remote sensing

MDPI

Article

SSGTN: Spectral–Spatial Graph Transformer Network for Hyperspectral Image Classification

Haotian Shi [1], Zihang Luo [1], Yiyang Ma [2], Guanquan Zhu [3] and Xin Dai [1,*]

[1] School of Electronic and Communication Engineering, Guangzhou University, Guangzhou 510182, China; shihaotian@gzhu.edu.cn (H.S.)
[2] School of Mechanical and Electrical Engineering, Guangzhou University, Guangzhou 510182, China
[3] School of Mathematics and Information Science, Guangzhou University, Guangzhou 510006, China
* Correspondence: xindai@gzhu.edu.cn

Highlights

What are the main findings?

- We propose a Spectral–Spatial Graph Transformer Network (SSGTN), a dual-branch framework for hyperspectral image classification that combines local feature extraction with global context reasoning.
- The framework builds region-level graphs from superpixels, applies lightweight spectral denoising, and introduces a parameter-free Spectral–Spatial Shift Module (SSSM) to strengthen spectral–spatial feature interaction.

What are the implications of the main findings?

- With only 1% training samples, the proposed method achieves state-of-the-art performance on three benchmark datasets (Indian Pines, WHU-Hi-LongKou, and Houston2018).
- The results suggest that combining region-level structural modeling with global reasoning is an effective and efficient strategy for hyperspectral remote sensing under scarce labels, and may benefit other spectral–spatial learning problems.

Abstract

Hyperspectral image (HSI) classification is fundamental to a wide range of remote sensing applications, such as precision agriculture, environmental monitoring, and urban planning, because HSIs provide rich spectral signatures that enable the discrimination of subtle material differences. Deep learning approaches, including Convolutional Neural Networks (CNNs), Graph Convolutional Networks (GCNs), and Transformers, have achieved strong performance in learning spatial–spectral representations. However, these models often face difficulties in jointly modeling long-range dependencies, fine-grained local structures, and non-Euclidean spatial relationships, particularly when labeled training data are scarce. This paper proposes a **Spectral–Spatial Graph Transformer Network (SSGTN),** a dual-branch architecture that integrates superpixel-based graph modeling with Transformer-based global reasoning. SSGTN consists of four key components, namely (1) an **LDA-SLIC superpixel graph construction module** that preserves discriminative spectral–spatial structures while reducing computational complexity, (2) a lightweight **spectral denoising module** based on 1×1 convolutions and batch normalization to suppress redundant and noisy bands, (3) a **Spectral–Spatial Shift Module (SSSM)** that enables efficient multi-scale feature fusion through channel-wise and spatial-wise shift operations, and (4) a **dual-branch GCN-Transformer block** that jointly models local graph topology and global spectral–spatial dependencies. Extensive experiments on three public HSI datasets (Indian Pines, WHU-Hi-LongKou, and Houston2018) under limited supervision (1% training samples) demonstrate

Check for updates

Academic Editors: Zhigang Tu, Zhitong Xiong, Hao Sun and Le Dong

Received: 1 December 2025
Revised: 31 December 2025
Accepted: 5 January 2026
Published: 7 January 2026

Copyright: © 2026 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license.

that SSGTN consistently outperforms state-of-the-art CNN-, Transformer-, Mamba-, and GCN-based methods in overall accuracy, Average Accuracy, and the κ coefficient. The proposed framework provides an effective baseline for HSI classification under limited supervision and highlights the benefits of integrating graph-based structural priors with global contextual modeling.

Keywords: hyperspectral image classification; deep learning; graph convolutional networks; Transformers

1. Introduction

Hyperspectral images (HSIs) capture the continuous reflectance spectrum of surface materials across hundreds of narrow and contiguous spectral bands. As a result, a HSI can be represented as a three-dimensional data cube that combines spatial information ($H \times W$ pixels) with rich spectral signatures (B bands). This dense spectral resolution enables the discrimination of subtle material differences that are difficult to observe using conventional RGB or multispectral sensors. Therefore, HSIs are valuable for precision agriculture [1], environmental monitoring [2], mineral exploration [3], and urban studies [4]. Hyperspectral image classification (HSIC) is a central task in these applications, in which each pixel is assigned a semantic land-cover or material label to support large-scale mapping and automated decision support.

Early HSIC methods largely relied on handcrafted feature engineering and shallow classifiers. Techniques such as band selection [5], spectral derivatives [6], and linear dimensionality reduction (e.g., PCA and LDA) were commonly used to alleviate spectral redundancy and the curse of dimensionality. Spatial context was often incorporated through morphological profiles [7] or heuristic filtering. Although these approaches are interpretable, they have limited adaptability to complex nonlinear spectral mixing patterns, depend heavily on domain expertise, and often generalize poorly across diverse scenes [8]. These limitations are particularly pronounced in high-dimensional, spatially heterogeneous, and label-scarce environments. Beyond classification-oriented pipelines, non-deep-learning hyperspectral image analysis has also explored noise-aware weighting and outlier removal to improve the robustness of spectral–spatial criteria for object-based processing and scale selection [9].

The advent of deep learning has dramatically reshaped the HSIC landscape. Convolutional Neural Networks (CNNs) have become a dominant paradigm in this area. In particular, 2D CNNs [10] extract spatial textures from spectral bands, and 3D CNNs [11] jointly model spectral–spatial dependencies. Hybrid architectures such as HybridSN [12] further improve efficiency by combining 2D and 3D convolutions. Despite their success, CNNs are limited by local receptive fields and fixed grid processing, which restrict their ability to capture long-range dependencies and adapt to irregular object boundaries. To mitigate these limitations, *Transformers* have been introduced and use self-attention to model global contextual relationships across both spatial and spectral dimensions [13,14]. More recently, *state–space models (SSMs)* such as Mamba [15] have emerged as efficient alternatives to Transformers by offering linear complexity with global receptive fields. However, these sequence-based models can be sensitive to spectral noise, may not preserve fine-grained local structures, and do not explicitly model irregular spatial relationships.

In parallel, *graph neural networks (GNNs)* have gained traction due to their ability to model non-Euclidean relationships among pixels or superpixels. Early graph convolutional networks (GCNs) [16] demonstrated promising results in semi-supervised HSIC by propagating node features over adjacency graphs. Subsequent efforts introduced multi-

scale GCNs [17], cross-attention GCNs [18], and object-based graph constructions [19] to enhance feature aggregation and boundary preservation. More recently, hybrid graph state–space models such as Graph Mamba [20] have bridged graph structural learning with sequence modeling. Nevertheless, graph-based approaches still face several fundamental challenges. First, graph convolutions primarily capture local connectivity and may fail to model long-range dependencies effectively. Second, graph construction is often heuristic and scene-dependent. Third, many models lack dynamic multi-scale fusion mechanisms and do not fully integrate spatial and spectral cues. Finally, spectral noise and redundancy can degrade input quality and reduce robustness.

To address these limitations, we propose the **Spectral–Spatial Graph Transformer Network (SSGTN)**, a unified dual-branch architecture that integrates graph-based structural modeling with Transformer-based global reasoning. The proposed framework includes four key components. First, an **LDA-SLIC superpixel graph construction module** combines linear discriminant analysis (LDA) for spectral compaction with Simple Linear Iterative Clustering (SLIC) for spatially homogeneous region segmentation to obtain a structurally informed and computationally efficient graph representation. Second, a lightweight **spectral denoising module** based on 1×1 convolutions and batch normalization suppresses redundant and noisy spectral bands while preserving discriminative features. Third, a **Spectral–Spatial Shift Module (SSSM)** performs cyclic shifts along spectral, height, and width dimensions to enable efficient multi-scale feature interaction without introducing additional parameters. Fourth, a **dual-branch GCN-Transformer block** jointly models local graph topology and global dependencies, where a spatial Transformer guided by GCNs captures long-range spatial information and a spectral Transformer models cross-band correlations; the two branches are fused through a residual graph convolution.

The main contributions of this work are summarized as follows:

1. We propose a novel **dual-branch graph–Transformer hybrid architecture** that jointly models local graph structures and global spectral–spatial dependencies, effectively overcoming the limitations of conventional single-paradigm models.
2. We design a **dynamic Spectral–Spatial Shift Module** that enables efficient multi-dimensional feature fusion through parameter-free shift operations, enhancing the model's ability to capture contextual interactions across scales.
3. We develop a **superpixel-driven graph construction strategy** using LDA-SLIC, which adaptively captures spatial homogeneity and spectral discriminability while maintaining computational efficiency via sparse graph representations.
4. We introduce a **spectral denoising module** that refines input representations through lightweight convolutions and normalization, improving robustness to spectral noise and redundancy.
5. We conduct comprehensive experiments and ablation studies across multiple datasets and training regimes, validating the superiority, generality, and interpretability of SSGTN in HSI classification under limited supervision.

The remainder of this paper is organized as follows. Section 2 introduces related work in hyperspectral remote sensing image classification. Section 3 presents the proposed SSGTN architecture. Section 4 reports experimental results on three benchmark hyperspectral datasets. Finally, Section 6 concludes the paper and outlines future research directions.

2. Related Work

In this section, we systematically review the evolution of deep learning-based hyperspectral image classification (HSIC) methods, which can be broadly categorized into convolutional, attention-based, and graph-based approaches. We highlight the strengths

and limitations of each paradigm, paving the way to introduce our proposed Spectral–Spatial Graph Transformer Network (SSGTN).

2.1. CNN-Based Hyperspectral Image Classification Methods

Convolutional Neural Networks have become a cornerstone in HSIC due to their strong ability to extract spatially structured features [12,21–29]. Early work by Hu et al. [10] demonstrated that 2D CNNs can effectively leverage local spatial textures within HSI patches, significantly improving classification accuracy over purely spectral methods. To better model the spectral–spatial dependencies inherent in HSIs, Li et al. [11] extended CNNs to three dimensions and proposed 3D CNNs that jointly process spectral cubes. Further innovations led to hybrid architectures, such as the synergistic 2D/3D CNN by Yang et al. [30], which integrates spectral–spatial fusion through 3D convolutions and uses complementary 2D spatial context modeling to balance accuracy and computational efficiency. Overall, these developments reflect a progression from purely spatial 2D CNNs to more advanced 3D and hybrid architectures for comprehensive spectral–spatial integration.

Despite these advances, CNN-based methods exhibit several intrinsic limitations. Standard 2D-CNNs often disrupt spectral continuity by treating bands independently, leading to potential misclassification of spectrally similar materials. While 3D-CNNs can preserve spectral–spatial coherence, they dramatically increase model size and computational burden, creating scalability issues for high-dimensional HSIs. Moreover, the fixed receptive fields and inherently local inductive biases of convolutional kernels restrict their ability to capture long-range dependencies and multi-scale contextual information. These limitations hinder the generalization of CNN-based models in heterogeneous environments and motivate the exploration of more flexible architectures beyond convolution.

2.2. Attention-Based Hyperspectral Image Classification Methods

To overcome the locality bias of convolutions, attention-based architectures have been introduced into HSIC to model long-range spatial–spectral dependencies [13,15,31–38]. Representative examples include Transformers and, more recently, state–space models such as Mamba. For instance, Hong et al. [14] proposed *SpectralFormer* to strengthen inter-band relationships via self-attention, yielding competitive gains over convolutional baselines. Gu et al. [39] designed a multi-scale lightweight Transformer to reduce computational cost while preserving global modeling capacity. On the state–space side, He et al. [15] introduced *3DSS-Mamba*, which organizes spectral–spatial tokens for efficient long-range dependency modeling. *CenterMamba* [38] adopts a center-scan strategy to enhance semantic representation with linear-complexity sequence processing. These designs provide two complementary approaches to scalable global spatial–spectral representation learning in HSIC.

Notwithstanding their progress, attention-based approaches still face several limitations. First, Transformer models can be computationally demanding and may struggle to reconcile global dependency modeling with fine-grained local detail, especially under high spectral dimensionality and limited labels. Second, many Transformer pipelines rely on fixed tokenization or single-scale processing, leading to insufficient dynamic multi-scale adaptation across heterogeneous scenes. Third, both Transformers and Mamba variants can be sensitive to spectral redundancy and noise, benefiting from explicit denoising or channel re-weighting to stabilize training. Finally, while Mamba/SSM models offer efficiency gains, they may suffer from slow convergence and hyper-parameter sensitivity, and by design, they do not explicitly account for irregular spatial relations. These shortcomings have spurred increasing interest in graph-based architectures, which provide a more flexible representation for non-Euclidean spatial–spectral structures.

2.3. Graph-Based Hyperspectral Image Classification Methods

Graph-based methods have recently emerged as powerful tools for HSIC because they can represent spatial–spectral relations on irregular and non-Euclidean domains [16,19,40–47]. Early studies demonstrated that graph convolutional networks can capture contextual dependencies through message passing over pixels or superpixels [16]. Subsequent advances introduced more adaptive designs. For example, Wan et al. [17] proposed a multi-scale dynamic GCN that aggregates information across spatial neighborhoods, while Yang et al. [18] introduced a cross-attention-driven spatial–spectral GCN to better integrate heterogeneous features. More recently, object-based strategies such as MOB-GCN [19] have further emphasized multi-scale structural cues, improving boundary delineation and robustness to noise.

Building on these advances, researchers have extended attention mechanisms to graph formulations. Zheng et al. [48] proposed a graph Transformer that fuses spatial–spectral features via self-attention to enhance long-range dependency modeling. In parallel, Ahmad et al. [20] introduced a hybrid Graph Mamba model that tokenizes hyperspectral data into graph representations and leverages state–space modeling to balance efficiency and global context capture.

Although graph-based methods have significantly advanced hyperspectral image classification, they remain constrained by several factors. First, neighborhood aggregation in graph convolutions primarily captures local connectivity, limiting the capture of complex long-range dependencies. Second, graph construction is often heuristic and scene-dependent, reducing adaptability across diverse scenes. Third, most models process features at fixed scales, hindering their adaptability from heterogeneous spatial–spectral patterns. Fourth, spatial and spectral cues are not always effectively integrated, leading to suboptimal joint representations. Finally, redundant or noisy bands degrade input quality and reduce classification robustness, particularly under scarce supervision. These challenges highlight the need for a more integrated approach that combines the strengths of graph structural learning with dynamic multi-scale fusion and global dependency modeling.

The proposed SSGTN is designed to address the aforementioned limitations in a unified framework. Unlike CNNs, SSGTN captures long-range dependencies via Transformer blocks while preserving local structure through graph convolutions. In contrast to pure Transformers, it incorporates an LDA-SLIC superpixel graph to model non-Euclidean spatial relationships and employs a spectral denoising module to enhance input representations. Compared to existing graph-based methods, SSGTN introduces a novel Spectral–Spatial Shift Module for dynamic multi-scale feature fusion and a dual-branch GCN-Transformer architecture to jointly model local topology and global dependencies. By synergistically integrating adaptive graph priors, spectral purification, shift-based feature interaction, and Transformer-based global reasoning, SSGTN achieves expressive and efficient hyperspectral representation learning under high-dimensional and structurally complex conditions, particularly under limited supervision.

3. Materials and Methods

The overall architecture of the proposed SSGTN is depicted in Figure 1. This hybrid framework synergistically integrates convolutional operations for local feature extraction, graph convolutions for topological modeling, and Transformer blocks for global dependency learning. The network comprises four meticulously designed components: (1) LDA-SLIC superpixel segmentation with graph construction (inspired by the superpixel-based graph modeling paradigm in CEGCN [42]), (2) spectral denoising module, (3) Spectral–Spatial Shift Module, and (4) dual-branch spectral–spatial GCN-Transformer module. Each

component addresses specific challenges in hyperspectral image classification while maintaining computational efficiency.

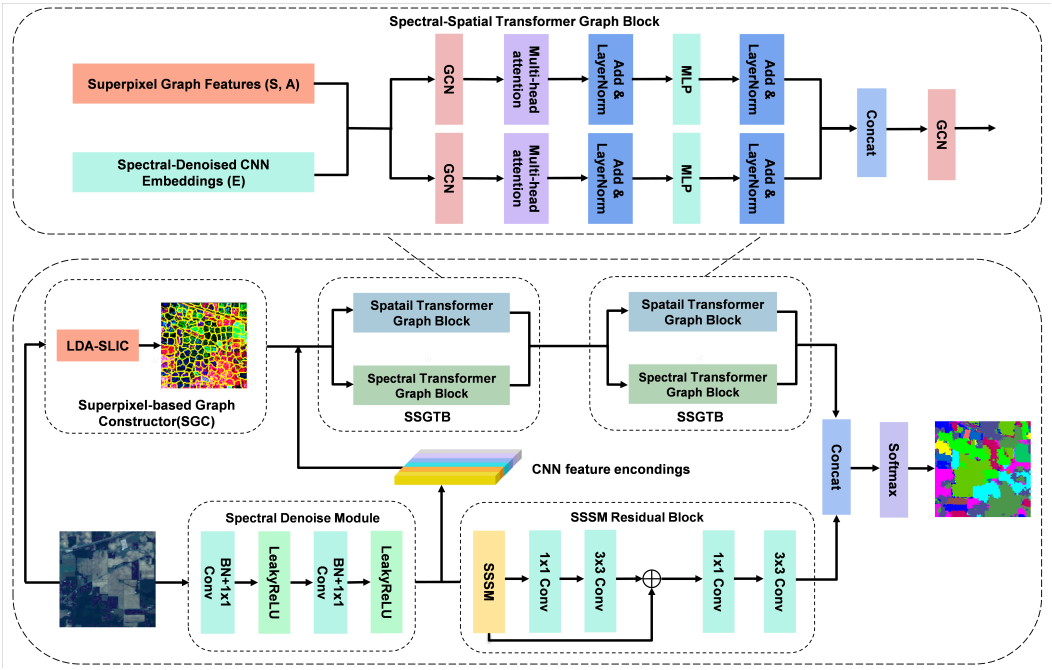


Figure 1. Overall architecture of the proposed SSGTN. The framework integrates four key components: (1) LDA-SLIC superpixel segmentation for graph construction, (2) spectral denoising module for noise suppression, (3) Spectral–Spatial Shift Module for multi-scale feature fusion, and (4) dual-branch GCN-Transformer module for joint local and global dependency modeling.

3.1. LDA-SLIC Superpixel Segmentation Module

The LDA-SLIC module integrates Linear Discriminant Analysis (LDA) for spectral dimensionality reduction with Simple Linear Iterative Clustering (SLIC) for spatial superpixel segmentation, providing a compact, class-discriminative representation while producing spatially homogeneous regions that serve as graph nodes.

3.1.1. LDA-Based Spectral Dimensionality Reduction

Given an input hyperspectral image $\mathbf{X} \in \mathbb{R}^{H \times W \times B}$ with sparse supervision $\mathbf{Y} \in \mathbb{R}^{H \times W}$, LDA projects the spectral data into a lower-dimensional subspace by maximizing inter-class separability:

$$\mathbf{X}_{\mathrm{LDA}} = \mathbf{X}\mathbf{W}_{\mathrm{LDA}} \in \mathbb{R}^{H \times W \times C}, \tag{1}$$

where $\mathbf{W}_{\mathrm{LDA}} \in \mathbb{R}^{B \times C}$ denotes the projection matrix learned by maximizing the Fisher criterion [49] $J(\mathbf{W}) = \frac{\mathbf{W}^T \mathbf{S}_B \mathbf{W}}{\mathbf{W}^T \mathbf{S}_W \mathbf{W}}$, with $\mathbf{S}_B$ and $\mathbf{S}_W$ representing between-class and within-class scatter matrices, respectively. In practice, LDA is fitted only on the labeled training pixels (1% of the image in our low-label setting), while all unlabeled, validation, and test pixels are treated as background and excluded from the optimization, preventing any information leakage from the test set. The resulting projection reduces the spectral dimensionality from B to at most, $C - 1$ class-discriminative components, which are then applied to the entire cube. Because LDA is a shallow linear transformation and the main model capacity resides in the subsequent CNN-GCN-Transformer modules, this supervised preprocessing acts as a light-weight spectral pre-conditioning step rather than a deep classifier, empirically yielding stable performance across different random training splits even under limited supervision.

3.1.2. SLIC Superpixel Segmentation

The dimension-reduced representation $\mathbf{X}_{\mathrm{LDA}}$ is subsequently partitioned into superpixels using the Simple Linear Iterative Clustering (SLIC) algorithm [50]. SLIC operates by iteratively optimizing a composite distance metric in the joint spectral–spatial domain:

$$D = \sqrt{\left\|\mathbf{x}^{(i)}_{\mathrm{spectral}} - \mathbf{x}^{(j)}_{\mathrm{spectral}}\right\|_2^2 + \lambda \left\|\mathbf{x}^{(i)}_{\mathrm{spatial}} - \mathbf{x}^{(j)}_{\mathrm{spatial}}\right\|_2^2}, \tag{2}$$

where $\lambda = (m/S)^2$ controls the trade-off between spectral similarity and spatial proximity, m is the compactness parameter, and $S = \sqrt{HW/K}$ is the nominal grid interval associated with the target number of superpixels K. The scale (or equivalently K) determines the expected number of pixels per superpixel and hence the granularity of the graph, while the compactness parameter governs whether regions adhere more closely to spectral boundaries (small m) or favor smoother, more spatially regular shapes (large m). We deliberately choose a moderately fine scale and balanced compactness so that mixed pixels and small objects are not overly merged, and each superpixel remains approximately homogeneous in the LDA space. It is important to note that the superpixels are used to define region-level nodes and the sparsity pattern of the graph, but final predictions are produced at the pixel level by fusing the graph branch with a parallel CNN branch. In particular, the CNN and SSSM modules operate directly on the full-resolution pixel grid and preserve fine-grained local details, while the dual-branch GCN-Transformer stack captures long-range and nonlinear spectral–spatial dependencies on the superpixel graph. As a result, LDA-SLIC provides a stable structural prior that is complemented and refined by deeper, nonlinear feature learning in the downstream network.

3.1.3. Graph Representation Learning

To capture region-level interactions while keeping computation tractable, we construct a superpixel graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ where each node v_i corresponds to a superpixel region $\mathcal{R}_i$, and edges are defined only between spatially adjacent regions. The node feature matrix $\mathbf{S} \in \mathbb{R}^{K \times C}$ is computed by averaging the LDA-projected hyperspectral features within each region. This step reduces the spectral dimensionality to a compact, class-discriminative space and substantially lowers the memory requirements of subsequent graph operations. Based on these region features, we construct an initial adjacency matrix using a Gaussian kernel constrained by spatial neighborhood:

$$\mathbf{A}_{i,j} = \exp\left(-\gamma \left\|\mathbf{S}_i - \mathbf{S}_j\right\|_2^2\right) \cdot \mathbb{I}[j \in \mathcal{N}(i)]. \tag{3}$$

Since each superpixel interacts only with its directly bordering regions, the resulting adjacency is naturally sparse, and the number of non-zero entries scales linearly with the number of superpixels. Graph operations therefore scale with $\mathcal{O}(K\tilde{d})$ rather than $\mathcal{O}(K^2)$, where $\tilde{d}$ denotes the small node degree induced by the superpixel topology.

The kernel bandwidth is determined in a simple data-dependent way: σ is estimated as the median distance between each superpixel and its spatial neighbors, yielding $\gamma = 1/\sigma^2$. In practice, the Gaussian-weighted adjacency serves as a structural prior and sparsity pattern rather than the final propagation operator. Within the GCN layer, we refine the edge weights by first projecting batch-normalized node features through a linear mapping to obtain feature embeddings, computing a learned affinity matrix via a sigmoid of the pairwise inner products, and then applying the sparsity mask from $\mathbf{A}$. A row-wise softmax yields a stochastic propagation matrix $\widetilde{\mathbf{A}}$ that preserves the superpixel topology while allowing data-driven adjustment of edge strengths.

3.2. Spectral Denoising Module

The spectral denoising module is designed as a lightweight feature refinement cascade to suppress spectral noise while preserving discriminative band correlations. Given an input hyperspectral cube $\mathbf{X} \in \mathbb{R}^{H\times W\times B}$, the module applies two sequential stages, each consisting of batch normalization (BN) and 1×1 convolution for spectral filtering and dimensionality reduction:

$$\mathbf{F}_1 = \mathrm{Conv}_{1\times1}(\mathrm{BN}(\mathbf{X})), \quad \mathbf{X}_{\mathrm{denoised}} = \mathrm{Conv}_{1\times1}(\mathrm{BN}(\mathbf{F}_1)), \tag{4}$$

where $\mathrm{BN}(\cdot)$ denotes channel-wise batch normalization with learnable affine parameters, and $\mathrm{Conv}_{1\times1}(\cdot)$ represents a 1×1 convolution that reduces spectral dimensionality ($B \rightarrow D$) while acting as an adaptive spectral filter. This design effectively mitigates spectral noise and redundancy while maintaining computational efficiency.

3.3. SSSM Residual Convolution Module

The SSSM module (as shown in Figure 2) enables efficient multi-dimensional feature interaction through parameter-free shift operations. For an input tensor $\mathbf{X} \in \mathbb{R}^{H\times W\times B}$, we define cyclic shift operators along three dimensions:

$$\mathrm{Shift}_B(\mathbf{X})_{h,w,b} = \mathbf{X}_{h,w,(b+1) \bmod B}, \tag{5}$$

$$\mathrm{Shift}_H(\mathbf{X})_{h,w,b} = \mathbf{X}_{(h+1) \bmod H,w,b}, \tag{6}$$

$$\mathrm{Shift}_W(\mathbf{X})_{h,w,b} = \mathbf{X}_{h,(w+1) \bmod W,b}. \tag{7}$$

These shifted features are concatenated with the original input along the channel dimension:

$$\mathbf{X}_{\mathrm{concat}} = \mathrm{Concat}(\mathbf{X}, \mathrm{Shift}_B(\mathbf{X}), \mathrm{Shift}_H(\mathbf{X}), \mathrm{Shift}_W(\mathbf{X})), \tag{8}$$

$$\mathbf{X}_2 = \mathrm{Conv}_{3\times3}(\mathrm{Conv}_{1\times1}(\mathbf{X}_{\mathrm{concat}})), \tag{9}$$

The concatenated features are then processed by a 1×1 convolution followed by a 3×3 convolution for effective feature fusion and extraction, producing $\mathbf{X}_2$. A residual connection is employed to enhance representation capacity and alleviate gradient vanishing as follows:

$$\mathbf{Y} = \mathbf{X}_2 + \mathbf{X}. \tag{10}$$

This design facilitates comprehensive spectral–spatial interaction without introducing additional parameters, making it computationally efficient for high-dimensional HSI data.

3.4. Dual-Branch Spectral–Spatial GCN-Transformer

The dual-branch module constitutes the core of SSGTN (as shown in Figure 3), designed to jointly model local graph structures and global dependencies. We initialize the node embeddings as $\mathbf{X}^{(0)} = \mathbf{S} + \tilde{\mathbf{E}}$, where $\tilde{\mathbf{E}}$ denotes learnable positional encodings. The superpixel-level features $\mathbf{S}$ are obtained by aggregating pixel representations after the spectral denoising and SSSM modules using the normalized assignment matrix $\mathbf{Q}$:

$$\mathbf{S} = \mathbf{Q}^\top \mathbf{X}_{\mathrm{pix}}, \tag{11}$$

which averages pixel features within each superpixel to form node descriptors for the dual-branch graph–Transformer module, as shown in Algorithm 1.

Algorithm 1 Dual-Branch Spectral–Spatial GCN-Transformer (SSGTB)

Require: Superpixel node features $\mathbf{S}$; adjacency matrix $\mathbf{A}$; positional encodings $\tilde{\mathbf{E}}$; layer number L; dropout rate p
Ensure: Final node features $\mathbf{Z}_g$

$\mathbf{X}^{(0)} \leftarrow \mathbf{S} + \tilde{\mathbf{E}}$ ▷ $\mathbf{X}^{(0)} = \mathbf{S} + \tilde{\mathbf{E}}$
for $\ell = 1$ **to** L **do**
  $\mathbf{H}_{\text{spa}}^{(\ell)} \leftarrow \sigma(\hat{\mathbf{A}}\mathbf{X}^{(\ell-1)}\mathbf{W}_{\text{gcn}})$ ▷ $\mathbf{H}_{\text{spa}}^{(\ell)} = \sigma(\hat{\mathbf{A}}\mathbf{X}^{(\ell-1)}\mathbf{W}_{\text{gcn}})$
  $\mathbf{Z}_{\text{spa}}^{(\ell)} \leftarrow \text{LN}\Big(\mathbf{X}^{(\ell-1)} + \text{MHSA}(\mathbf{H}_{\text{spa}}^{(\ell)})\Big)$
  $\mathbf{H}_{\text{spa}}^{(\ell)} \leftarrow \text{LN}\Big(\mathbf{Z}_{\text{spa}}^{(\ell)} + \text{FFN}(\mathbf{Z}_{\text{spa}}^{(\ell)})\Big)$
  $\mathbf{H}_{\text{spec}}^{(\ell)} \leftarrow \text{GCN}(\mathbf{X}^{(\ell-1)}, \mathbf{A})$
  $\mathbf{Z}_{\text{spec}}^{(\ell)} \leftarrow \text{LN}\Big(\mathbf{X}^{(\ell-1)} + \text{MHSA}(\mathbf{H}_{\text{spec}}^{(\ell)})\Big)$
  $\mathbf{H}_{\text{spec}}^{(\ell)} \leftarrow \text{LN}\Big(\mathbf{Z}_{\text{spec}}^{(\ell)} + \text{FFN}(\mathbf{Z}_{\text{spec}}^{(\ell)})\Big)$
  $\mathbf{U}^{(\ell)} \leftarrow \text{Dropout}_p\Big(\mathbf{W}_f^{(\ell)}\big[\mathbf{H}_{\text{spa}}^{(\ell)} \,\|\, \mathbf{H}_{\text{spec}}^{(\ell)}\big]\Big)$ ▷ $\mathbf{U}^{(\ell)} = \text{Dropout}\Big(\mathbf{W}_f^{(\ell)}[\mathbf{H}_{\text{spa}}^{(\ell)} \,\|\, \mathbf{H}_{\text{spec}}^{(\ell)}]\Big)$
  $\mathbf{X}^{(\ell)} \leftarrow \sigma\Big(\text{GCN}(\mathbf{U}^{(\ell)}, \mathbf{A})\Big)$ ▷ $\mathbf{X}^{(\ell)} = \sigma\Big(\text{GCN}(\mathbf{U}^{(\ell)}, \mathbf{A})\Big)$
end for
$\mathbf{Z}_g \leftarrow \mathbf{X}^{(L)}$
return $\mathbf{Z}_g$

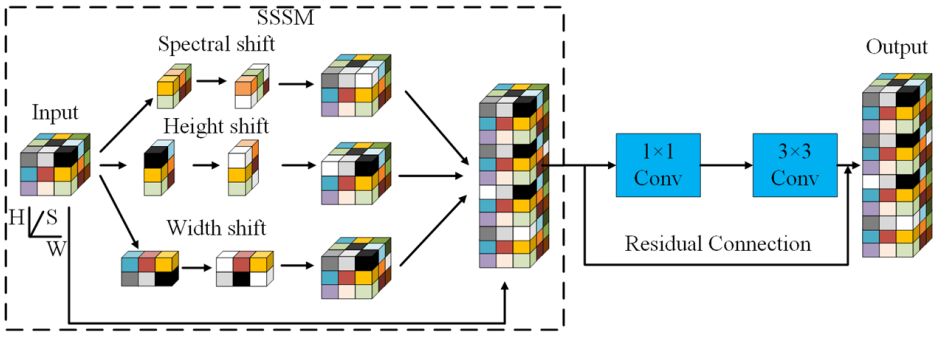


Figure 2. Detailed structure of the Spectral–Spatial Shift Module. The module performs cyclic shifts along spectral (S), height (H), and width (W) dimensions, followed by concatenation and convolutional fusion. The residual connection preserves gradient flow and stabilizes training. Shift operations enable efficient cross-dimensional interaction without introducing additional parameters.

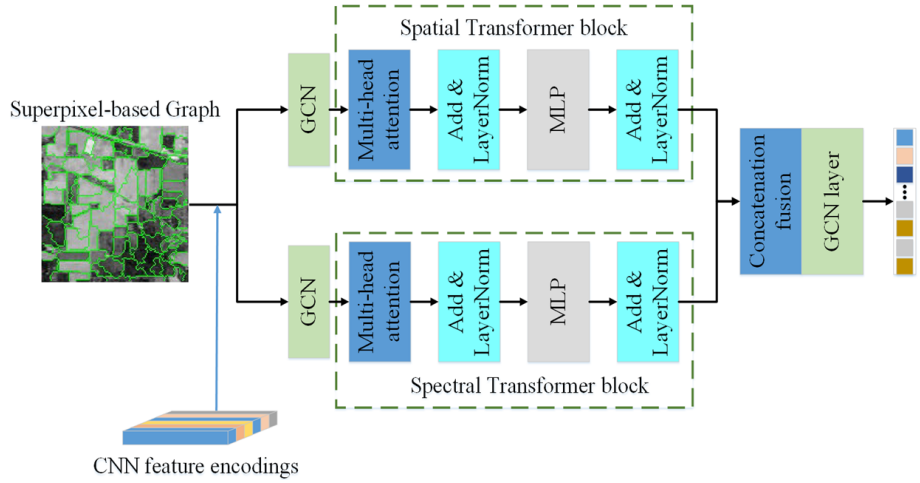


Figure 3. Architecture of the dual-branch spectral–spatial Graph Transformer block. The spatial path (**top**) processes graph-convolved features through multi-head self-attention to capture long-range

spatial dependencies. The spectral path (**bottom**) employs similar mechanisms for cross-band correlation modeling. Both branches are fused via concatenation and graph convolution, with layer normalization and residual connections applied throughout.

3.4.1. Spatial Path

The spatial path captures long-range spatial dependencies through graph-contextualized self-attention:

$$\mathbf{H}_{\text{spa}}^{(\ell)} = \text{GCN}(\mathbf{X}^{(\ell-1)}, \mathbf{A}) = \sigma(\hat{\mathbf{A}}\mathbf{X}^{(\ell-1)}\mathbf{W}_{\text{gcn}}), \tag{12}$$

$$\mathbf{Z}_{\text{spa}}^{(\ell)} = \text{LN}(\mathbf{X}^{(\ell-1)} + \text{MHSA}(\mathbf{H}_{\text{spa}}^{(\ell)})), \tag{13}$$

$$\mathbf{H}_{\text{spa}}^{(\ell)} = \text{LN}(\mathbf{Z}_{\text{spa}}^{(\ell)} + \text{FFN}(\mathbf{Z}_{\text{spa}}^{(\ell)})), \tag{14}$$

where $\hat{\mathbf{A}} = \tilde{\mathbf{D}}^{-\frac{1}{2}}\tilde{\mathbf{A}}\tilde{\mathbf{D}}^{-\frac{1}{2}}$ denotes the normalized adjacency matrix with self-loops [51], MHSA represents multi-head self-attention [52], LN is layer normalization, and FFN is a position-wise feed-forward network.

3.4.2. Spectral Path

The spectral path focuses on cross-band correlations through spectral attention mechanisms:

$$\mathbf{H}_{\text{spec}}^{(\ell)} = \text{GCN}(\mathbf{X}^{(\ell-1)}, \mathbf{A}), \tag{15}$$

$$\mathbf{Z}_{\text{spec}}^{(\ell)} = \text{LN}(\mathbf{X}^{(\ell-1)} + \text{MHSA}(\mathbf{H}_{\text{spec}}^{(\ell)})), \tag{16}$$

$$\mathbf{H}_{\text{spec}}^{(\ell)} = \text{LN}(\mathbf{Z}_{\text{spec}}^{(\ell)} + \text{FFN}(\mathbf{Z}_{\text{spec}}^{(\ell)})). \tag{17}$$

3.4.3. Feature Fusion and Propagation

The outputs from both paths are integrated through concatenation and graph-convolutional fusion:

$$\mathbf{U}^{(\ell)} = \text{Dropout}\Big(\mathbf{W}_f^{(\ell)}[\mathbf{H}_{\text{spa}}^{(\ell)} \parallel \mathbf{H}_{\text{spec}}^{(\ell)}]\Big), \tag{18}$$

$$\mathbf{X}^{(\ell)} = \sigma\Big(\text{GCN}(\mathbf{U}^{(\ell)}, \mathbf{A})\Big). \tag{19}$$

After L layers, the final node representations $\mathbf{Z}_g = \mathbf{X}^{(L)}$ are projected back to pixel space using the assignment matrix $\mathbf{Q}$ for classification:

$$\mathbf{Z}_{\text{pixel}} = \mathbf{Q}\mathbf{Z}_g. \tag{20}$$

This architecture enables SSGTN to capture complementary information: graph convolutions encode local structural priors, while self-attention mechanisms model long-range spatial and spectral dependencies, resulting in a comprehensive representation for hyperspectral image classification.

4. Results

To comprehensively evaluate the performance of the proposed SSGTN framework, we conduct extensive experiments on three benchmark hyperspectral datasets with diverse spatial resolutions, spectral configurations, and land-cover characteristics.

4.1. Datasets

4.1.1. Indian Pines

The Indian Pines dataset was acquired by the Airborne Visible/Infrared Imaging Spectrometer (AVIRIS) sensor over an agricultural area in Northwestern Indiana. The

dataset comprises 145 × 145 pixels with 224 spectral bands in the wavelength range of 400–2500 nm. After removing noisy and water-absorption bands, 200 spectral bands are retained for analysis. The spatial resolution is 20 m per pixel, and the scene contains 16 distinct land-cover classes, primarily consisting of various crop types, forests, and natural vegetation. This dataset presents challenges due to its moderate spatial resolution and significant spectral similarities between different crop species.

4.1.2. WHU-Hi-LongKou

The WHU-Hi-LongKou dataset was collected by a Headwall Nano-Hyperspec imaging sensor mounted on a DJI Matrice 600 Pro UAV platform. The imagery covers 550 × 400 pixels with 270 spectral bands spanning the visible to near-infrared spectrum (400–1000 nm). With a high spatial resolution of 0.463 m, this dataset captures detailed agricultural patterns in Longkou, Hubei, China. It encompasses nine land-cover classes including various crop types (corn, cotton, and rice) and water bodies, totaling 204,542 labeled pixels. The high spatial resolution and rich spectral information make this dataset suitable for evaluating fine-grained classification performance.

4.1.3. Houston2018

The Houston2018 dataset was provided as part of the 2018 IEEE GRSS Data Fusion Contest, covering an urban–rural area in Houston, Texas. The dataset consists of 601 × 2384 pixels with 50 spectral bands in the 380–1050 nm range at 2.5 m spatial resolution. It includes 20 land-cover classes with significant class imbalance, ranging from abundant categories like healthy grass (9799 samples) to rare classes such as synthetic turf (684 samples). The total of 504,856 labeled pixels and the complex urban landscape make this dataset particularly challenging for classification algorithms.

4.2. Experimental Setup

4.2.1. Evaluation Metrics

We employ four standard evaluation metrics to comprehensively assess classification performance from different perspectives. Overall Accuracy (OA) measures the global classification correctness:

$$\mathrm{OA} = \frac{1}{N}\sum_{i=1}^{N}\mathbb{I}(y_i = \hat{y}_i), \tag{21}$$

where N is the total number of test samples, y_i and $\hat{y}_i$ denote the true and predicted labels, respectively, and $\mathbb{I}(\cdot)$ is the indicator function.

Average Accuracy (AA) computes the mean of per-class accuracies, providing a balanced performance assessment across classes:

$$\mathrm{AA} = \frac{1}{C}\sum_{i=1}^{C}\frac{\mathrm{TP}_i}{\mathrm{TP}_i + \mathrm{FN}_i}, \tag{22}$$

where C is the number of classes, and TP_i and FN_i represent true positives and false negatives for class i.

The Kappa coefficient (κ) quantifies classification agreement beyond chance:

$$\kappa = \frac{p_o - p_e}{1 - p_e}, \tag{23}$$

Per-class accuracy provides detailed insights into individual class performance:

$$\mathrm{acc}_i = \frac{\mathrm{TP}_i}{\mathrm{TP}_i + \mathrm{FN}_i}. \tag{24}$$

4.2.2. Implementation Details

All experiments are conducted on a computational platform equipped with an NVIDIA GeForce RTX 4090 GPU and Intel Xeon Silver 4310 CPU. To rigorously evaluate model performance under limited supervision, we adopt only 1% of labeled pixels per class (minimum one sample per class) for training, with an additional 1% for validation and the remaining 98% for testing. The validation set is used exclusively for early stopping and checkpoint selection (patience = 100), and the checkpoint with the best validation performance is used for final testing. The model is implemented using PyTorch 2.9.1 and trained for 600 epochs with the Adam optimizer, employing an initial learning rate of 5×10^{-4} with cosine annealing scheduling. No external pretraining, data augmentation, or other additional training techniques are used, and no extra pre-processing beyond what is described in this manuscript is applied. The superpixel graph construction uses $K = 300$ superpixels for Indian Pines and WHU-Hi-LongKou, and $K = 500$ for Houston2018, with compactness parameter $m = 0.05$ unless otherwise specified.

4.2.3. Benchmark Methods

We compare SSGTN against twelve state-of-the-art methods spanning four representative paradigms:

1. **CNN-based methods:** CNN-2D [10] extracts spatial features through cascaded 2D convolutions; SSRN [12] employs 3D convolutional layers for joint spectral–spatial modeling; HybridSN [12] integrates 2D and 3D convolutions for hierarchical feature extraction.
2. **Transformer-based methods:** SSFTT [31] utilizes spectral–spatial feature tokenization for sequence modeling; MorphFormer [33] incorporates morphological operations with Transformer architecture.
3. **Mamba-based methods:** Mamba-HSI adapts state–space models for hyperspectral imaging; MFormer [15] combines Mamba with Transformer components for long-range dependency modeling.
4. **GCN-based methods:** GCN [16] operates on superpixel-based graphs; CEGCN [42] enhances graph representations with CNN features; GraphMamba [20] integrates graph structures with state–space models.

For a fair comparison, all baseline methods are trained and evaluated using the training protocol as described in the previous subsection, while method-specific hyperparameters are set following the recommendations in the original papers or official repositories, and all other common settings are kept identical across methods.

4.3. Ablation Studies

4.3.1. Impact of Spectral–Spatial Graph Parameters

In the LDA-SLIC superpixel construction, the *scale* and *compactness* parameters jointly determine the granularity and regularity of the induced superpixel graph. The parameter study in Tables 1 and 2 reveals a consistent trade-off: overly fine segmentation tends to generate small, fragmented regions whose descriptors are more vulnerable to spectral noise and mixed pixels, whereas overly coarse segmentation increases region heterogeneity and blurs class boundaries, weakening the homogeneity assumption underlying region-level graph reasoning. The compactness parameter further mediates boundary adherence versus spatial regularity; moderate compactness typically preserves meaningful object contours while avoiding irregular, elongated superpixels that may distort adjacency relations. Notably, the sensitivity to these parameters is dataset-dependent: scenes with clearer object extents and higher spatial resolution exhibit a wider feasible range, while coarse-resolution agricultural scenes with high inter-class spectral similarity require a more careful balance to

reduce mixed superpixels. Overall, these observations motivate selecting a moderately fine graph granularity together with boundary-aware superpixels to best support subsequent graph propagation and global reasoning.

Table 1. Comprehensive parameter analysis on the Indian Pines dataset: performance metrics (OA, AA, κ in %) across different combinations of superpixel scale and compactness parameters with 1% training ratio. Optimal performance is achieved at scale = 30 and compactness = 0.05.

Compactness	Scale Parameter														
	5			10			30			60			100		
	OA	AA	κ	OA	AA	κ	OA	AA	κ	OA	AA	κ	OA	AA	κ
0.05	84.56	78.61	82.37	86.70	80.08	84.83	88.75	79.47	87.16	87.54	77.63	85.77	85.79	79.45	83.78
0.10	83.30	76.07	80.92	87.12	80.22	85.31	85.75	75.50	83.78	87.34	79.57	85.56	86.37	74.95	84.45
0.50	85.02	75.97	82.85	84.79	75.45	82.64	84.55	76.74	82.36	81.93	71.43	79.40	80.31	71.92	77.52
1.00	83.67	72.62	81.32	85.38	77.12	83.35	85.72	76.70	83.75	83.01	71.23	80.65	78.99	63.10	76.06
5.00	83.39	75.58	81.07	85.75	78.09	83.75	84.22	74.73	81.95	83.88	75.37	81.64	80.01	70.20	77.10

Table 2. Parameter sensitivity analysis on the WHU-Hi-LongKou dataset: performance metrics (OA, AA, κ in %) across scale and compactness configurations. The model maintains robust performance with OA consistently above 99.4%.

Compactness	Scale Parameter														
	30			60			100			200			300		
	OA	AA	κ	OA	AA	κ	OA	AA	κ	OA	AA	κ	OA	AA	κ
0.1	99.48	98.73	99.31	99.57	98.95	99.44	99.64	99.07	99.52	99.64	98.99	99.53	99.69	99.23	99.59
0.5	99.41	98.44	99.23	99.47	98.80	99.30	99.51	98.73	99.36	99.49	98.71	99.33	99.62	97.80	99.50
1.0	99.48	98.44	99.32	99.41	98.54	99.22	99.48	98.66	99.31	99.54	98.64	99.39	99.39	98.46	99.20
5.0	99.49	98.81	99.33	99.43	98.62	99.25	99.47	98.59	99.30	99.60	98.98	99.48	99.53	98.67	99.38
10	99.44	98.48	99.26	99.48	98.56	99.32	99.46	98.72	99.28	99.60	98.98	99.47	99.65	98.90	99.54

4.3.2. Component-Wise Analysis

We conduct component-wise ablation studies on the Houston2018 dataset to examine the contribution of each module in SSGTN (Table 3). Overall, the results indicate that the modules address different failure modes in low-label HSIC and thus provide complementary benefits. The Spectral Denoising Module (SDM) improves the reliability of region descriptors and affinity estimation by suppressing band redundancy and sensor noise prior to graph construction, which helps mitigate error propagation during subsequent message passing. The Spectral–Spatial Shift Module (SSSM) introduces an explicit yet parameter-free spectral–spatial interaction prior on the pixel grid before region aggregation, enhancing robustness to mixed pixels and local boundary perturbations. Building upon these strengthened low-level representations, the Spatial Transformer Branch captures long-range spatial context over the superpixel topology, whereas the Spectral Transformer Branch emphasizes non-local cross-band correlations that are not fully recovered by local propagation alone; their joint use therefore integrates spatial continuity with spectral correlation into a unified representation. Importantly, the additional overhead remains controlled because SSSM introduces no learnable parameters and attention operates on a compact superpixel graph rather than the full pixel lattice.

Table 3. Comprehensive ablation study on the Houston2018 dataset evaluating individual contributions of Spectral Denoising Module (SDM), Spectral–Spatial Shift Module (SSSM), Spatial Transformer Branch (SpaT), and Spectral Transformer Branch (SpeT) (mean ± std over five seeds).

Module Configuration				Performance Metrics (%)		
SDM	**SSSM**	**SpaT**	**SpeT**	κ	**OA**	**AA**
×	×	×	×	87.30 ± 0.25	90.26 ± 0.20	83.14 ± 0.99
×	×	×	✓	87.26 ± 0.22	90.23 ± 0.16	82.16 ± 1.37
×	×	✓	×	87.16 ± 0.33	90.15 ± 0.25	83.01 ± 1.37
×	×	✓	✓	87.34 ± 0.41	90.30 ± 0.32	82.91 ± 0.84
×	✓	×	×	90.11 ± 0.27	92.40 ± 0.21	84.00 ± 1.03
×	✓	×	✓	88.86 ± 0.10	91.42 ± 0.07	84.84 ± 1.27
×	✓	✓	×	89.73 ± 0.29	92.11 ± 0.23	83.14 ± 1.22
×	✓	✓	✓	91.78 ± 0.44	93.69 ± 0.34	87.25 ± 0.89
✓	×	×	×	89.57 ± 0.26	91.99 ± 0.19	84.44 ± 1.36
✓	×	×	✓	89.55 ± 0.26	91.98 ± 0.20	84.03 ± 2.56
✓	×	✓	×	89.77 ± 0.43	92.14 ± 0.33	84.58 ± 1.57
✓	×	✓	✓	89.93 ± 0.17	92.26 ± 0.13	84.87 ± 2.57
✓	✓	×	×	92.02 ± 0.26	93.87 ± 0.20	87.40 ± 0.94
✓	✓	×	✓	89.91 ± 0.35	92.25 ± 0.27	84.77 ± 3.13
✓	✓	✓	×	90.98 ± 0.24	93.07 ± 0.18	86.21 ± 1.86
✓	✓	✓	✓	92.55 ± 0.25	94.28 ± 0.19	88.14 ± 1.06

4.3.3. Training Ratio Analysis

To assess robustness under limited supervision, we vary the training sample ratio from 1% to 5% across all datasets (Table 4). The results highlight strong label efficiency: SSGTN benefits noticeably from small increases in supervision and exhibits diminishing returns as the annotation budget grows, which is consistent with the model's ability to exploit structural priors and global context when labels are scarce. The improvement pattern also differs across datasets. For WHU-Hi-LongKou, the accuracy saturates rapidly due to clearer object extents and higher spatial resolution, suggesting that structural regularity and strong spatial cues allow effective learning with very few labels. In contrast, Indian Pines and Houston2018 require more supervision to stabilize class-balanced performance, which can be attributed to stronger spectral confusion, mixed pixels, and class imbalance that make minority categories harder to learn. Overall, this analysis indicates that SSGTN is particularly suitable for low-label settings, while further gains in highly imbalanced scenes are more likely to be constrained by data scarcity in rare classes rather than representation capacity alone.

Table 4. OA, AA, and kappa (%) of SSGTN on three datasets under different training ratios. (mean ± std over five seeds)

Dataset	Metric	0.01	0.02	0.03	0.04	0.05
Indian Pines	OA	87.12	92.33	93.67	96.44	95.96
	AA	80.22	84.07	85.63	91.63	87.19
	κ	85.31	91.25	92.78	95.93	95.39
WHU-Hi-LongKou	OA	99.39	99.72	99.74	99.81	99.85
	AA	98.62	99.30	99.24	99.41	99.58
	κ	99.20	99.63	99.65	99.75	99.81
Houston2018	OA	94.24	95.94	96.52	96.99	97.38
	AA	87.73	91.04	92.54	93.22	94.17
	κ	92.50	94.71	95.47	96.08	96.59

4.3.4. Computational Complexity

To complement the performance-oriented ablations, we further report a per-image complexity comparison on the WHU-Hi-LongKou dataset in terms of FLOPs and model parameters (Table 5). A key observation is that the FLOPs are dominated more by the *token/region granularity* than by the parameter count. Patch-based 3D CNN/Transformer pipelines typically operate on dense spectral–spatial patches or long token sequences on the full pixel lattice, which leads to 10^3–10^4 G FLOPs per image. In contrast, superpixel-graph methods (GCN, CEGCN, and SSGTN) compress the image into a much smaller set of region nodes and perform message passing/attention on a sparse adjacency, substantially reducing computation while still leveraging contextual aggregation. Importantly, G-Mamba does not adopt superpixel segmentation and thus incurs high FLOPs despite being categorized as a graph/SSM hybrid, since its global modeling is still carried out at a much finer granularity. Meanwhile, the notably low FLOPs of MambaHSI are largely attributed to its tile-based processing strategy, which reduces the effective sequence length and computation per forward pass compared to patch-based tokenization used by several Transformer-style baselines. We note that peak GPU memory consumption is also strongly implementation-dependent (e.g., batch size, precision, and activation storage), and therefore we report FLOPs/parameters as hardware-agnostic indicators while providing qualitative discussion of memory trends through token/region granularity. Overall, SSGTN provides a favorable computation–capacity trade-off with a clear advantage in FLOPs, while its parameter size is comparatively larger and remains a noticeable drawback.

Table 5. Per-image complexity comparison on the WHU-Hi-LongKou dataset. FLOPs are reported in G, and parameters are reported in MB.

Methods	FLOPs (G)	Param (MB)
CNN-2D	4059.60	1.56
SSRN	8705.63	0.34
HybridSN	1686.79	1.40
SSFTT	1578.37	0.61
MorphFormer	1356.35	0.56
MambaHSI	12.08	0.51
MFormer	4979.38	1.19
GCN	7.86	0.14
CEGCN	66.99	0.70
G-Mamba	3008.70	1.30
SSGTN	60.89	1.79

4.4. Experimental Results

4.4.1. Comparative Performance Analysis

Tables 6–8 and Figures 4–6 indicate that SSGTN provides consistent improvements across three representative benchmarks under the 1% per-class protocol. We additionally report a classical support vector machine (SVM) baseline as a non-deep-learning reference. Across all three datasets, SVM yields noticeably lower OA and κ compared with deep models, and its large variance on Indian Pines indicates limited robustness under scarce supervision and strong spectral confusion. With Indian Pines, the dominant challenge is the combination of coarse spatial resolution, strong spectral similarity among crop types, and extreme class imbalance. In this regime, methods that rely heavily on local neighborhoods tend to be more affected by mixed pixels and scarce supervision for minority categories. The region-level structural prior and global reasoning in SSGTN improve spatial consistency and overall robustness, while class-balanced performance remains constrained by the lack of samples in rare classes. In WHU-Hi-LongKou, the scene exhibits clearer object extents and higher spatial resolution, and superpixel regions align well

with meaningful structures. Consequently, region-level graph modeling becomes highly effective and most competitive methods approach saturation, with SSGTN maintaining clean boundaries and fewer spurious fragments in the prediction maps. In Houston2018, the urban environment introduces high intra-class variability and pronounced imbalance. In this setting, combining graph-regularized spatial coherence with global contextual modeling is particularly beneficial for complex man-made structures, whereas residual errors are mainly concentrated in rare categories, suggesting that data imbalance is a primary bottleneck for further gains.

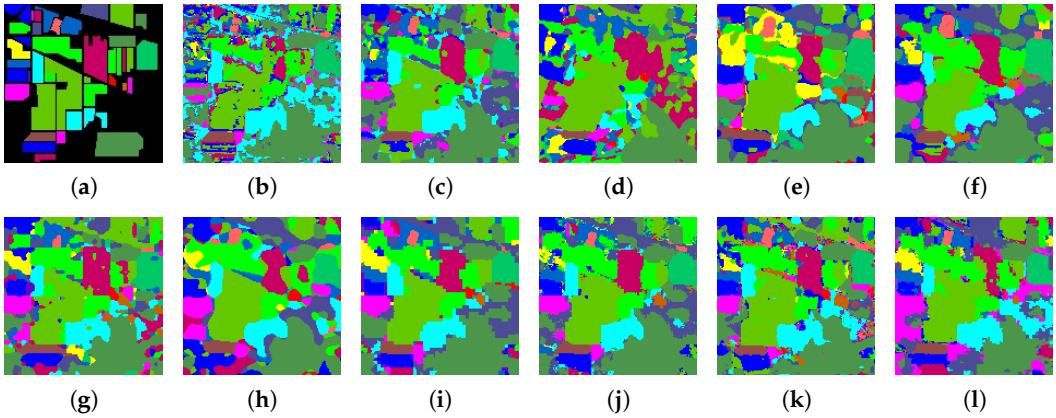


Figure 4. Classification maps of different models on the Indian Pines dataset with 1% samples per class as the training set. (**a**) Ground truth; (**b**) CNN-2D; (**c**) SSRN; (**d**) HybridSN; (**e**) SSFTT; (**f**) MorphFormer; (**g**) MambaHSI; (**h**) MFormer; (**i**) GCN; (**j**) CEGCN; (**k**) Graph-Mamba; (**l**) SSGTN (ours).

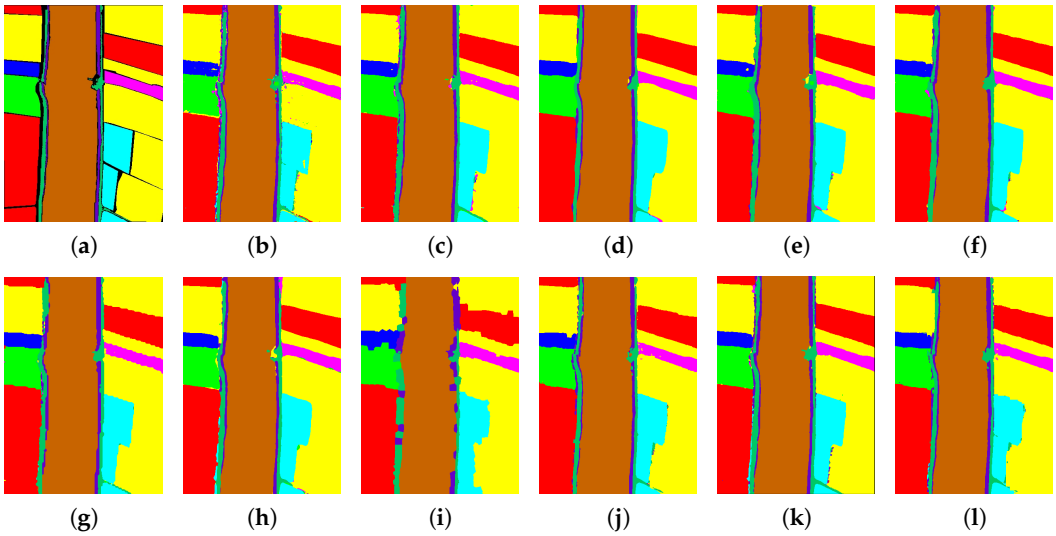


Figure 5. Classification maps of different models on the WHU-Hi-LongKou dataset with 1% samples per class as the training set. (**a**) Ground truth; (**b**) CNN-2D; (**c**) SSRN; (**d**) HybridSN; (**e**) SSFTT; (**f**) MorphFormer; (**g**) MambaHSI; (**h**) MFormer; (**i**) GCN; (**j**) CEGCN; (**k**) Graph-Mamba; (**l**) SSGTN (ours).

4.4.2. Qualitative Results and Visual Analysis

Classification maps in Figures 4–6 are examined from three complementary perspectives, i.e., intra-region homogeneity, boundary adherence, and robustness to sparse supervision. Compared with CNN-based baselines, which tend to produce fragmented predictions around mixed pixels, and purely global models, which may introduce scattered misclassifications when local structure is ambiguous, SSGTN yields spatially coherent regions while preserving class transitions at object boundaries. This behavior is consistent

with the intended role of the superpixel graph in enforcing structural regularity and the Transformer branch in compensating for long-range contextual dependencies, thereby mitigating both salt-and-pepper noise and excessive over-smoothing.

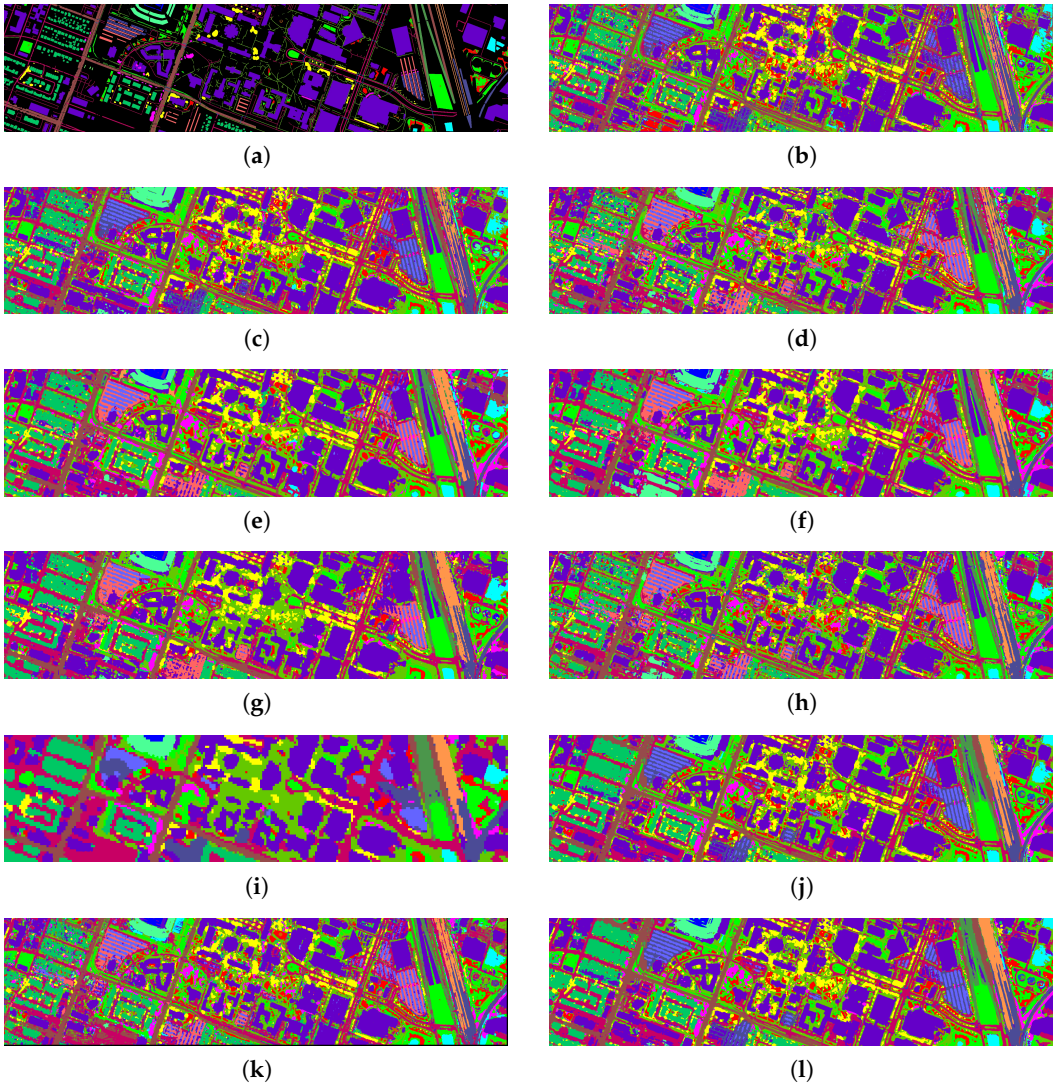


Figure 6. Classification maps of different models on the Houston2018 dataset with 1% samples per class as the training set. (**a**) Ground truth; (**b**) CNN-2D; (**c**) SSRN; (**d**) HybridSN; (**e**) SSFTT; (**f**) MorphFormer; (**g**) MambaHSI; (**h**) MFormer; (**i**) GCN; (**j**) CEGCN; (**k**) Graph-Mamba; (**l**) SSGTN (ours).

On Indian Pines (Figure 4), where small agricultural parcels and spectrally similar crop types often induce local confusion, SSGTN reduces isolated mislabeled pixels and maintains more continuous field patterns, indicating improved handling of mixed pixels under scarce labels. On WHU-Hi-LongKou (Figure 5), the dominant challenge is aligning predictions with elongated field boundaries; SSGTN better follows these boundaries and suppresses cross-field label leakage, reflecting effective region-level regularization. On the urban Houston2018 scene (Figure 6), fine-grained man-made structures and strong material similarity can cause boundary blur or sporadic errors; SSGTN preserves sharper transitions between adjacent classes and produces cleaner structural layouts, suggesting that combining graph-based locality with attention-driven context is beneficial for discriminating spectrally close categories.

Table 6. Results on the Indian Pines dataset with 1% training samples (mean ± std over five seeds).

Class No.	Hand-Crafted	CNNs			Transformer-Based		Mamba-Based		GCN			
	SVM	CNN-2D	SSRN	HybridSN	SSFTT	MorphFormer	MambaHSI	MFormer	GCN	CEGCN	G-Mamba	SSGTN
1	38.18 ± 14.26	35.41 ± 2.23	16.52 ± 1.39	56.36 ± 11.54	31.82 ± 31.53	62.27 ± 38.22	78.18 ± 12.16	85.33 ± 9.59	55.97 ± 4.56	28.29 ± 1.49	92.92 ± 6.08	39.74 ± 18.95
2	54.61 ± 5.93	76.94 ± 0.59	73.59 ± 0.75	56.89 ± 9.44	79.20 ± 14.88	85.38 ± 4.33	85.79 ± 3.55	73.86 ± 9.52	75.95 ± 1.25	79.29 ± 0.75	73.15 ± 7.69	85.11 ± 6.54
3	36.53 ± 12.81	47.47 ± 1.22	34.29 ± 1.58	65.31 ± 3.48	69.90 ± 12.02	66.28 ± 11.57	77.98 ± 6.96	78.69 ± 6.40	73.95 ± 1.05	81.58 ± 0.87	68.49 ± 11.31	75.82 ± 10.06
4	34.03 ± 8.06	35.45 ± 0.89	14.92 ± 1.33	27.64 ± 8.29	70.39 ± 23.59	73.25 ± 19.97	89.96 ± 7.19	65.02 ± 11.91	90.96 ± 0.89	69.03 ± 2.83	52.91 ± 6.09	83.67 ± 10.17
5	79.32 ± 4.02	54.76 ± 1.67	54.81 ± 2.29	42.45 ± 5.63	67.70 ± 11.09	72.98 ± 7.15	80.04 ± 6.88	86.57 ± 3.67	81.38 ± 0.92	68.35 ± 2.76	79.57 ± 8.21	89.47 ± 3.53
6	84.96 ± 4.97	98.29 ± 0.11	97.87 ± 0.22	90.17 ± 1.74	91.99 ± 6.31	90.64 ± 5.29	90.59 ± 5.05	96.71 ± 1.76	91.04 ± 0.53	95.81 ± 0.13	94.96 ± 2.62	97.12 ± 1.17
7	86.15 ± 6.25	78.29 ± 0.85	40.25 ± 2.01	66.15 ± 12.02	86.92 ± 18.76	91.54 ± 6.88	96.15 ± 4.21	0.00 ± 0.00	98.52 ± 0.18	77.69 ± 1.82	100.00 ± 0.00	72.59 ± 20.23
8	89.49 ± 7.05	78.04 ± 2.18	91.79 ± 0.56	99.06 ± 1.88	98.12 ± 3.75	99.83 ± 0.38	99.36 ± 1.28	97.63 ± 4.00	99.70 ± 0.02	99.74 ± 0.02	91.30 ± 7.57	98.84 ± 1.04
9	34.44 ± 12.86	42.75 ± 1.15	14.97 ± 1.04	86.67 ± 10.30	35.56 ± 35.92	31.11 ± 16.01	76.67 ± 19.37	0.00 ± 0.00	57.66 ± 3.65	21.17 ± 1.66	98.82 ± 2.35	40.00 ± 21.73
10	49.71 ± 7.81	63.94 ± 0.68	68.75 ± 0.82	66.68 ± 7.53	69.14 ± 13.19	74.35 ± 3.48	78.78 ± 3.87	76.69 ± 4.76	74.22 ± 0.85	79.41 ± 0.73	66.78 ± 3.84	80.72 ± 11.00
11	71.92 ± 3.87	85.54 ± 0.57	90.66 ± 0.54	87.98 ± 2.82	86.25 ± 5.44	85.96 ± 4.54	90.71 ± 4.02	87.59 ± 1.60	91.17 ± 0.73	90.39 ± 0.69	83.96 ± 5.89	90.69 ± 6.60
12	24.72 ± 9.33	37.03 ± 0.94	26.48 ± 1.61	45.99 ± 9.78	42.62 ± 17.18	44.99 ± 16.78	71.57 ± 7.84	57.41 ± 13.61	80.21 ± 0.68	65.81 ± 1.33	56.84 ± 11.19	70.23 ± 4.52
13	91.66 ± 2.47	98.60 ± 0.06	96.72 ± 0.66	65.07 ± 4.54	80.50 ± 18.87	97.99 ± 2.73	99.50 ± 0.64	92.32 ± 5.86	92.52 ± 0.52	97.01 ± 0.45	96.01 ± 3.05	99.00 ± 1.37
14	91.19 ± 1.30	98.99 ± 0.04	99.87 ± 0.02	89.99 ± 3.03	92.56 ± 5.80	95.00 ± 3.37	95.58 ± 2.39	98.40 ± 0.75	94.54 ± 0.39	98.76 ± 0.09	98.71 ± 0.89	98.66 ± 0.86
15	25.08 ± 4.14	46.89 ± 1.03	30.28 ± 0.83	39.42 ± 1.33	61.59 ± 16.17	75.56 ± 11.33	84.07 ± 5.46	71.41 ± 10.03	90.46 ± 1.13	75.97 ± 2.14	42.94 ± 8.56	74.26 ± 12.63
16	75.38 ± 12.60	45.12 ± 2.91	29.16 ± 2.39	21.10 ± 9.84	72.97 ± 35.01	83.08 ± 27.69	76.26 ± 13.59	80.65 ± 14.85	37.36 ± 0.31	26.01 ± 1.11	98.88 ± 1.41	79.66 ± 16.48
OA (%)	63.88 ± 1.80	74.15 ± 0.27	72.89 ± 0.26	71.64 ± 1.19	78.80 ± 3.39	81.64 ± 1.12	86.96 ± 0.75	83.06 ± 1.28	85.07 ± 0.31	84.55 ± 0.18	78.33 ± 2.39	87.12 ± 2.91
AA (%)	60.46 ± 2.03	63.97 ± 0.32	55.06 ± 0.35	62.93 ± 2.10	71.08 ± 7.82	76.89 ± 4.15	85.70 ± 2.17	71.77 ± 1.32	80.35 ± 0.71	72.15 ± 0.45	81.02 ± 1.27	80.22 ± 3.74
κ (%)	58.37 ± 2.29	70.05 ± 0.31	68.29 ± 0.32	67.31 ± 1.62	75.75 ± 3.99	79.01 ± 1.29	85.10 ± 0.89	80.59 ± 1.44	82.99 ± 0.35	82.32 ± 0.21	75.86 ± 2.69	85.31 ± 3.25

Table 7. Results on the WHU-Hi-LongKou dataset with 1% training samples (mean ± std over five seeds).

Class No.	Hand-Crafted	CNNs			Transformer-Based		Mamba-Based		GCN			
	SVM	CNN-2D	SSRN	HybridSN	SSFTT	MorphFormer	MambaHSI	MFormer	GCN	CEGCN	G-Mamba	SSGTN
1	98.07 ± 0.45	99.82 ± 0.04	99.88 ± 0.04	99.91 ± 0.08	99.88 ± 0.14	99.86 ± 0.08	99.74 ± 0.17	99.94 ± 0.04	97.39 ± 0.38	99.23 ± 0.96	99.95 ± 0.04	99.76 ± 0.23
2	74.66 ± 5.69	98.05 ± 0.61	98.16 ± 0.81	99.62 ± 0.18	99.47 ± 0.12	99.24 ± 0.24	98.83 ± 0.81	99.62 ± 0.31	97.64 ± 1.37	99.77 ± 0.42	99.81 ± 0.14	99.62 ± 0.43
3	73.62 ± 7.77	96.29 ± 0.88	97.33 ± 0.29	99.39 ± 0.44	98.65 ± 0.65	98.59 ± 1.08	98.24 ± 0.91	97.11 ± 1.87	91.13 ± 2.33	96.63 ± 1.30	98.91 ± 1.07	97.93 ± 1.28
4	96.15 ± 0.39	99.67 ± 0.04	99.73 ± 0.11	99.64 ± 0.06	99.75 ± 0.06	99.66 ± 0.08	99.56 ± 0.17	99.76 ± 0.10	97.49 ± 0.66	99.66 ± 1.49	99.53 ± 0.23	99.88 ± 0.32
5	57.78 ± 9.29	94.59 ± 1.50	92.93 ± 1.02	95.32 ± 1.59	96.41 ± 2.95	97.73 ± 2.10	96.52 ± 0.69	97.98 ± 0.84	98.58 ± 0.01	96.63 ± 1.23	98.72 ± 0.36	99.96 ± 0.01
6	94.55 ± 1.93	99.76 ± 0.34	99.67 ± 0.46	99.84 ± 0.05	99.87 ± 0.15	99.49 ± 0.41	99.34 ± 0.40	99.68 ± 0.16	96.59 ± 1.86	99.86 ± 0.11	96.41 ± 1.47	97.78 ± 0.79
7	99.96 ± 0.02	99.98 ± 0.01	99.98 ± 0.01	99.96 ± 0.01	99.89 ± 0.11	99.93 ± 0.05	99.85 ± 0.07	99.91 ± 0.04	98.96 ± 0.17	99.91 ± 0.39	96.41 ± 0.61	98.42 ± 0.73
8	79.88 ± 4.74	96.52 ± 0.82	96.19 ± 0.96	93.84 ± 1.47	94.52 ± 2.54	94.66 ± 2.26	90.25 ± 1.91	95.64 ± 1.34	79.67 ± 6.13	97.33 ± 0.99	99.59 ± 0.07	97.93 ± 0.54
9	63.54 ± 3.12	94.89 ± 0.92	94.43 ± 1.84	95.38 ± 0.44	97.04 ± 1.23	95.18 ± 1.18	86.44 ± 4.44	95.44 ± 0.96	55.34 ± 5.20	98.44 ± 1.00	98.91 ± 0.23	97.68 ± 0.53
OA (%)	94.24 ± 0.57	99.28 ± 0.09	99.33 ± 0.08	99.39 ± 0.10	99.48 ± 0.11	99.41 ± 0.13	98.90 ± 0.10	99.50 ± 0.05	96.14 ± 0.11	99.47 ± 0.02	99.59 ± 0.07	99.62 ± 0.10
AA (%)	82.02 ± 2.47	97.39 ± 0.51	97.56 ± 0.11	98.07 ± 0.39	98.39 ± 0.42	98.26 ± 0.42	96.53 ± 0.51	98.34 ± 0.26	90.31 ± 0.80	98.61 ± 1.19	98.91 ± 0.23	98.97 ± 0.12
κ (%)	92.40 ± 0.77	99.06 ± 0.13	99.12 ± 0.10	99.20 ± 0.14	99.31 ± 0.15	99.22 ± 0.17	98.55 ± 0.14	99.34 ± 0.07	94.92 ± 0.15	99.30 ± 0.02	99.46 ± 0.09	99.51 ± 0.11

Table 8. Results on the Houston2018 dataset with 1% training samples (mean ± std over five seeds).

Class No.	Hand-Crafted	CNNs			Transformer-Based		Mamba-Based		GCN			
	SVM	CNN-2D	SSRN	HybridSN	SSFTT	MorphFormer	MambaHSI	MFormer	GCN	CEGCN	G-Mamba	SSGTN
1	96.94 ± 0.96	89.74 ± 0.40	83.87 ± 3.72	76.84 ± 5.61	79.26 ± 8.74	78.55 ± 1.33	79.22 ± 3.41	73.53 ± 9.66	61.07 ± 4.24	85.13 ± 1.86	84.20 ± 4.16	86.47 ± 0.94
2	95.20 ± 0.47	95.19 ± 0.41	93.57 ± 2.33	91.88 ± 1.37	90.99 ± 3.97	92.22 ± 1.05	91.67 ± 0.47	91.74 ± 1.50	86.39 ± 1.33	94.91 ± 0.70	84.20 ± 4.16	94.04 ± 0.65
3	100.00 ± 0.00	96.61 ± 2.76	99.58 ± 0.51	86.15 ± 1.71	99.10 ± 0.64	97.88 ± 4.08	100.00 ± 0.00	88.77 ± 4.21	99.38 ± 1.25	98.78 ± 0.78	96.72 ± 3.69	99.55 ± 0.40
4	95.67 ± 0.75	96.91 ± 0.26	94.59 ± 1.87	96.13 ± 1.55	96.34 ± 1.11	96.41 ± 1.33	94.06 ± 0.47	92.28 ± 1.61	77.50 ± 1.53	96.46 ± 0.75	95.58 ± 0.89	96.30 ± 0.31
5	86.74 ± 2.86	83.19 ± 0.99	93.52 ± 2.06	71.73 ± 2.83	85.45 ± 4.87	79.77 ± 6.74	80.48 ± 3.94	77.33 ± 5.33	58.65 ± 2.27	78.49 ± 4.13	83.34 ± 3.12	81.81 ± 3.88
6	96.43 ± 0.92	96.02 ± 1.24	98.78 ± 0.86	96.26 ± 1.07	97.36 ± 2.98	96.43 ± 2.80	97.30 ± 3.13	94.55 ± 6.85	98.21 ± 2.69	97.11 ± 2.37	98.96 ± 0.22	98.70 ± 1.41
7	85.92 ± 10.81	83.08 ± 9.47	99.44 ± 1.12	70.00 ± 10.22	62.31 ± 5.29	76.23 ± 15.11	73.38 ± 8.34	84.03 ± 12.37	26.06 ± 24.51	56.64 ± 21.23	87.83 ± 11.94	80.43 ± 9.24
8	91.50 ± 0.94	91.08 ± 0.88	94.19 ± 1.62	90.93 ± 1.23	94.72 ± 1.31	96.29 ± 0.89	97.29 ± 0.32	90.34 ± 2.16	97.41 ± 0.63	97.58 ± 0.83	91.54 ± 1.08	98.68 ± 0.19
9	84.26 ± 2.38	97.45 ± 0.07	97.55 ± 0.49	98.57 ± 0.14	98.14 ± 0.52	97.58 ± 0.39	98.73 ± 0.10	97.91 ± 0.44	97.27 ± 0.19	98.70 ± 0.12	97.84 ± 0.23	98.66 ± 0.16
10	58.85 ± 2.45	73.79 ± 1.03	83.42 ± 2.76	78.36 ± 1.98	78.87 ± 4.46	80.40 ± 4.35	81.71 ± 1.18	75.36 ± 2.43	76.32 ± 1.06	83.75 ± 1.70	75.86 ± 1.45	84.92 ± 1.09
11	65.17 ± 1.21	69.46 ± 0.81	74.97 ± 2.25	62.00 ± 2.98	71.01 ± 2.23	67.71 ± 4.62	69.75 ± 1.86	64.92 ± 3.38	49.76 ± 1.52	75.53 ± 2.53	72.15 ± 1.45	75.59 ± 3.11
12	39.31 ± 3.57	5.90 ± 4.75	37.81 ± 5.42	11.65 ± 2.21	5.77 ± 7.66	3.55 ± 3.54	16.70 ± 3.86	10.27 ± 2.62	0.85 ± 1.66	8.19 ± 3.95	11.79 ± 4.34	16.37 ± 6.35
13	64.53 ± 3.12	83.88 ± 1.05	84.01 ± 6.09	91.07 ± 0.65	91.05 ± 2.11	88.90 ± 4.99	92.95 ± 0.51	85.60 ± 2.58	93.62 ± 0.57	93.83 ± 1.67	85.75 ± 0.67	93.81 ± 0.83
14	91.24 ± 0.66	88.69 ± 0.78	89.63 ± 1.95	92.93 ± 1.70	93.13 ± 3.96	95.78 ± 1.46	94.59 ± 1.10	86.99 ± 9.79	92.44 ± 2.17	95.12 ± 0.82	89.04 ± 1.48	96.05 ± 1.56
15	98.00 ± 0.79	98.15 ± 0.65	98.19 ± 0.93	96.11 ± 2.17	98.85 ± 0.89	97.65 ± 1.72	98.82 ± 0.44	98.48 ± 0.80	91.74 ± 2.17	99.08 ± 0.62	98.95 ± 0.64	98.20 ± 1.85
16	92.99 ± 0.46	91.97 ± 0.85	94.49 ± 1.65	93.87 ± 1.05	95.35 ± 2.19	92.94 ± 3.48	93.50 ± 0.85	93.76 ± 4.08	75.03 ± 6.48	96.58 ± 0.60	94.36 ± 0.36	96.64 ± 1.67
17	89.44 ± 3.75	57.53 ± 27.67	0.00 ± 0.00	1.54 ± 2.44	45.63 ± 45.16	23.52 ± 28.44	58.31 ± 18.19	22.07 ± 16.53	0.00 ± 0.00	0.00 ± 0.00	88.03 ± 5.24	65.26 ± 26.31
18	82.23 ± 2.38	84.25 ± 2.06	94.68 ± 2.42	96.79 ± 0.92	87.07 ± 3.58	90.91 ± 4.00	87.99 ± 1.41	87.43 ± 4.26	58.09 ± 12.37	93.74 ± 2.89	88.55 ± 2.05	93.73 ± 1.65
19	86.55 ± 3.84	92.29 ± 0.94	96.56 ± 2.46	96.94 ± 0.83	93.63 ± 2.61	91.58 ± 6.01	95.50 ± 1.83	94.09 ± 2.59	84.32 ± 2.22	98.27 ± 0.42	92.66 ± 0.75	99.65 ± 0.32
20	98.23 ± 0.40	97.54 ± 0.44	99.18 ± 0.42	98.71 ± 0.65	98.98 ± 0.96	98.76 ± 1.97	98.29 ± 1.29	98.69 ± 0.90	99.90 ± 0.10	99.89 ± 0.09	99.32 ± 0.29	99.75 ± 0.21
OA (%)	81.41 ± 1.48	90.40 ± 0.27	92.14 ± 0.49	91.17 ± 0.25	91.99 ± 0.48	91.59 ± 0.70	92.79 ± 0.09	89.89 ± 0.39	87.89 ± 0.12	93.97 ± 0.11	91.02 ± 0.15	94.24 ± 0.16
AA (%)	84.96 ± 0.67	83.64 ± 1.41	85.40 ± 0.15	79.97 ± 1.20	83.15 ± 2.24	82.15 ± 2.80	85.01 ± 0.97	80.41 ± 0.98	71.20 ± 1.11	82.39 ± 1.20	86.24 ± 1.01	87.73 ± 1.97
κ (%)	76.58 ± 1.74	87.46 ± 0.36	89.78 ± 0.61	88.46 ± 0.33	89.57 ± 0.61	89.07 ± 0.91	90.57 ± 0.12	86.79 ± 0.50	84.19 ± 0.15	92.14 ± 0.14	88.31 ± 0.20	92.50 ± 0.21

5. Discussion

The experimental results demonstrate that SSGTN effectively addresses key challenges in hyperspectral image classification under limited labeled data through its novel architectural design. The dual-branch framework successfully leverages complementary strengths: the graph-based branch preserves discriminative spectral patterns in homogeneous regions, while the Transformer branch captures long-range dependencies essential for complex landscapes.

However, SSGTN exhibits limitations in handling severely underrepresented classes, as evidenced by the poor performance on Class 12 in Houston2018. This limitation stems from graph sparsity in rare classes and attention bias toward dominant categories. Future work should explore topology-aware graph sampling and attention regularization to improve minority class representation.

Compared to CNN-based methods, SSGTN achieves superior spatial coherence through graph-structured regularization. Relative to pure GCN approaches, the Transformer branch mitigates oversmoothing in heterogeneous scenes. The computational complexity of joint graph-attention learning necessitates careful hardware considerations for large-scale deployments, though the sparse graph construction provides significant efficiency gains.

The consistent performance advantage across diverse datasets and low-label training regimes demonstrates that SSGTN is a robust and generalizable framework for hyperspectral image classification, particularly in practical scenarios where labeled data are limited and computational efficiency is critical.

6. Conclusions

This paper has presented the SSGTN, a novel dual-branch architecture that advances hyperspectral image classification under limited supervision. The proposed framework integrates four key innovations: an LDA-SLIC superpixel graph construction module that preserves discriminative spectral–spatial features, a spectral denoising module for noise suppression, a parameter-efficient Spectral–Spatial Shift Module for multi-dimensional feature interaction, and a dual-branch GCN-Transformer that jointly models local topology and global dependencies. Extensive experiments on three benchmark datasets demonstrate that SSGTN consistently outperforms state-of-the-art methods under limited supervision conditions while maintaining computational efficiency through sparse graph representations and optimized module design.

Future research will focus on enhancing SSGTN's capability to handle severe class imbalance through advanced graph sampling strategies and attention regularization techniques. We will also explore self-supervised pretraining approaches to further improve sample efficiency and investigate adaptive graph construction mechanisms for better boundary preservation in heterogeneous landscapes. These developments aim to extend the framework's applicability to more complex real-world scenarios while addressing current limitations in computational scalability and minority class representation.

Author Contributions: Conceptualization, H.S. and Z.L.; methodology, H.S.; investigation, H.S., Z.L., Y.M., G.Z. and X.D.; editing, H.S. and Z.L.; writing—review, H.S., Z.L., Y.M., G.Z. and X.D. All authors have read and agreed to the published version of the manuscript.

Funding: This work was jointly supported by the National Natural Science Foundation of China (NSFC) under Grant 62403155; the Guangzhou Basic and Applied Basic Research Topics under Grant 2024A04J2081; the Guangdong Basic and Applied Basic Research Fund under Grant 2023A1515011850; the Guangzhou Science and Technology Planning Project under Grant 2024A03J0460; and the College

Students' Science and Technology Innovation Cultivation Project of Guangdong Province, China under Grant pdjh2024a293.

Data Availability Statement: The hyperspectral datasets used in this study are publicly available. The Indian Pines dataset is available at https://www.ehu.eus/ccwintco/index.php/Hyperspectral_Remote_Sensing_Scenes#Indian_Pines, accessed on 28 November 2025. The WHU-Hi-LongKou dataset is part of the WHU-Hi hyperspectral benchmark and is available at https://rsidea.whu.edu.cn/e-resource_WHUHi_sharing.htm, accessed on 28 November 2025. The Houston2018 dataset was released as part of the 2018 IEEE GRSS Data Fusion Contest, jointly provided by the National Center for Airborne Laser Mapping (NCALM) and the University of Houston; detailed information and access are available at https://machinelearning.ee.uh.edu/2018-ieee-grss-data-fusion-challenge-fusion-of-multispectral-lidar-and-hyperspectral-data/, accessed on 28 November 2025.

Conflicts of Interest: The authors declare no conflicts of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript; or in the decision to publish the results.

References

1. Mahlein, A.K. Plant disease detection by imaging sensors–parallels and specific demands for precision agriculture and plant phenotyping. *Plant Dis.* **2016**, *100*, 241–251. [CrossRef] [PubMed]
2. Tan, K.; Ma, W.; Chen, L.; Wang, H.; Du, Q.; Du, P.; Yan, B.; Liu, R.; Li, H. Estimating the distribution trend of soil heavy metals in mining area from HyMap airborne hyperspectral imagery based on ensemble learning. *J. Hazard. Mater.* **2021**, *401*, 123288. [CrossRef]
3. Van der Meer, F.D.; Van der Werff, H.M.; Van Ruitenbeek, F.J.; Hecker, C.A.; Bakker, W.H.; Noomen, M.F.; Van Der Meijde, M.; Carranza, E.J.M.; De Smeth, J.B.; Woldai, T. Multi- and hyperspectral geologic remote sensing: A review. *Int. J. Appl. Earth Obs. Geoinf.* **2012**, *14*, 112–128. [CrossRef]
4. Wu, Y.; Wang, Y.; Zhang, D. Design and Analysis of Spaceborne Hyperspectral Imaging System for Coastal Studies. *Remote Sens.* **2025**, *17*, 986. [CrossRef]
5. Xu, J.L.; Esquerre, C.; Sun, D.W. Methods for performing dimensionality reduction in hyperspectral image classification. *J. Near Infrared Spectrosc.* **2018**, *26*, 61–75. [CrossRef]
6. Ye, Z.; He, M.; Fowler, J.E.; Du, Q. Hyperspectral image classification based on spectra derivative features and locality preserving analysis. In Proceedings of the 2014 IEEE China Summit & International Conference on Signal and Information Processing (Chinasip), Xi'an, China, 9–13 July 2014; pp. 138–142.
7. Tan, K.; Li, E.; Du, Q.; Du, P. Hyperspectral image classification using band selection and morphological profiles. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2013**, *7*, 40–48. [CrossRef]
8. Hao, T.; Zhang, Z.; Crabbe, M.J.C. Few-Shot Hyperspectral Remote Sensing Image Classification via an Ensemble of Meta-Optimizers with Update Integration. *Remote Sens.* **2024**, *16*, 2988. [CrossRef]
9. Dao, P.D.; Mantripragada, K.; He, Y.; Qureshi, F.Z. Improving hyperspectral image segmentation by applying inverse noise weighting and outlier removal for optimal scale selection. *ISPRS J. Photogramm. Remote Sens.* **2021**, *171*, 348–366. [CrossRef]
10. Hu, W.; Huang, Y.; Wei, L.; Zhang, F.; Li, H. Deep convolutional neural networks for hyperspectral image classification. *J. Sens.* **2015**, *2015*, 258619. [CrossRef]
11. Li, Y.; Zhang, H.; Shen, Q. Spectral–spatial classification of hyperspectral imagery with 3D convolutional neural network. *Remote Sens.* **2017**, *9*, 67. [CrossRef]
12. Roy, S.K.; Krishna, G.; Dubey, S.R.; Chaudhuri, B.B. HybridSN: Exploring 3-D–2-D CNN feature hierarchy for hyperspectral image classification. *IEEE Geosci. Remote Sens. Lett.* **2019**, *17*, 277–281. [CrossRef]
13. He, X.; Chen, Y.; Lin, Z. Spatial-spectral transformer for hyperspectral image classification. *Remote Sens.* **2021**, *13*, 498. [CrossRef]
14. Hong, D.; Han, Z.; Yao, J.; Gao, L.; Zhang, B.; Plaza, A.; Chanussot, J. SpectralFormer: Rethinking hyperspectral image classification with transformers. *IEEE Trans. Geosci. Remote Sens.* **2021**, *60*, 5518615. [CrossRef]
15. He, Y.; Tu, B.; Liu, B.; Li, J.; Plaza, A. 3DSS-Mamba: 3D-spectral-spatial mamba for hyperspectral image classification. *IEEE Trans. Geosci. Remote Sens.* **2024**, *62*, 5534216. [CrossRef]
16. Hong, D.; Gao, L.; Yao, J.; Zhang, B.; Plaza, A.; Chanussot, J. Graph convolutional networks for hyperspectral image classification. *IEEE Trans. Geosci. Remote Sens.* **2020**, *59*, 5966–5978. [CrossRef]
17. Wan, S.; Gong, C.; Zhong, P.; Du, B.; Zhang, L.; Yang, J. Multiscale dynamic graph convolutional network for hyperspectral image classification. *IEEE Trans. Geosci. Remote Sens.* **2019**, *58*, 3162–3177. [CrossRef]

18. Yang, J.Y.; Li, H.C.; Hu, W.S.; Pan, L.; Du, Q. Adaptive cross-attention-driven spatial–spectral graph convolutional network for hyperspectral image classification. *IEEE Geosci. Remote Sens. Lett.* **2021**, *19*, 6004705. [CrossRef]
19. Yang, T.A.; Hy, T.S.; Dao, P.D. MOB-GCN: A Novel Multiscale Object-Based Graph Neural Network for Hyperspectral Image Classification. *arXiv* **2025**, arXiv:2502.16289.
20. Ahmad, M.; Butt, M.H.F.; Usama, M.; Mazzara, M.; Distefano, S.; Khan, A.M.; Hong, D. Hybrid State-Space and GRU-based Graph Tokenization Mamba for Hyperspectral Image Classification. *arXiv* **2025**, arXiv:2502.06427.
21. Yang, X.; Ye, Y.; Li, X.; Lau, R.Y.; Zhang, X.; Huang, X. Hyperspectral image classification with deep learning models. *IEEE Trans. Geosci. Remote Sens.* **2018**, *56*, 5408–5423. [CrossRef]
22. Alkhatib, M.Q.; Al-Saad, M.; Aburaed, N.; Almansoori, S.; Zabalza, J.; Marshall, S.; Al-Ahmad, H. Tri-CNN: A three branch model for hyperspectral image classification. *Remote Sens.* **2023**, *15*, 316. [CrossRef]
23. Ahmad, M.; Khan, A.M.; Mazzara, M.; Distefano, S.; Ali, M.; Sarfraz, M.S. A fast and compact 3-D CNN for hyperspectral image classification. *IEEE Geosci. Remote Sens. Lett.* **2020**, *19*, 5502205. [CrossRef]
24. Ge, Z.; Cao, G.; Li, X.; Fu, P. Hyperspectral image classification method based on 2D–3D CNN and multibranch feature fusion. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2020**, *13*, 5776–5788. [CrossRef]
25. Ye, A.; Zhou, X.; Miao, F. Innovative hyperspectral image classification approach using optimized CNN and ELM. *Electronics* **2022**, *11*, 775. [CrossRef]
26. Yu, C.; Han, R.; Song, M.; Liu, C.; Chang, C.I. Feedback attention-based dense CNN for hyperspectral image classification. *IEEE Trans. Geosci. Remote Sens.* **2021**, *60*, 5501916. [CrossRef]
27. Hang, R.; Li, Z.; Liu, Q.; Ghamisi, P.; Bhattacharyya, S.S. Hyperspectral image classification with attention-aided CNNs. *IEEE Trans. Geosci. Remote Sens.* **2020**, *59*, 2281–2293. [CrossRef]
28. Diakite, A.; Jiangsheng, G.; Xiaping, F. Hyperspectral image classification using 3D 2D CNN. *IET Image Process.* **2021**, *15*, 1083–1092. [CrossRef]
29. Munishamaiaha, K.; Kannan, S.K.; Venkatesan, D.; Jasiński, M.; Novak, F.; Gono, R.; Leonowicz, Z. Hyperspectral image classification with deep CNN using an enhanced elephant herding optimization for updating hyper-parameters. *Electronics* **2023**, *12*, 1157. [CrossRef]
30. Yang, X.; Zhang, X.; Ye, Y.; Lau, R.Y.; Lu, S.; Li, X.; Huang, X. Synergistic 2D/3D convolutional neural network for hyperspectral image classification. *Remote Sens.* **2020**, *12*, 2033. [CrossRef]
31. Sun, L.; Zhao, G.; Zheng, Y.; Wu, Z. Spectral–spatial feature tokenization transformer for hyperspectral image classification. *IEEE Trans. Geosci. Remote Sens.* **2022**, *60*, 5522214. [CrossRef]
32. Yang, X.; Cao, W.; Lu, Y.; Zhou, Y. Hyperspectral image transformer classification networks. *IEEE Trans. Geosci. Remote Sens.* **2022**, *60*, 5528715. [CrossRef]
33. Roy, S.K.; Deria, A.; Shah, C.; Haut, J.M.; Du, Q.; Plaza, A. Spectral–spatial morphological attention transformer for hyperspectral image classification. *IEEE Trans. Geosci. Remote Sens.* **2023**, *61*, 5503615. [CrossRef]
34. Zhang, J.; Sun, M.; Chang, S. Spatial and Spectral Structure-Aware Mamba Network for Hyperspectral Image Classification. *Remote Sens.* **2025**, *17*, 2489. [CrossRef]
35. Peng, H.; Lin, K.; Liu, H. HS-Mamba: Full-Field Interaction Multi-Groups Mamba for Hyperspectral Image Classification. *arXiv* **2025**, arXiv:2504.15612.
36. Jin, C.; Teng, X.; Chu, M.; Hao, Y.; Qin, S.; Li, X.; Yu, X. LDBMamba: Language-guided Dual-Branch Mamba for hyperspectral image domain generalization. *Expert Syst. Appl.* **2025**, *280*, 127620. [CrossRef]
37. Zhang, Y.; Jin, X.; Zhang, X.; Wu, Y.; Tu, L. EchoMamba: A new Mamba model for fast and efficient hyperspectral image classification. *PLoS ONE* **2025**, *20*, e0330678. [CrossRef] [PubMed]
38. Zhang, T.; Xuan, C.; Cheng, F.; Tang, Z.; Gao, X.; Song, Y. CenterMamba: Enhancing semantic representation with center-scan Mamba network for hyperspectral image classification. *Expert Syst. Appl.* **2025**, *287*, 127985. [CrossRef]
39. Gu, Q.; Luan, H.; Huang, K.; Sun, Y. Hyperspectral image classification using multi-scale lightweight transformer. *Electronics* **2024**, *13*, 949. [CrossRef]
40. Qin, A.; Shang, Z.; Tian, J.; Wang, Y.; Zhang, T.; Tang, Y.Y. Spectral–spatial graph convolutional networks for semisupervised hyperspectral image classification. *IEEE Geosci. Remote Sens. Lett.* **2018**, *16*, 241–245. [CrossRef]
41. Mou, L.; Lu, X.; Li, X.; Zhu, X.X. Nonlocal graph convolutional networks for hyperspectral image classification. *IEEE Trans. Geosci. Remote Sens.* **2020**, *58*, 8246–8257. [CrossRef]
42. Liu, Q.; Xiao, L.; Yang, J.; Wei, Z. CNN-enhanced graph convolutional network with pixel-and superpixel-level feature fusion for hyperspectral image classification. *IEEE Trans. Geosci. Remote Sens.* **2020**, *59*, 8657–8671. [CrossRef]
43. Zhang, Y.; Li, X.; Xu, Y. Pixel-level graph neural networks based on optimized feature representation for hyperspectral image classification. *IEEE Access* **2025**, *13*, 140830–140846. [CrossRef]
44. Shang, R.; Zhu, K.; Chang, H.; Zhang, W.; Feng, J.; Xu, S. Hyperspectral image classification based on mixed similarity graph convolutional network and pixel refinement. *Appl. Soft Comput.* **2025**, *170*, 112657. [CrossRef]

45. Chu, Y.; Cao, J.; Huang, J.; Ju, H.; Liu, G.; Cao, H.; Ding, W. Global-local graph convolutional broad network for hyperspectral image classification. *Appl. Soft Comput.* **2025**, *170*, 112723. [CrossRef]
46. Yao, D.; Zhi-li, Z.; Xiao-feng, Z.; Wei, C.; Fang, H.; Yao-ming, C.; Cai, W.W. Deep hybrid: Multi-graph neural network collaboration for hyperspectral image classification. *Def. Technol.* **2023**, *23*, 164–176. [CrossRef]
47. Jia, S.; Jiang, S.; Zhang, S.; Xu, M.; Jia, X. Graph-in-graph convolutional network for hyperspectral image classification. *IEEE Trans. Neural Netw. Learn. Syst.* **2022**, *35*, 1157–1171. [CrossRef]
48. Zheng, Z.; Debbagh, M.; Zhou, X.; Sun, S.; Huang, Y. Graph-Transformer with spatial-spectral features fusion for hyperspectral image classification. *Expert Syst. Appl.* **2025**, *264*, 125962. [CrossRef]
49. Fisher, R.A. The use of multiple measurements in taxonomic problems. *Ann. Eugen.* **1936**, *7*, 179–188. [CrossRef]
50. Achanta, R.; Shaji, A.; Smith, K.; Lucchi, A.; Fua, P.; Süsstrunk, S. SLIC superpixels compared to state-of-the-art superpixel methods. *IEEE Trans. Pattern Anal. Mach. Intell.* **2012**, *34*, 2274–2282. [CrossRef]
51. Kipf, T.N.; Welling, M. Semi-supervised classification with graph convolutional networks. *arXiv* **2016**, arXiv:1609.02907.
52. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, Ł.; Polosukhin, I. Attention is all you need. *Adv. Neural Inf. Process. Syst.* **2017**, *30*, 600–610.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.