 16519–16529.

[46] J. Chen, Y. Lu, Q. Yu, X. Luo, E. Adeli, Y. Wang, L. Lu, A.L. Yuille, Y. Zhou, Transunet: transformers make strong encoders for medical image segmentation, arXiv preprint arXiv:2102.04306, 2021.

[47] J. Zhuang, J. Yang, L. Gu, N. Dvornek, Shelfnet for fast semantic segmentation, in: Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops, 2019.

[48] W. Xu, Y. Xu, T. Chang, Z. Tu, Co-scale conv-attentional image transformers, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2021, pp. 9981–9990.

[49] L. Wang, R. Li, C. Zhang, S. Fang, C. Duan, X. Meng, P.M. Atkinson, Unetformer: a UNet-like transformer for efficient semantic segmentation of remote sensing urban scene imagery, ISPRS J. Photogram. Remote Sens. 190 (2022) 196–214.

[50] D. Zhang, H. Li, D. He, N. Liu, L. Cheng, J. Wang, J. Han, Unsupervised pre-training with language-vision prompts for low-data instance segmentation, IEEE Trans. Pattern Anal. Mach. Intell. 47 (10) (2025) 8642–8657.

[51] P. Huang, D. Zhang, D. Cheng, L. Han, P. Zhu, J. Han, M-rrfs: a memory-based robust region feature synthesizer for zero-shot object detection, Int. J. Comput. Vis. 132 (10) (2024) 4651–4672.

Author biography

Yuntao Shou received the B.S. degree in School of Computer and Information Engineering, Central South University of Forestry and Technology, Changsha, China, in 2023. He is currently pursuing the graduation degree with Xi'an Jiaotong University, Xian, China. His research interests include emotion recognition and graph representation learning.

Tao Meng received the Ph.D. degree in the College of Computer Science and Electronic Engineering, Hunan University, Changsha, China. His research interests include data mining, artificial intelligence, machine learning, natural language processing, graph and network analysis.

Wei Ai received the Ph.D. degree in the College of Computer Science and Electronic Engineering, Hunan University, Changsha, China. Her research interests include date mining, big data, cloud computing, and parallel computing.

Haiyan Liu holds an associate professor at Changsha Medical University and deputy director of the Hunan Provincial University Key Laboratory of the Fundamental and Clinical Research on Functional Nucleic Acid. Her primary research interests encompass machine learning and biological information computing.

Keqin Li is a SUNY Distinguished Professor of Computer Science with the State University of New York. He is also a National Distinguished Professor with Hunan University, China. His current research interests include cloud computing, fog computing and mobile edge computing, energy-efficient computing and communication, embedded systems and cyber-physical systems, heterogeneous computing systems, big data computing, high-performance computing, CPU-GPU hybrid and cooperative computing, computer architectures and systems, computer networking, machine learning, intelligent and soft computing. He has authored or coauthored over 890 journal articles, book chapters, and refereed conference papers, and has received several best paper awards. He holds nearly 70 patents announced or authorized by the Chinese National Intellectual Property Administration. He is among the world's top 5 most influential scientists in parallel and distributed computing in terms of both single-year impact and career-long impact based on a composite indicator of Scopus citation database. He has chaired many international conferences. He is currently an associate editor of the ACM Computing Surveys and the CCF Transactions on High Performance Computing. He has served on the editorial boards of the IEEE Transactions on Parallel and Distributed Systems, the IEEE Transactions on Computers, the IEEE Transactions on Cloud Computing, the IEEE Transactions on Services Computing, and the IEEE Transactions on Sustainable Computing. He is an AAAS Fellow, an IEEE Fellow, and an AAIA Fellow. He is also a Member of Academia Europaea (Academician of the Academy of Europe).