Preprints.org

Article

Not peer-reviewed version

DESDA-Mamba: Diagonal-Enabled Strided Scanning and Direction-Adaptive Mamba for Hyperspectral Image Classiffcation

Chang Liu and Haibo Jin *

Posted Date: 28 May 2026

doi: 10.20944/preprints202605.1990.v1

Keywords: hyperspectral image classiffcation; mamba SSM; direction adaptation; strided scanning; diagonal scanning

Preprints.org is a free multidisciplinary platform providing preprint service that is dedicated to making early versions of research outputs permanently available and citable. Preprints posted at Preprints.org appear in Web of Science, Crossref, Google Scholar, Scilit, Europe PMC, OpenAlex.

Copyright: This open access article is published under a Creative Commons CC BY 4.0 license, which permit the free download, distribution, and reuse, provided that the author and preprint are cited in any reuse.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 28 May 2026 doi:10.20944/preprints202605.1990.v1

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.

Article

DESDA-Mamba: Diagonal-Enabled Strided Scanning and Direction-Adaptive Mamba for Hyperspectral Image Classification

Chang Liu and Haibo Jin *

School of Software, Liaoning Technique University, Huludao, Liaoning 125105, China
* Correspondence: jinhaibo@lntu.edu.cn

Abstract

Recently, Mamba based on State Space Models (SSMs) has shown great potential for hyperspectral image (HSI) classification due to its long-range modeling capability and linear complexity. However, existing Mamba-based methods usually employ fixed and limited scanning directions, restricting anisotropic spatial modeling. Moreover, full-pixel scanning introduces substantial computational redundancy. To address these issues, this paper proposes DESDA-Mamba, a direction-adaptive Mamba network with diagonal-enabled strided scanning for HSI classification. Specifically, a lightweight direction adaptation module is designed to implicitly predict suitable scanning directions from learned direction-sensitive feature-channel responses and perform batch-level unified direction aggregation, revealing that finer patch-level direction routing does not necessarily improve performance. In addition, a strided scanning strategy is introduced to skip redundant adjacent pixels during sequence serialization, reducing computational cost while enlarging the effective receptive field. Furthermore, two diagonal scanning modes, namely main-diagonal and anti-diagonal scanning, are proposed to improve the modeling of oblique spatial structures. Efficient diagonal scanning is implemented through coordinate-sequence indexing and caching mechanisms, enabling flexible diagonal strided scanning. Extensive comparison, ablation, and model-variant experiments on four public HSI datasets demonstrate that DESDA-Mamba achieves superior classification performance with competitive efficiency. The source code is available at https://github.com/ll-netizen/DESDA-MAMBA.

Keywords: hyperspectral image classiffcation; mamba SSM; direction adaptation; strided scanning; diagonal scanning

1. Introduction

In recent years, hyperspectral images (HSIs) have attracted increasing attention in earth science and remote sensing due to their rich spectral information provided by hundreds of contiguous bands. Unlike conventional RGB images with only three channels, HSIs record reflectance characteristics across dozens or even hundreds of continuous bands via imaging spectrometers [1]. Owing to distinct spectral response properties, HSIs can effectively distinguish visually similar objects by exploiting subtle spectral differences [2,3]. HSI classification assigns a semantic label to each pixel [4,5] and has been widely applied in tasks such as precision agriculture [6,7], urban planning [8,9], environmental monitoring [10,11], mineral exploration [12], and military reconnaissance [13].

Over the past two decades, numerous HSI classification methods have been proposed. Early approaches mainly relied on traditional machine learning techniques, such as support vector machines (SVMs) [12] and random forests (RFs) [14]. However, these methods depend on handcrafted features and thus have limited representation capability, making it difficult to fully exploit the rich spectral–spatial information in HSIs [16]. With the rapid development of deep learning, convolutional neural networks (CNNs) have been extensively adopted for HSI classification [17,18]. Lee et al. [19]

© 2026 by the author(s). Distributed under a Creative Commons CC BY license.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 28 May 2026 doi:10.20944/preprints202605.1990.v1

introduced 2D convolutions to jointly model spectral and spatial information. Hamida et al. [20] employed 3D CNNs to directly process spectral–spatial cubes. Ahmad et al. [21] designed a lightweight 3D CNN that reduces computational cost while maintaining competitive performance. Roy et al. [22] proposed HybridSN by cascading 3D and 2D convolutions. Dai et al. [23] further improved efficiency using hybrid 2D–3D depthwise separable convolutions. Nevertheless, conventional CNNs are limited by local receptive fields and struggle to capture long-range dependencies [24].

To overcome these limitations, Vision Transformers (ViTs) have been introduced into HSI classification. Benefiting from the self-attention mechanism, Transformers can model global contextual dependencies more effectively [25,26]. Hong et al. [27] proposed SpectralFormer, the first work to introduce Transformers into HSI classification by modeling spectral signatures as token sequences. Peng et al. [28] proposed a spectral–spatial Transformer with cross-attention. Mei et al. [29] designed a group-aware hierarchical Transformer to reduce computational complexity. Qi et al. [30] developed a global–local 3D convolutional Transformer network. For lightweight design, Zhao et al. [31] proposed GSC-ViT, which employs groupwise separable convolutions and self-attention to capture local spectral–spatial information while reducing parameters. Despite their superior performance, Transformer-based methods suffer from quadratic computational complexity, resulting in considerable memory and computational burdens for high-resolution HSIs.

Recently, State Space Models (SSMs) have provided a new direction for long-sequence modeling. Gu et al. [32] proposed the Structured State Space Sequence (S4) model, achieving efficient long-sequence modeling with linear complexity. However, conventional SSMs are linear time-invariant systems with fixed parameters, making it difficult to process information dynamically. To address this, Gu and Dao [33] proposed Mamba, which introduces an input-dependent selective SSM (S6), enabling model parameters to vary with the input. Thus, Mamba becomes content-aware: important information can be selectively retained over long ranges while irrelevant information is discarded. Inspired by this, Liu et al. [34] extended Mamba to 2D vision by proposing VMamba, which adopts a multi-directional scanning strategy and achieves competitive performance in image classification. In HSI classification, Mamba-based methods have also progressed rapidly. Li et al. [35] proposed MambaHSI, one of the earliest Mamba-based frameworks, designing spatial and spectral Mamba modules for joint representation learning. He et al. [36] proposed 3DSS-Mamba to process 3D hyperspectral cubes via a 3D spectral–spatial selective scanning mechanism. Sheng et al. [37] proposed DualMamba, a lightweight dual-stream Mamba–CNN architecture. Chen et al. [38] proposed RSMamba, which employs a dynamic multi-path activation mechanism to model non-causal spatial dependencies. Cao et al. [39] proposed DBMCMamba, integrating Mamba with a symmetric positive definite (SPD) manifold convolution module to capture spectral correlations through covariance representations.

Although existing Mamba-based methods have achieved promising performance, they suffer from three major limitations. First, they adopt fixed scanning directions for all inputs without adapting to diverse spatial structures. For instance, MambaHSI [36] divides features into four channel groups and flattens each in a row-wise manner, ignoring that different ground objects (e.g., straight roads, winding rivers, irregular farmlands) may require different scanning directions to preserve structural continuity. Such a fixed assumption is reasonable for 1D text but overly restrictive for images with anisotropic spatial characteristics. Second, existing methods commonly employ full-pixel scanning, incurring considerable computational overhead. Since adjacent pixels in HSIs are often highly similar, scanning all pixels introduces substantial redundancy, limiting deployment in resource-constrained scenarios such as onboard or airborne systems. Third, current methods are largely confined to horizontal and vertical scanning and lack true diagonal scanning. For example, VMamba [35] introduces four directions (left-to-right, right-to-left, top-to-bottom, bottom-to-top) but does not implement diagonal scanning, limiting its ability to capture oblique spatial structures.

To address these issues, we propose DESDA-Mamba, a Direction-Adaptive Mamba network with diagonal-enabled strided scanning for HSI classification. The framework introduces three innovations.First, a direction adaptation module dynamically predicts appropriate scanning directions

© 2026 by the author(s). Distributed under a Creative Commons CC BY license.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 28 May 2026 doi:10.20944/preprints202605.1990.v1

from learned direction-sensitive feature responses and assigns a unified direction to each feature group at the batch level, thereby enhancing anisotropic spatial modeling while improving directional robustness. Second, a strided scanning mechanism serializes features with a stride of 2, reducing sequence length to one-third of full-pixel scanning. This lowers computational burden while enlarging the effective receptive field and improving generalization. Third, a diagonal scanning strategy based on coordinate-sequence indexing and caching is proposed. Sampling-point coordinates are dynamically generated and sorted according to the scanning order, enabling efficient feature extraction. This extends the scanning direction space from horizontal and vertical to four directions: horizontal, vertical, main-diagonal, and anti-diagonal, enhancing Mamba's ability to represent complex spatial structures.

Extensive experiments on public HSI datasets demonstrate that DESDA-Mamba achieves superior classification performance while maintaining competitive computational efficiency. The main contributions are summarized as follows:

1. A direction adaptation module that dynamically predicts suitable scanning directions from learned direction-sensitive feature-channel responses, overcoming the limitation of fixed-direction scanning while revealing that finer patch-level routing is not necessarily more effective.
2. A strided scanning strategy that performs sparse sampling, reducing sequence length, computational cost, and redundancy while enlarging the receptive field.
3. Main-diagonal and anti-diagonal scanning strategies implemented via coordinate-sequence indexing, effectively extending Mamba's scanning direction space to model oblique spatial structures.

The remainder of this paper is organized as follows. Section 2 details the proposed DESDA-Mamba framework. Section 3 presents experimental settings and results. Section 4 concludes the paper and discusses future directions.

2. The Proposed DESDA-Mamba Model

This section provides a detailed exposition of our proposed DESDA-Mamba model. First, we elucidate the overall framework of the DESDA-Mamba model and its workffow for HSI classification. Second, we delve into the motivation and details of the three core mechanisms—direction adaptation, strided scanning, and diagonal scanning in the DESDA-Mamba architecture.

2.1. Overall Framework

The overall framework of DESDA-Mamba is illustrated in Figure 1. Initially, the hyperspectral image data undergoes preprocessing using principal component analysis (PCA) to reduce redundant spectral bands and lower computational cost for subsequent modules. The processed input is $I \in \mathbb{R}^{B\times C_{\text{in}}\times H\times W}$, where B, C_{in}, H, and W represent batch

© 2026 by the author(s). Distributed under a Creative Commons CC BY license.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 28 May 2026 doi:10.20944/preprints202605.1990.v1

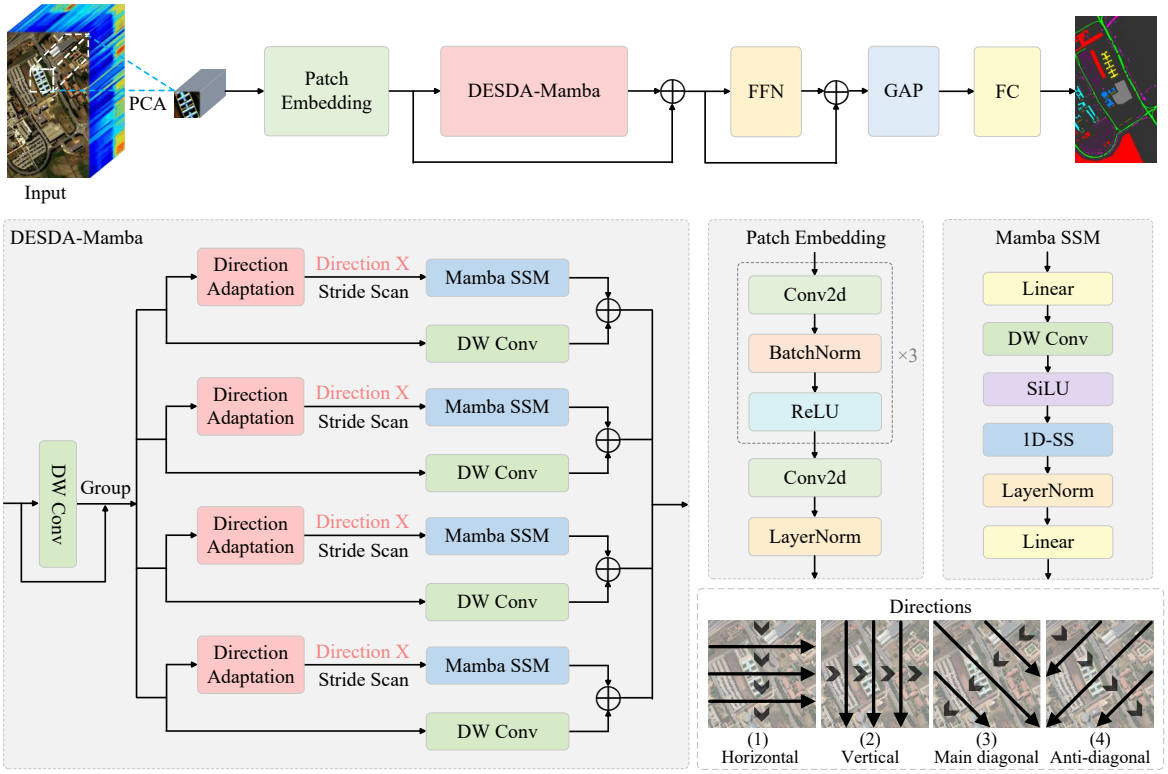


Figure 1. Overview of the proposed DESDA-Mamba framework, in which "Group" denotes the feature grouping operation along the channel dimension, "DW Conv" refers to depthwise convolution, "ReLU" and "SiLU" stands for the activation function, "Conv2d" denotes 2D convolutional layer, "Linear" is also known as a fully connected layer, "1D-SS" refers to one-dimensional selective scan. Schematic of the Direction Adaptation module is shown in Figure 2.

size, the reduced number of spectral bands, image height, and width, respectively. Next, the input passes through the patch embedding module, which consists of three consecutive convolutional layers to extract shallow spectral-spatial features progressively. A projection convolution layer then maps the feature dimension to the target embedding dimension, producing an output feature map of size $B \times C \times H \times W$, where C denotes the number of feature channels.

The feature map is then fed into the core DESDA-Mamba module. Residual-connected depthwise convolutions are first applied to inject local details and mitigate over-smoothing caused by successive global aggregation in the deep Mamba blocks. Since hyperspectral images contain dozens to hundreds of continuous spectral bands, different spectral band groups exhibit varying degrees of discrepancy and complementarity. For example, visible and near-infrared band groups show significantly different response patterns to vegetation, whereas adjacent bands are usually highly similar. To fully exploit the diversity among different spectral band groups and extract more fine-grained features, the feature map after residual depthwise convolution is permuted as $B \times C \times H \times W \rightarrow B \times H \times W \times C$ and evenly divided along the channel dimension into four groups, each with shape $B \times H \times W \times (C/4)$. Subsequently, each feature group enters an independent direction adaptation module, where a lightweight direction prediction network dynamically selects the optimal scanning direction based on the group's spatial characteristics. Four candidate directions

© 2026 by the author(s). Distributed under a Creative Commons CC BY license.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 28 May 2026 doi:10.20944/preprints202605.1990.v1

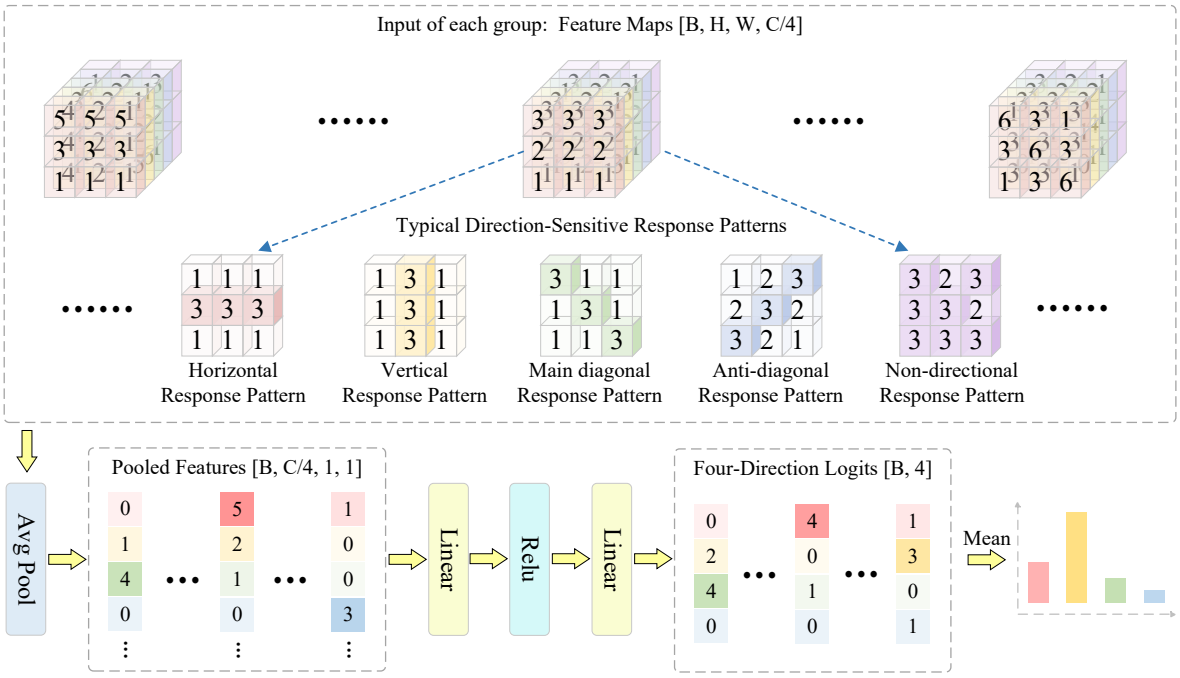


Figure 2. Schematic of the Direction Adaptation module, where several possible direction-sensitive response patterns are illustrated. Horizontal response patterns denote similar feature values within rows but large variations across rows, while the other directional patterns follow analogous directional consistency characteristics. Non-directional response patterns exhibit irregular spatial distributions without obvious directional consistency.

are illustrated in Figure 1 ("Directions" part). According to the selected direction, a strided scanning strategy with stride $s = 2$ is applied to transform the spatial feature map of each group into a serialized token sequence $T \in \mathbb{R}^{(B\times L)\times C/4}$, where $L = \lceil H/2 \rceil \times \lceil W/2 \rceil$.The serialized sequence is then processed by the Mamba SSM module to model long-range dependencies via the selective state space mechanism. The serialized features processed by Mamba are then reshaped back to the spatial layout and restored to the feature map format $B \times H \times W \times (C/4)$ through interpolation. Simultaneously, the original feature map of each group undergoes depthwise convolution in parallel to enhance local details. The resulting local features are element-wise added to the Mamba output, complementing global representations with local context. Finally, the output of four groups are concatenated along the channel dimension to restore the feature map to $B \times H \times W \times C$.

Subsequently, the feature map is flattened and fed into a feed-forward network for nonlinear transformation to enhance the representational capacity. A classification head consisting of global average pooling and a fully connected layer then projects the features into the category space to obtain the final classification results.

2.2. Direction Adaptation Module

Existing Mamba-based HSI classification methods usually employ fixed scanning directions for all inputs, which limits their ability to adapt to diverse spatial structures and semantic patterns. To address this issue, we propose a lightweight direction adaptation module that dynamically selects the most suitable scanning direction for each feature group according to the learned feature responses.

An overview of the proposed direction adaptation module is shown in Figure 2. For each feature group, the input feature map is represented as $X_g \in \mathbb{R}^{B\times H\times W\times C/4}$. Global average pooling is first applied to compress the spatial dimensions into channel-wise global descriptors, resulting in a tensor of shape $[B, C/4, 1, 1]$. Although this operation removes explicit spatial layouts, the pooled statistics still preserve implicit directional semantics encoded in learned feature responses. Specifically, during network optimization, certain feature channels gradually become sensitive to specific directional structures and exhibit corresponding direction-sensitive response patterns. These response patterns are not direct reflections of the original ground-object textures, but high-level semantic abstractions

© 2026 by the author(s). Distributed under a Creative Commons CC BY license.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 28 May 2026 doi:10.20944/preprints202605.1990.v1

progressively learned by the network. Consequently, the pooled channel statistics can implicitly encode directional semantic information, enabling effective direction prediction without explicitly modeling handcrafted geometric gradients or directional convolutions.

Subsequently, two fully connected layers are employed to learn the nonlinear mapping between feature statistics and directional preferences, producing the confidence scores of four candidate scanning directions with shape $[B, 4]$.The confidence scores are further averaged across the batch dimension to obtain the batch-level directional preference. This averaging strategy improves both robustness and computational efficiency, as it suppresses noisy decisions from individual patches and stabilizes the overall directional preference of the batch. Then, the direction index with the highest averaged confidence is selected as the final scanning direction. The direction prediction process is described as follows:

$$DirectionIndex = \arg\max\big(\mathrm{Mean}\big(\mathrm{FC}_2\big(\mathrm{ReLU}\big(\mathrm{FC}_1\big(\mathrm{Flatten}\big(\mathrm{AvgPool}(X_g)\big)\big)\big)\big)\big)\big) \tag{1}$$

In Section 3.5, we further analyze two direction-related variants, including a patch regrouping variant (V4) and an explicit directional feature extraction variant (V5). V4 assigns each patch a preferred direction using the same direction adaptation network as DESDA-Mamba and then regroups patches into four directional sub-groups for separate Mamba processing. V5 replaces implicit direction adaptation with explicit geometric priors, where four predefined directional convolution kernels (horizontal, vertical, main-diagonal, and anti-diagonal) are used to extract directional responses for direction prediction. Experimental results show that both variants fail to consistently improve classification accuracy compared with DESDA-Mamba, while introducing additional computational or structural overhead. These observations indicate that neither patch-level routing nor strongly handcrafted directional constraints lead to further performance gains. In contrast, the proposed batch-level implicit direction adaptation achieves a more effective balance between directional flexibility, robustness, and computational efficiency.

2.3. Strided Scanning Mechanism

Before global dependency modeling, the spatial feature map must be serialized. Existing Mamba-based methods adopt full-pixel scanning, which traverses all pixels. Although this ensures complete information coverage, it suffers from redundancy due to high similarity between adjacent pixels in HSI, leading to unnecessary computational overhead.

To alleviate this issue, a strided scanning mechanism is introduced in this work. Specifically, adjacent pixels are skipped with a fixed stride s to perform sparse sampling on the two-dimensional spatial domain, resulting in a sequence length of

$$L = \lceil H/s \rceil \times \lceil W/s \rceil \tag{2}$$

In this work, the patch size is set to 9×9 and the stride is fixed as $s = 2$. Under this setting, the sequence length of full-pixel scanning is $L' = 9 \times 9 = 81$, whereas the sequence length

© 2026 by the author(s). Distributed under a Creative Commons CC BY license.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 28 May 2026 doi:10.20944/preprints202605.1990.v1

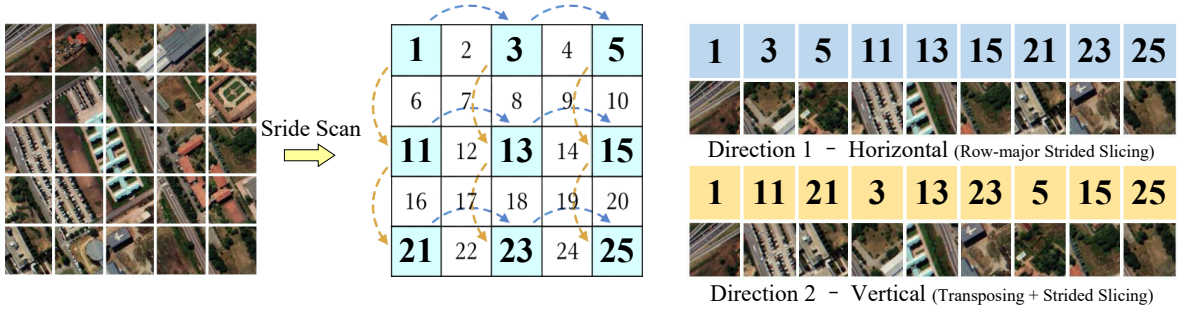


Figure 3. Visualization of horizontal and vertical strided scanning (non-actual implementation schematic), token sequence examples resulting from horizontal and vertical strided scanning of a 5×5 feature map are shown on the right.

of strided scanning becomes $L = \lceil 9/2 \rceil \times \lceil 9/2 \rceil = 5 \times 5 = 25$, thereby reducing the sequence length by approximately 69% compared with full-pixel scanning. Consequently, the computational cost is significantly reduced. Meanwhile, the spatial interval between adjacent sequence elements is enlarged to 2, thereby enabling the Mamba module to aggregate broader contextual information within fewer scanning steps.

Since the implementation of horizontal and vertical strided scanning cannot be directly extended to diagonal directions, a dedicated diagonal strided scanning strategy is proposed in this work, which will be introduced in the next subsection. The implementation details of horizontal and vertical strided scanning are first described below.

The visualization of horizontal and vertical strided scanning is illustrated in Figure 3. The core idea is to first transform the feature map along the spatial dimensions and then perform the same strided slicing operation. The complete procedure consists of five steps: feature map transformation, strided slicing, flattening, long-range dependency modeling, and interpolation restoration.

In the first step, feature map transformation is performed. For horizontal (left-to-right) scanning, no transformation is required because the default traversal order of the slicing operation already follows a row-major pattern, which naturally scans pixels from left to right and from top to bottom across rows. In contrast, vertical (top-to-bottom) scanning requires transposing the feature map, allowing the same slicing operation to achieve column-major traversal, namely scanning pixels from top to bottom and from left to right across columns.

In the second step, strided slicing is conducted. Let the transformed feature map be denoted as X_g with shape $B \times H \times W \times C/4$. The slicing operation is defined as

$$X'_g = X_g[:, :, 0 :: s, 0 :: s] \tag{3}$$

where the resulting feature map X'_g has shape $B \times \lceil H/s \rceil \times \lceil W/s \rceil \times C/4$. This operation starts from position $(0, 0)$ and samples one pixel after skipping every $s - 1$ adjacent pixels along both spatial dimensions.In the third step, the sampled feature map X'_g is flattened into a token sequence T with shape $(B \times L) \times (C/4)$, where $L = \lceil H/s \rceil \times \lceil W/s \rceil$. In the fourth step, the flattened token sequence T is fed into the Mamba operator for long-range dependency modeling.

In the fifth step, interpolation restoration recovers the original feature map shape $B \times H \times W \times C/4$ from the Mamba output. Each output token is first restored to its sampled position, reconstructing a sparse feature map of shape $B \times \lceil H/s \rceil \times \lceil W/s \rceil \times C/4$. Inverse transformation is then applied: none for horizontal scanning, and a transpose for vertical scanning. Finally, bilinear interpolation upsamples the sparse map to the original size, where sampled positions are directly filled with Mamba outputs and unsampled positions are estimated via weighted interpolation from their four nearest sampled neighbors (weights inversely proportional to spatial distance). This interpolation strategy enables the efficiency benefits of strided scanning while maintaining compatibility with subsequent modules.

© 2026 by the author(s). Distributed under a Creative Commons CC BY license.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 28 May 2026 doi:10.20944/preprints202605.1990.v1

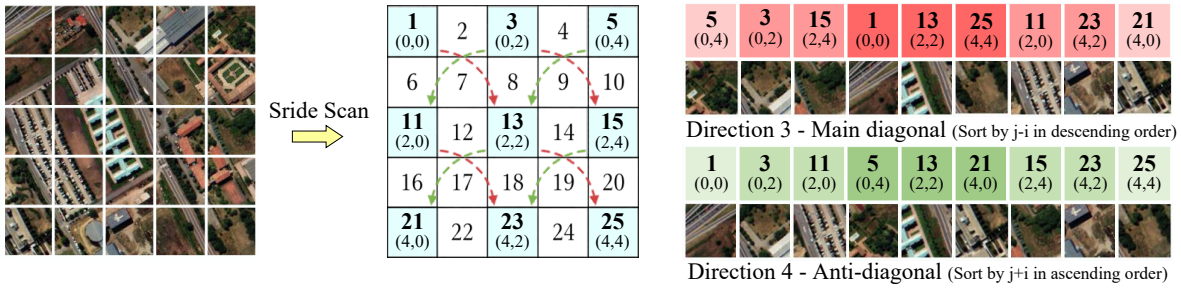


Figure 4. Visualization of main diagonal and anti-diagonal strided scanning (non-actual implementation schematic), in the token sequence on the right, points sharing the same shade of color lie in the same diagonal and are ordered by increasing row coordinate.

2.4. Diagonal Scanning Strategy

Conventional Mamba-based methods typically employ four scanning directions: horizontal (left-to-right), reverse horizontal (right-to-left), vertical (top-to-bottom), and reverse vertical (bottom-to-top). However, these are limited to horizontal and vertical orientations, restricting their ability to capture oblique spatial structures common in HSI, such as rivers, ridge lines, and farmland boundaries.

To address this limitation, this paper introduces two additional scanning directions: main-diagonal (upper-left to lower-right, switching along the anti-diagonal) and anti-diagonal (upper-right to lower-left, switching along the main diagonal). Together with the original horizontal and vertical directions, these form an extended scanning-direction space that enables more effective modeling of diverse spatial patterns. Moreover, the proposed diagonal scanning, implemented via coordinate-sequence indexing, is highly efficient and flexibly supports strided scanning by adjusting the stride value. Specifically, setting the stride $s = 1$ recovers full-pixel diagonal scanning, while $s > 1$ enables diagonal strided scanning.

Figure 4 visualizes the main-diagonal and anti-diagonal strided scanning. The core idea is to dynamically generate coordinate-sequence indices corresponding to the scanning order and gather feature values accordingly. The complete process comprises five steps: sampling-point coordinate generation, coordinate sorting, coordinate-sequence indexing and value gathering, long-range dependency modeling, and interpolation restoration.

In the first step, sampling-point coordinates are generated. Given an input feature map with spatial size $H \times W$ and stride s, the sampling-point set can be formulated as

$$P = \{(i, j) \mid i \in \{0, s, 2s, \dots\},\ j \in \{0, s, 2s, \dots\}\} \tag{4}$$

in which i denotes row coordinate and j denotes column coordinate. In this work, strid $s = 2$.

In the second step, the sampling-point coordinates are sorted according to the diagonal scanning order. For main diagonal scanning, the sampling-point coordinates are sorted in descending order of $(j - i)$, as the value $(j - i)$ reflects the deviation degree from the main diagonal. Larger $(j - i)$ values correspond to positions closer to the upper-right corner, whereas smaller values correspond to positions closer to the lower-left corner. For points with identical $(j - i)$ values, namely points located on the same main diagonal line, the row coordinate i is further sorted in ascending order, corresponding to sampling along the upper-left to lower-right direction within each scanning line. For the 5×5 example in Figure 4, the resulting main diagonal scanning coordinate sequence is $[(0,4), (0,2), (2,4), (0,0), (2,2), (4,4), (2,0), (4,2), (4,0)]$. For anti-diagonal scanning, the sampling points are sorted in ascending order of $(i + j)$, as the value $(i + j)$ reflects the deviation degree from the anti-diagonal. Smaller $(i + j)$ values correspond to positions closer to the upper-left corner, whereas larger values correspond to positions closer to the lower-right corner. For points with identical $(i + j)$ values, namely points located on the same anti-diagonal line, the row coordinate i is further sorted in ascending order, corresponding to sampling along the upper-right to lower-left direction within

© 2026 by the author(s). Distributed under a Creative Commons CC BY license.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 28 May 2026 doi:10.20944/preprints202605.1990.v1

each scanning line. For the 5×5 example in Figure 4, the resulting anti-diagonal scanning coordinate sequence is $[(0,0),(0,2),(2,0),(0,4),(2,2),(4,0),(2,4),(4,2),(4,4)]$.

In the third step, coordinate-sequence indexing and value gathering are performed. After obtaining the reordered coordinate sequence, a linear-index transformation is applied to enable fully vectorized gathering. Specifically, the feature map is flattened in row-major order, and the feature value at coordinate (i,j) corresponds to the linear index

$$idx = i \times W + j \tag{5}$$

Using the 5×5 example, the index sequence for main-diagonal scanning is $[4,2,14,0,12,24,10,22,20]$, while that for anti-diagonal scanning is $[0,2,10,4,12,20,14,22,24]$. All feature values are then gathered simultaneously via vectorized indexing, which is significantly more efficient than point-by-point traversal.

Since the coordinate sequences and index lists remain unchanged for a given feature-map size, stride, and scanning direction, a caching mechanism is introduced. The index list is stored in cache upon its first invocation and directly reused for subsequent calls with the same configuration, avoiding redundant computation. The fourth and fifth steps are nearly identical to those of horizontal and vertical strided scanning and are thus omitted.

3. Experiments

In this section, we first describe the experimental setup, including datasets, evaluation metrics, comparison methods, and implementation details. Then, we comprehensively compare with SOTAs methods both qualitatively and quantitatively. Finally, we conduct a detailed ablation study to analyze the effect of the proposed modules.

3.1. Data Description

Four public hyperspectral image datasets are employed to evaluate the proposed DESDA-Mamba model, including University of Pavia, Salinas, Indian Pines, and WHU-Hi-LongKou.

1. University of Pavia dataset: The University of Pavia dataset was acquired by the Reflective Optics System Imaging Spectrometer (ROSIS) sensor over the University of Pavia, Italy, consisting of 103 spectral bands with a spatial size of 610×340. The dataset contains 42,776 labeled pixels of 9 classes. The samples are randomly split into 1% as the training set, 49.5% as the validation set, and 49.5% as the testing set. Figure **??** illustrates the category and sample settings, false color image, and ground truth map.
2. Indian Pines dataset: The Indian Pines dataset was acquired by the NASA AVIRIS sensor over northwestern Indiana, USA, consisting of 200 spectral bands with a spatial size of 145×145. The dataset contains 10,249 labeled pixels of 16 classes.

Table 1. University of Pavia dataset (train ratio 1%): Category and sample settings.

ID	Color	Category	Train	Valid	Test
1		Asphalt	66	3,283	3,282
2		Meadows	186	9,232	9,231
3		Gravel	21	1,039	1,039
4		Trees	31	1,517	1,516
5		Metal sheets	13	666	666
6		Bare soil	50	2,490	2,489
7		Bitumen	13	658	659
8		Bricks	37	1,823	1,822
9		Shadows	9	469	469
Total		42,776	426	21,177	21,173

© 2026 by the author(s). Distributed under a Creative Commons CC BY license.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 28 May 2026 doi:10.20944/preprints202605.1990.v1

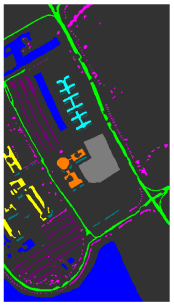
The samples are randomly split into 10% as the training set, 45% as the validation set, and 45% as the testing set. Figure **??** illustrates the detailed information.

3. Salinas dataset: The Salinas dataset was collected by the Airborne Visible/Infrared Imaging Spectrometer (AVIRIS) sensor over Salinas Valley, California, USA, consisting of 204 spectral bands with a spatial size of 512 × 217. The dataset contains 54,129 labeled pixels of 16 classes. The samples are randomly split into 1% as the training set, 49.5% as the validation set, and 49.5% as the testing set. Figure 5 illustrates the detailed information.
4. WHU-Hi-LongKou dataset: The WHU-Hi-LongKou dataset was collected by the Headwall Nano-Hyperspec sensor over LongKou Town, Hubei Province, China, consisting of 270 spectral bands with a spatial size of 550 × 400. The dataset contains 204,542 labeled pixels of 9 classes. The samples are randomly split into 0.2% as the training set, 49.9% as the validation set, and 49.9% as the testing set. Figure 6 illustrates the detailed information.

ID	Color	Category	Train	Valid	Test
1		Asphalt	66	3,283	3,282
2		Meadows	186	9,232	9,231
3		Gravel	21	1,039	1,039
4		Trees	31	1,517	1,516
5		Metal sheets	13	666	666
6		Bare soil	50	2,490	2,489
7		Bitumen	13	658	659
8		Bricks	37	1,823	1,822
9		Shadows	9	469	469
Total		42,776	426	21,177	21,173

(a)

(b)

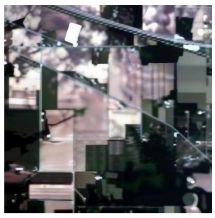


(c)

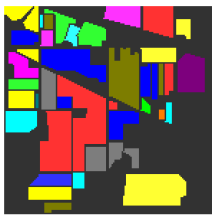
Figure 5. Salinas dataset (train ratio 1%): (a) Category and sample settings; (b) False color image; (c) Ground truth.

ID	Color	Category	Train	Valid	Test
1		Alfalfa	5	21	20
2		Corn-notill	143	643	642
3		Corn-mintill	83	374	373
4		Corn	24	107	106
5		Grass-pasture	48	218	217
6		Grass-trees	73	329	328
7		Grass-pasture-mowed	3	13	12
8		Hay-windrowed	48	215	215
9		Oats	2	9	9
10		Soybean-notill	97	438	437
11		Soybean-mintill	246	1,105	1,104
12		Soybean-clean	59	267	267
13		Wheat	21	92	92
14		Woods	127	569	569
15		Buildings-Grass-Trees-Drives	39	174	173
16		Stone-Steel-Towers	9	42	42
Total		10,249	1,025	4,616	4,608

(a)



(b)



(c)

Figure 6. WHU-Hi-LongKou dataset (train ratio 0.2%): (a) Category and sample settings; (b) False color image; (c) Ground truth.

3.2. *Experimental Settings and Assessment Criteria*

The input patch size is uniformly set to 9 × 9. PCA is applied to all compared models for spectral dimensionality reduction. The number of retained principal components is set to 15 for University of Pavia, 30 for Salinas and Indian Pines, and 20 for WHU-Hi-LongKou. The AdamW optimizer is adopted for model optimization, with an initial learning rate of 1×10^{-3} and a weight decay coefficient of 0.05. A cosine annealing warm restart strategy is employed for learning-rate scheduling, where $T_0 = 50$, $T_{\text{mult}} = 2$, and the minimum learning rate is set to 1×10^{-6}. All models are trained for 200 epochs with a batch size of 16. The standard cross-entropy loss is adopted as the loss function. All experiments are implemented using PyTorch 2.5.1 and Python 3.12 under the Ubuntu 22.04 operating

© 2026 by the author(s). Distributed under a Creative Commons CC BY license.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 28 May 2026 doi:10.20944/preprints202605.1990.v1

system. The hardware platform consists of a single NVIDIA GeForce RTX 4090 GPU with 24 GB memory, an Intel Xeon Platinum 8352V CPU with 16 vCPUs at 2.10 GHz, and 120 GB system memory.

Three standard metrics are adopted to evaluate classification performance, including overall accuracy (OA), average accuracy (AA), and the Kappa coefficient. OA represents the proportion of correctly classified pixels among all testing samples, AA denotes the average classification accuracy across all categories, and the Kappa coefficient measures the consistency between classification results and ground-truth labels. Higher values of these metrics indicate better classification performance. To evaluate model stability, each experiment is repeated 10 times, and the average results together with standard deviations are reported.

Four metrics are further employed to evaluate model complexity and efficiency, including the number of parameters (Params), floating-point operations (FLOPs), training time (Train Time), and testing time (Test Time). Params represents the total number of trainable parameters, reflecting the storage requirement of the model. FLOPs denote the number of floating-point operations required for one forward propagation, reflecting computational complexity. Train Time refers to the average time required to complete one training epoch, excluding validation time, averaged over 200 epochs. Test Time denotes the inference time for the entire testing set, reflecting practical deployment efficiency. Lower Params and FLOPs indicate a more lightweight model, while shorter Train Time and Test Time indicate higher operational efficiency.

3.3. Classiffcation Comparison With State-of-the-Art Models

To validate the effectiveness of DESDA-Mamba, we compare it against eight state-of-the-art methods: three CNN-based (LSSAN [42], Lite-HCNet [43], MSDAN [41]), four Transformer-based (SSFTT [40], SimPoolFormer [44], SpectralFormer [27], GSC-ViT [31]), and one SSM-based (MambaHSI [35]). Brief descriptions are as follows:

1. LSSAN [42]: Uses MobileNetV3 as a lightweight feature extractor with a simplified spatial attention module and multi-class focal loss to balance sample difficulty.
2. Lite-HCNet [43]: Designs lightweight depthwise separable convolution via channel attention and kernel decomposition, with an efficient hybrid convolution block bridging 3D and 2D CNNs.
3. MSDAN [41]: Employs multi-scale densely connected modules for feature reuse, with spectral-spatial-channel attention to enhance discriminative representation while reducing parameters.
4. SSFTT [40]: Extracts shallow spectral-spatial features, uses a Gaussian-weighted feature tokenizer to generate token representations, and processes them with a Transformer encoder.
5. SimPoolFormer [44]: Adopts a dual-stream architecture, replacing heavy multi-head self-attention with SimPool for efficiency, and incorporates ResMLP in the second stream to boost performance.
6. SpectralFormer [27]: Models HSI classification as sequence learning, extracting local spectral sequences from adjacent bands with cross-layer skip connections to preserve deep representations.
7. GSC-ViT [31]: Introduces grouped separable convolution and grouped separable multi-head self-attention to efficiently capture local and global spectral-spatial features with fewer parameters.
8. MambaHSI [35]: Designs spatial and spectral Mamba blocks to model long-range interactions and spectral correlations, followed by an adaptive spatial-spectral fusion module.

Tables 2–5 report the comparison results on four HSI datasets, with the best results highlighted in bold. DESDA-Mamba achieves the best overall performance on the University of Pavia, Indian Pines, and Salinas datasets, and ranks second on the WHU-Hi-LongKou dataset, only 0.06% lower than GSC-ViT under the extremely limited 0.2% training ratio setting. In addition, DESDA-Mamba generally exhibits relatively small standard deviations across repeated experiments, indicating stable optimization behavior and robust generalization under limited training conditions.

© 2026 by the author(s). Distributed under a Creative Commons CC BY license.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 28 May 2026 doi:10.20944/preprints202605.1990.v1

Table 2. Comparison results of different methods on the University of Pavia dataset (train ratio 1%).

Class	CNN-based Methods			Transformer-based Methods				SSM-based Methods	
	LSSAN	Lite-HCNet	MSDAN	SSFTT	SimPoolFormer	SpectralFormer	GSC-ViT	MambaHSI	DESDA-Mamba
1	92.01±2.90	97.27±1.28	96.18±0.89	98.21±0.55	92.77±1.09	92.15±1.33	98.64±0.65	98.99±0.46	**99.17±0.31**
2	99.29±0.34	99.80±0.16	99.61±0.20	99.71±0.16	99.20±0.19	99.02±0.38	99.74±0.35	99.85±0.08	**99.98±0.03**
3	71.23±8.83	89.47±2.09	90.91±2.95	89.79±3.29	84.00±1.29	75.41±5.23	90.52±3.34	**94.81±1.01**	92.03±1.78
4	82.08±3.16	93.64±1.26	94.37±1.54	95.96±1.74	88.53±0.80	88.71±2.11	95.46±1.72	95.06±1.06	**96.42±0.92**
5	95.11±4.29	99.09±0.37	**99.99±0.05**	99.82±0.25	98.07±0.31	99.79±0.37	98.65±1.08	99.61±0.28	99.54±0.29
6	96.66±1.45	99.16±0.69	99.55±0.47	98.39±0.96	90.89±1.45	95.27±1.80	98.86±0.60	**99.96±0.07**	99.15±0.77
7	85.03±6.08	98.91±1.35	98.56±1.05	98.42±0.91	89.13±4.48	83.92±8.97	99.88±0.16	**99.94±0.08**	99.73±0.37
8	71.37±6.40	90.90±1.99	95.40±1.10	94.83±2.29	77.85±0.92	83.93±1.95	97.00±1.04	95.91±0.60	**97.85±1.34**
9	83.95±11.06	97.37±1.30	93.80±2.49	**99.96±0.09**	86.35±3.50	91.26±3.57	98.59±0.71	95.58±1.48	98.70±0.85
OA	91.93±1.26	97.52±0.26	97.76±0.22	98.11±0.49	93.24±0.40	93.70±0.74	98.41±0.22	98.70±0.12	**98.87±0.18**
AA	86.30±2.53	96.18±0.36	96.48±0.40	97.23±0.66	89.64±0.95	89.94±1.55	97.48±0.43	97.75±0.27	**98.06±0.24**
Kappa	89.27±1.67	96.70±0.35	97.03±0.30	97.50±0.65	90.99±0.53	91.62±0.99	97.90±0.30	98.27±0.16	**98.51±0.24**
Params/M	0.04	**0.003**	3.34	0.08	1.47	0.05	0.13	0.12	0.03
FLOPs/M	1.92	**1.09**	1460	3.52	22.23	1.5	9.21	4.65	5.17
Train Time/s	0.22	1.96	3.64	**0.13**	0.36	0.28	0.33	0.51	0.47
Test Time/s	0.44	2.96	5.94	**0.31**	0.47	0.43	0.52	1.26	0.85

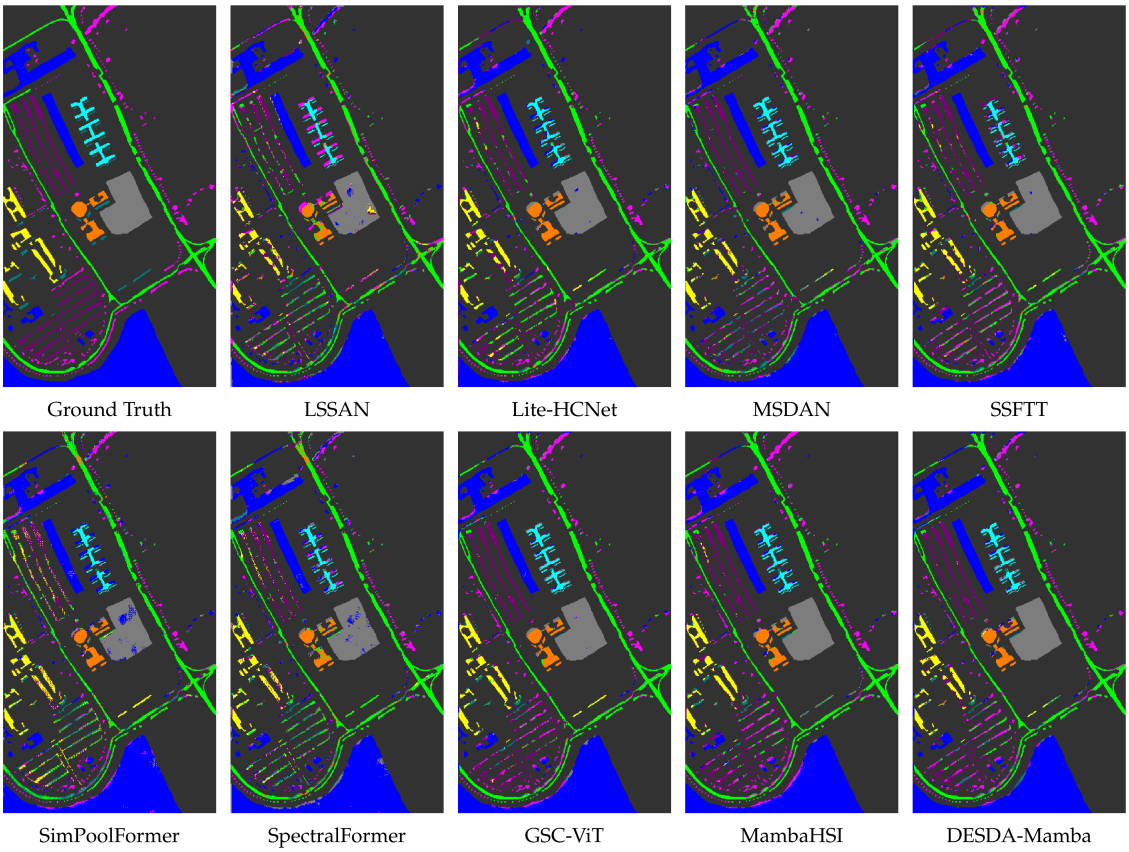


Figure 7. Ground Truth and classification maps of different methods on University of Pavia.

© 2026 by the author(s). Distributed under a Creative Commons CC BY license.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 28 May 2026 doi:10.20944/preprints202605.1990.v1

Table 3. Comparison results of different methods on the Indian Pines dataset (train ratio 10%).

Class	CNN-based Methods			Transformer-based Methods				SSM-based Methods	
	LSSAN	Lite-HCNet	MSDAN	SSFTT	SimPoolFormer	SpectralFormer	GSC-ViT	MambaHSI	DESDA-Mamba
1	54.00±23.55	98.50±2.42	**100.00±0.00**	86.00±8.76	90.50±6.43	66.00±18.38	97.00±4.22	95.50±5.50	92.50±8.90
2	88.16±2.24	93.39±0.97	93.75±1.31	93.28±0.67	83.72±0.89	80.37±1.81	95.13±0.77	93.84±0.72	**95.64±0.99**
3	90.75±4.81	99.71±0.20	99.63±0.36	98.80±1.12	88.93±0.86	83.34±3.53	**99.87±0.29**	99.55±0.25	99.79±0.33
4	78.40±10.44	97.36±1.24	96.04±2.70	91.61±2.15	77.74±5.27	66.51±6.96	**99.25±1.16**	98.30±1.59	98.21±1.69
5	92.29±4.57	99.49±0.34	99.13±0.93	99.49±0.46	95.27±0.75	94.40±1.66	98.53±0.80	**99.86±0.22**	99.40±0.44
6	96.78±1.98	98.21±0.74	99.91±0.14	99.42±0.34	97.69±0.92	98.36±0.58	99.76±0.32	**99.94±0.19**	98.94±0.36
7	81.67±19.16	99.17±2.63	**100.00±0.00**	99.17±2.63	86.67±5.83	89.17±11.82	**100.00±0.00**	**100.00±0.00**	99.17±2.63
8	99.67±0.23	99.95±0.15	**100.00±0.00**	99.44±0.81	99.63±0.75	99.77±0.33	**100.00±0.00**	**100.00±0.00**	**100.00±0.00**
9	66.67±24.57	**100.00±0.00**	95.56±7.77	83.33±21.76	58.89±11.77	91.11±11.47	83.34±14.10	96.67±5.37	97.78±7.03
10	90.75±2.15	97.33±0.79	98.18±0.64	96.49±2.28	89.06±1.07	86.67±2.66	98.13±0.89	98.15±0.64	**98.49±0.52**
11	95.28±1.38	99.11±0.32	98.91±0.43	97.91±0.74	94.55±0.54	91.41±1.35	99.46±0.26	**99.53±0.16**	99.41±0.25
12	87.12±4.42	97.15±0.96	97.68±0.72	94.64±2.97	82.47±2.54	73.41±5.11	**98.92±0.67**	97.79±0.78	98.69±0.48
13	95.54±2.92	99.56±0.56	99.89±0.34	**100.00±0.00**	99.89±0.34	99.35±1.17	99.78±0.46	98.37±0.57	98.37±1.38
14	97.28±1.48	99.93±0.09	99.96±0.08	99.75±0.20	98.70±0.49	96.95±0.76	99.89±0.12	99.91±0.12	**99.98±0.06**
15	90.06±4.14	**99.77±0.30**	99.31±0.76	97.00±2.26	96.47±1.76	89.54±5.42	99.08±0.95	98.61±0.91	99.48±0.74
16	85.00±7.94	95.00±4.95	**96.19±2.79**	89.29±5.97	90.24±4.69	85.00±11.28	92.86±2.75	92.15±3.19	92.86±2.51
OA	92.50±1.11	98.13±0.15	98.32±0.25	97.20±0.40	92.02±0.35	88.78±0.47	98.67±0.16	98.45±0.15	**98.71±0.15**
AA	86.84±2.94	98.35±0.31	**98.38±0.51**	95.35±1.82	89.40±1.03	86.96±1.16	97.56±1.07	98.01±0.61	98.05±0.63
Kappa	91.44±1.27	97.86±0.17	98.08±0.28	96.81±0.45	90.88±0.40	87.17±0.54	98.49±0.18	98.23±0.17	**98.53±0.17**
Params/M	0.05	**0.01**	3.63	0.15	1.54	0.05	0.13	0.12	0.05
FLOPs/M	3.22	**2.63**	2440	7.35	23.48	2.9	9.83	4.81	8.21
Train Time/s	0.49	7.62	10.79	**0.37**	0.91	0.70	0.80	1.00	1.08
Test Time/s	0.14	1.01	2.53	**0.11**	0.28	0.14	0.16	0.28	0.27

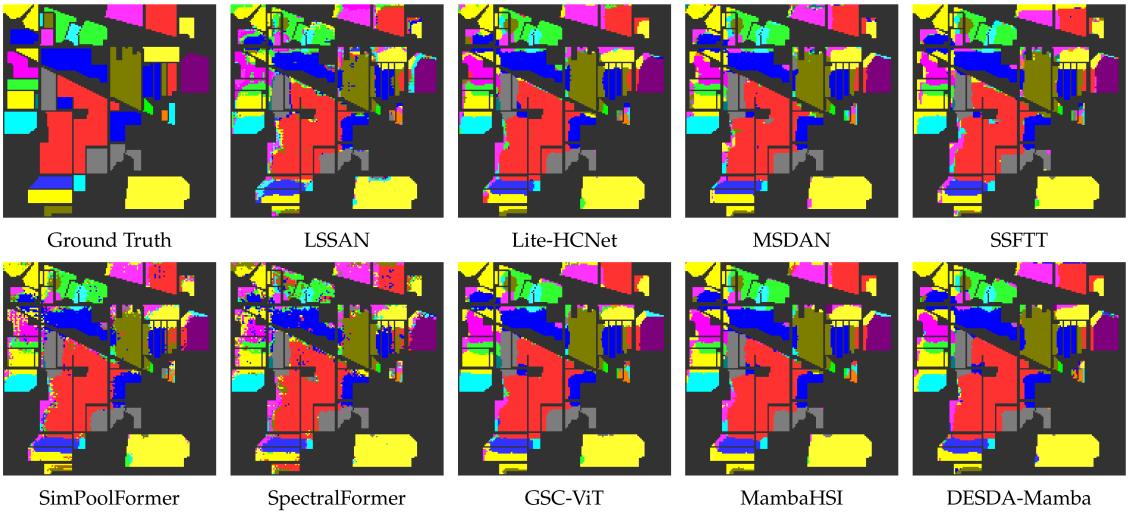


Figure 8. Ground Truth and classification maps of different methods on the Indian Pines dataset.

In terms of classification accuracy, DESDA-Mamba achieves the best overall performance on three datasets. On the University of Pavia dataset, using only 1% training samples, DESDA-Mamba attains an OA of 98.87±0.18%, outperforming the second-best MambaHSI and third-best GSC-ViT by 0.17% and 0.46%, respectively. DESDA-Mamba also achieves the highest AA and Kappa values, demonstrating more balanced category-wise classification performance. Although MambaHSI achieves slightly higher accuracy in several individual categories, DESDA-Mamba yields superior overall accuracy and lower fluctuation across repeated runs, indicating stronger robustness and generalization capability under small-sample conditions.

© 2026 by the author(s). Distributed under a Creative Commons CC BY license.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 28 May 2026 doi:10.20944/preprints202605.1990.v1

Table 4. Comparison results of different methods on the Salinas dataset (train ratio 1%).

Class	CNN-based Methods			Transformer-based Methods				SSM-based Methods	
	LSSAN	Lite-HCNet	MSDAN	SSFTT	SimPoolFormer	SpectralFormer	GSC-ViT	MambaHSI	DESDA-Mamba
1	95.82±3.10	99.77±0.70	**100.00±0.00**	99.66±0.50	99.91±0.29	98.69±1.26	**100.00±0.00**	**100.00±0.00**	99.91±0.17
2	99.80±0.27	99.99±0.02	**100.00±0.00**	99.98±0.07	**100.00±0.00**	99.95±0.12	**100.00±0.00**	**100.00±0.00**	**100.00±0.00**
3	97.88±1.84	99.89±0.10	**100.00±0.00**	99.97±0.07	**100.00±0.00**	99.81±0.22	99.98±0.06	**100.00±0.00**	99.95±0.07
4	98.79±1.07	99.32±0.37	99.85±0.11	99.90±0.23	98.81±0.33	98.60±1.84	**99.94±0.12**	99.67±0.19	99.86±0.18
5	96.95±1.30	96.41±0.71	96.75±0.52	98.82±0.92	97.20±0.66	**99.02±0.59**	97.70±1.19	97.53±0.87	98.89±0.68
6	99.13±1.05	99.76±0.40	**100.00±0.00**	99.95±0.11	99.93±0.12	99.99±0.02	99.92±0.15	99.98±0.04	99.96±0.08
7	97.93±1.52	99.61±0.40	99.98±0.04	99.78±0.15	99.99±0.03	99.63±0.28	99.89±0.10	**100.00±0.00**	99.95±0.08
8	95.61±2.57	98.40±0.31	97.82±0.44	96.79±0.66	95.65±0.67	93.94±1.13	98.90±0.44	**99.36±0.20**	98.99±0.27
9	99.83±0.15	**100.00±0.00**	**100.00±0.00**	**100.00±0.00**	**100.00±0.00**	99.96±0.05	**100.00±0.00**	**100.00±0.00**	**100.00±0.00**
10	94.68±3.03	98.84±0.56	99.31±0.49	99.12±0.44	98.24±0.90	97.21±1.18	99.77±0.23	**99.82±0.16**	99.66±0.26
11	93.10±4.87	97.90±1.19	**99.96±0.08**	98.11±1.13	97.67±3.03	97.86±3.29	99.34±0.60	99.79±0.11	98.70±1.15
12	97.98±2.05	**100.00±0.00**	99.70±0.32	99.90±0.27	99.96±0.10	99.22±0.70	99.97±0.05	**100.00±0.00**	99.99±0.03
13	95.26±3.48	97.18±0.82	97.82±1.29	97.95±2.18	90.53±3.23	96.50±2.69	**98.48±1.50**	94.69±1.27	98.33±1.84
14	93.93±2.97	98.43±0.44	98.49±0.50	97.94±0.81	98.62±0.20	96.09±2.12	**99.00±0.41**	98.49±0.13	98.83±0.36
15	91.62±2.23	97.24±0.82	94.99±0.88	90.92±1.21	91.40±0.69	89.24±2.82	97.04±0.82	97.36±0.70	**97.47±0.80**
16	94.37±3.21	99.24±0.18	99.44±0.17	98.85±1.11	99.56±0.21	98.75±1.46	99.62±0.24	99.77±0.16	**99.80±0.21**
OA	96.39±0.97	98.83±0.12	98.57±0.15	97.81±0.18	97.41±0.17	96.70±0.38	99.16±0.12	99.24±0.09	**99.27±0.11**
AA	96.42±0.86	98.87±0.12	99.01±0.13	98.60±0.18	97.97±0.29	97.78±0.24	99.35±0.14	99.15±0.07	**99.39±0.18**
Kappa	95.99±1.08	98.69±0.14	98.41±0.17	97.56±0.20	97.11±0.18	96.33±0.42	99.06±0.13	99.15±0.10	**99.19±0.12**
Params/M	0.05	**0.01**	3.63	0.15	1.54	0.05	0.13	0.12	0.05
FLOPs/M	3.22	**2.63**	2440	7.35	23.48	2.9	9.83	4.81	8.21
Train Time/s	0.26	2.79	5.44	**0.17**	0.47	0.36	0.42	0.65	0.53
Test Time/s	0.60	5.46	15.52	**0.46**	0.74	0.61	0.80	1.64	0.87

On the Indian Pines dataset, DESDA-Mamba achieves the highest OA of 98.71±0.15% and the highest Kappa coefficient of 98.53±0.17%. Although the margin over GSC-ViT is relatively small, DESDA-Mamba still surpasses MambaHSI by 0.26% in OA while maintaining competitive AA performance. Considering that many categories on Indian Pines are already close to saturation for advanced models, these improvements still demonstrate the effectiveness of the proposed directional modeling strategy.

On the Salinas dataset, DESDA-Mamba achieves the best OA, AA, and Kappa values of 99.27±0.11%, 99.39±0.18%, and 99.19±0.12%, respectively. In particular, the AA surpasses the second-best MambaHSI by 0.24%, indicating more balanced performance across different land-cover categories. These improvements can be attributed to the integration of diagonal scanning and direction-adaptive selection, which enhances the modeling capability for diverse spatial structures, while the proposed strided scanning enlarges the effective receptive field and alleviates redundant local sampling, thereby improving generalization performance.

With respect to computational efficiency, DESDA-Mamba maintains competitive training and inference costs while consistently improving classification performance. On the University of Pavia dataset, DESDA-Mamba requires 0.47 s training time per epoch and 0.85 s testing time, remaining close to MambaHSI in training efficiency while reducing inference time by approximately one-third. On the Salinas dataset, DESDA-Mamba further improves efficiency, achieving 0.53 s training time and 0.87 s testing time, both lower than those of MambaHSI. Although DESDA-Mamba is slightly slower than the lightweight Transformer model GSC-ViT on Indian Pines and WHU-Hi-LongKou, it still remains substantially more efficient than heavy CNN-based models such as MSDAN while achieving superior or highly competitive classification accuracy.

Overall, although DESDA-Mamba introduces additional directional modeling operations, the computational overhead remains well controlled. Compared with existing SSM- and Transformer-based methods, DESDA-Mamba achieves a favorable trade-off between representation capability, classification robustness, and computational efficiency.

© 2026 by the author(s). Distributed under a Creative Commons CC BY license.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 28 May 2026 doi:10.20944/preprints202605.1990.v1

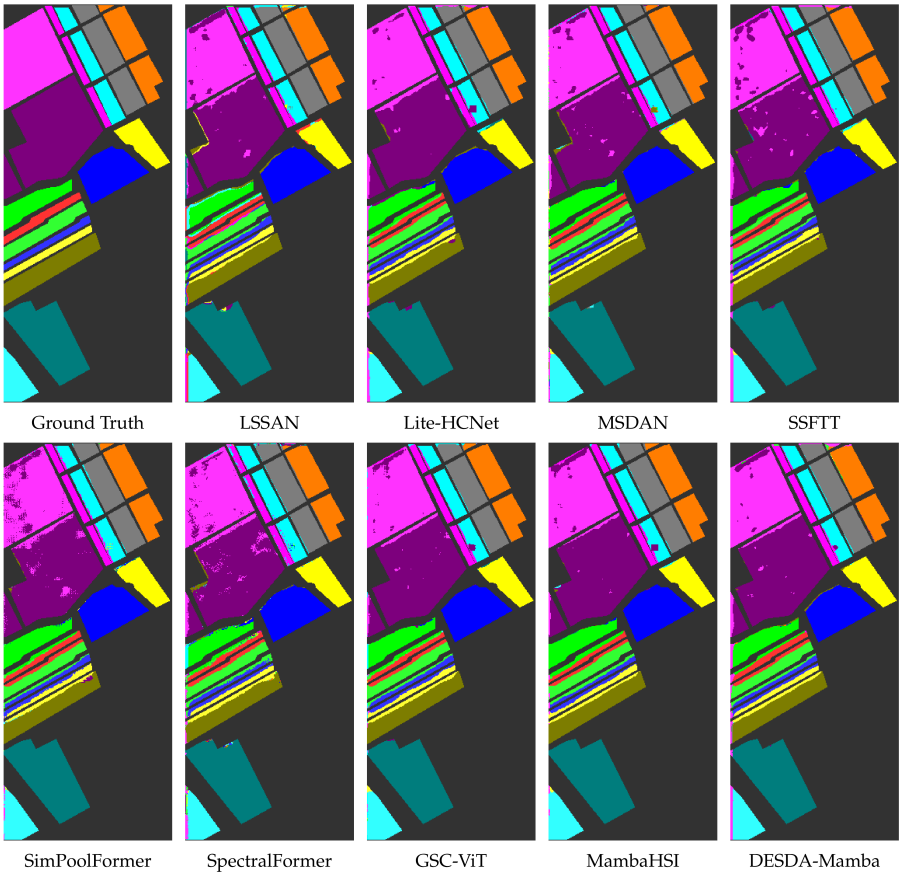


Figure 9. Ground Truth and classification maps of different methods on the Salinas dataset.

Figures 7–10 present the corresponding classification maps with the highest OA among ten repeated runs. From the classification maps, DESDA-Mamba produces spatially smoother and more structurally consistent predictions than competing methods, particularly around object boundaries and elongated regions. Compared with CNN- and Transformer-based models, the proposed method significantly suppresses isolated noisy pixels while preserving fine-grained spatial structures.

On the WHU-Hi-LongKou dataset with only 0.2% training samples, DESDA-Mamba achieves an OA of 99.13±0.14%, ranking second overall. It surpasses MambaHSI by 0.31% in OA and trails GSC-ViT by only 0.06%. DESDA-Mamba also achieves highly competitive AA and Kappa values while maintaining comparable computational efficiency. Specifically, the training and testing times are 0.48 s and 3.74 s, respectively, which are substantially lower than those of MambaHSI and remain within a reasonable range compared with GSC-ViT. As one of the only two methods achieving over 99% OA on this large-scale dataset, DESDA-Mamba still demonstrates strong competitiveness under extremely limited training conditions.

The relatively suboptimal performance under the 0.2% training ratio may be attributed to the extremely limited number of training samples. Although WHU-Hi-LongKou is significantly larger than the other datasets, the 0.2% training ratio provides only 408 training samples, fewer than those of University of Pavia (426), Salinas (523), and Indian Pines (1025), whereas the testing set contains 102,065 samples, approximately 5–20 times larger than those of the other datasets. Under such an extremely low-training and large-testing setting, the proposed strided scanning further reduces the number of scanned pixels by nearly 69%, which may limit the optimization of the feature extraction network and restrict the full potential of the directional modeling mechanism.

© 2026 by the author(s). Distributed under a Creative Commons CC BY license.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 28 May 2026 doi:10.20944/preprints202605.1990.v1

Table 5. Comparison results of different methods on the WHU-Hi-LongKou dataset (train ratio 0.2%).

Class	CNN-based Methods			Transformer-based Methods				SSM-based Methods	
	LSSAN	Lite-HCNet	MSDAN	SSFTT	SimPoolFormer	SpectralFormer	GSC-ViT	MambaHSI	DESDA-Mamba
1	97.44±1.62	99.90±0.05	99.67±0.16	99.81±0.18	99.87±0.06	99.28±0.28	99.91±0.08	99.91±0.06	**99.92±0.08**
2	85.90±7.81	98.11±0.56	98.74±0.80	98.64±0.82	89.65±1.60	90.57±3.00	99.18±0.48	**99.30±0.33**	99.22±0.42
3	81.62±11.38	96.65±1.40	**98.67±1.39**	95.11±1.32	96.03±1.00	95.46±2.37	98.63±1.05	98.25±0.92	97.53±1.53
4	97.54±0.46	99.71±0.11	99.24±0.27	99.29±0.22	99.42±0.13	98.75±0.17	**99.77±0.12**	99.70±0.20	99.74±0.10
5	60.83±11.73	87.78±4.54	79.20±7.86	84.09±6.69	79.01±2.17	60.96±8.37	**92.04±5.22**	82.05±3.95	91.66±5.00
6	90.78±4.59	99.29±0.16	98.64±0.83	99.60±0.18	**99.77±0.15**	97.82±0.93	99.42±0.23	99.21±0.22	99.06±0.36
7	98.55±0.94	99.90±0.05	99.87±0.08	99.80±0.10	**99.96±0.02**	99.91±0.12	99.92±0.05	**99.96±0.04**	**99.96±0.04**
8	58.25±13.84	87.01±3.66	91.20±2.90	**93.72±1.68**	86.97±1.52	88.36±4.34	92.62±1.98	90.27±1.86	90.88±2.59
9	57.75±16.96	91.03±1.81	93.78±2.01	92.43±1.85	88.68±1.21	81.92±2.20	92.68±2.16	90.03±1.88	**93.95±1.75**
OA	93.62±1.67	98.76±0.19	98.63±0.18	98.80±0.11	98.12±0.10	97.22±0.26	**99.19±0.13**	98.82±0.13	99.13±0.14
AA	80.96±5.89	95.49±0.79	95.44±0.98	95.83±0.67	93.26±0.32	90.34±1.25	**97.13±0.68**	95.41±0.46	96.88±0.58
Kappa	91.62±2.21	98.37±0.25	98.19±0.23	98.42±0.15	97.52±0.14	96.34±0.34	**98.94±0.17**	98.44±0.18	98.85±0.18
Params/M	0.04	**0.004**	3.42	0.11	1.49	0.05	0.13	0.12	0.04
FLOPs/M	2.28	**1.56**	1740	4.79	22.65	1.97	9.42	4.7	5.91
Train Time/s	0.22	3.19	3.77	**0.13**	0.33	0.26	0.35	0.60	0.48
Test Time/s	2.42	20.89	46.95	**1.70**	2.51	2.27	3.08	7.09	3.74

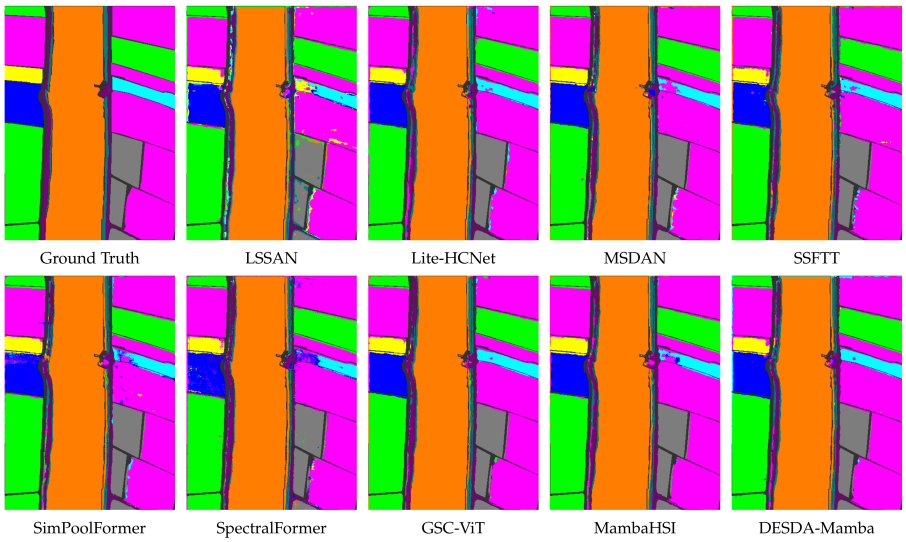


Figure 10. Ground Truth and classification maps of different methods on WHU-Hi-LongKou dataset.

Table 6. Comparison results of different methods on the WHU-Hi-LongKou dataset (Train Ratio 0.1%).

Class	CNN-based Methods			Transformer-based Methods				SSM-based Methods	
	LSSAN	Lite-HCNet	MSDAN	SSFTT	SimPoolFormer	SpectralFormer	GSC-ViT	MambaHSI	DESDA-Mamba
1	99.87±0.08	**99.97±0.02**	99.92±0.04	99.96±0.02	**99.97±0.02**	99.83±0.11	99.94±0.04	99.89±0.04	99.96±0.02
2	98.38±0.47	99.66±0.13	99.64±0.21	99.52±0.34	97.97±0.50	97.05±0.87	99.88±0.19	**99.97±0.03**	99.80±0.28
3	93.69±1.82	97.75±1.01	98.55±0.90	97.17±1.11	94.05±0.92	94.61±1.41	**98.66±0.94**	98.62±0.74	97.33±0.65
4	99.42±0.29	99.78±0.08	99.71±0.06	99.83±0.07	99.43±0.10	99.25±0.14	**99.92±0.06**	99.83±0.06	**99.92±0.06**
5	97.76±0.82	98.92±0.39	98.76±0.51	98.93±0.41	95.68±0.68	96.26±0.99	99.45±0.48	99.16±0.33	**99.68±0.26**
6	99.01±0.62	99.78±0.16	99.74±0.09	**99.90±0.12**	99.74±0.10	99.71±0.21	99.66±0.19	99.81±0.07	99.89±0.08
7	99.81±0.15	99.95±0.02	99.90±0.04	99.87±0.04	99.97±0.02	**99.98±0.02**	99.92±0.04	99.94±0.02	99.93±0.03
8	95.40±1.53	97.39±0.52	97.46±0.63	**97.53±0.46**	96.35±0.44	97.49±0.88	97.37±0.86	97.15±0.30	97.26±0.86
9	91.50±1.90	94.45±1.38	95.40±1.30	94.84±1.19	93.25±0.72	95.00±1.14	94.93±1.08	93.35±0.71	**96.37±1.16**
OA	99.10±0.13	99.60±0.06	99.58±0.03	99.59±0.02	99.24±0.03	99.23±0.07	99.66±0.03	99.59±0.04	**99.70±0.03**
AA	97.20±0.29	98.63±0.22	98.79±0.09	98.62±0.17	97.38±0.07	97.69±0.33	98.86±0.18	98.64±0.14	**98.90±0.13**
Kappa	98.81±0.17	99.47±0.08	99.45±0.04	99.46±0.03	98.99±0.04	98.98±0.10	99.56±0.05	99.46±0.05	**99.60±0.04**
Params/M	0.04	**0.004**	3.42	0.11	1.49	0.05	0.13	0.12	0.04
FLOPs/M	2.28	**1.56**	1740	4.79	22.65	1.97	9.42	4.7	5.91
Train Time/s	1.02	10.99	18.3	**0.63**	1.83	1.41	1.56	2.55	2.19
Test Time/s	2.15	16.54	45.01	**1.55**	2.29	2.06	2.6	6.32	3.68

We therefore infer that the limited training samples constrain the optimization of the proposed scanning strategy. With more sufficient supervision, the advantage of DESDA-Mamba becomes more evident, as verified by the subsequent 1% training-ratio experiment.

To further verify this inference, additional experiments were conducted on WHU-Hi-LongKou using a 1% training ratio for all competing methods, as reported in Table 6. Under this setting, DESDA-Mamba achieves the best OA of 99.70±0.03%, surpassing the second-best GSC-ViT by 0.04% and improving by 0.57% compared with its performance under the 0.2% training ratio. DESDA-Mamba also achieves the highest AA and Kappa values, further demonstrating its superior category-wise consistency and generalization capability when sufficient supervision is available.

© 2026 by the author(s). Distributed under a Creative Commons CC BY license.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 28 May 2026 doi:10.20944/preprints202605.1990.v1

In terms of efficiency, increasing the training ratio from 0.2% to 1% enlarges the training time of all methods by approximately five times, while the testing time changes only marginally. DESDA-Mamba requires 2.19 s training time and 3.68 s testing time, both remaining substantially lower than those of MambaHSI and within a reasonable overhead range compared with GSC-ViT. These results demonstrate that when sufficient training samples are available, DESDA-Mamba can fully exploit the advantages of direction-adaptive and diagonal-aware scanning, achieving state-of-the-art classification performance on large-scale datasets while maintaining competitive computational efficiency.

3.4. Ablation Experiment

The proposed DESDA-Mamba contains three key mechanisms: 1) direction adaptation, where each channel group dynamically selects the most suitable scanning direction according to its own features; 2) strided scanning, which adopts sparse sampling with a stride of 2; and 3) the new direction combination, consisting of horizontal (left-to-right), vertical (top-to-bottom), main diagonal, and anti-diagonal scanning directions. To evaluate the effectiveness of these mechanisms, we conducted ablation experiments by combining them into eight different configurations across four datasets.

Disabling direction adaptation indicates a fixed assignment strategy, where the four scanning directions are statically assigned to the four channel groups throughout the entire experiment. Disabling strided scanning corresponds to full-pixel scanning. Disabling the new direction combination indicates the use of the conventional direction combination, including horizontal (left-to-right), horizontal reverse (right-to-left), vertical (top-to-bottom), and vertical reverse (bottom-to-top) directions.

Table 7. Ablation Experiment results on four HSI datasets.

Dataset	New Direction Combination	Direction Adaptation	Strided Scanning	OA	AA	Kappa	Train Time/s	Test Time/s
University of Pavia				98.67±0.15	97.91±0.22	98.24±0.20	**0.34**	**0.60**
	✓			98.76±0.20	97.98±0.33	98.36±0.27	0.37	0.64
		✓		98.80±0.20	98.02±0.35	98.41±0.27	0.40	0.74
			✓	98.73±0.18	97.96±0.39	98.31±0.24	0.38	0.65
	✓	✓		98.81±0.28	98.17±0.35	98.42±0.37	0.41	0.76
	✓		✓	98.81±0.24	**98.19±0.29**	98.42±0.32	0.37	0.65
		✓	✓	98.74±0.19	98.00±0.30	98.33±0.25	0.44	0.82
	✓	✓	✓	**98.87±0.18**	98.06±0.24	**98.51±0.24**	0.47	0.85
Salinas				99.08±0.12	99.37±0.08	98.98±0.14	**0.49**	**0.73**
	✓			99.15±0.14	**99.39±0.13**	99.05±0.15	**0.49**	0.76
		✓		99.16±0.15	**99.39±0.11**	99.06±0.17	0.53	0.88
			✓	99.17±0.15	99.35±0.09	99.07±0.17	0.56	0.76
	✓	✓		99.21±0.17	99.38±0.14	99.12±0.13	0.54	0.82
	✓		✓	99.18±0.11	99.31±0.12	99.09±0.12	0.52	0.78
		✓	✓	99.18±0.12	99.36±0.16	99.09±0.13	0.63	1.01
	✓	✓	✓	**99.27±0.11**	**99.39±0.18**	**99.19±0.12**	0.53	0.87
Indian Pines				98.46±0.15	97.18±0.98	98.24±0.17	**0.86**	0.22
	✓			98.57±0.22	97.85±1.02	98.37±0.26	0.89	0.25
		✓		98.62±0.25	97.39±1.01	98.43±0.28	0.99	0.27
			✓	98.51±0.20	97.87±0.75	98.30±0.23	0.94	0.24
	✓	✓		98.52±0.25	97.58±0.62	98.31±0.29	1.03	**0.20**
	✓		✓	98.50±0.26	97.51±0.86	98.28±0.30	0.90	0.28
		✓	✓	98.51±0.23	97.60±1.01	98.30±0.26	1.06	0.28
	✓	✓	✓	**98.71±0.15**	**98.05±0.63**	**98.53±0.17**	1.08	0.27
WHU-Hi-LongKou				98.98±0.14	96.37±0.50	98.66±0.18	0.36	2.62
	✓			99.10±0.10	96.77±0.49	98.82±0.13	0.40	3.02
		✓		99.06±0.05	96.82±0.37	98.77±0.06	**0.34**	**2.44**
			✓	99.01±0.16	96.47±0.54	98.70±0.22	0.48	3.24
	✓	✓		99.05±0.14	96.54±0.62	98.75±0.18	0.44	3.53
	✓		✓	99.07±0.08	96.73±0.42	98.77±0.10	0.44	3.37
		✓	✓	99.02±0.09	96.58±0.40	98.71±0.12	0.51	4.04
	✓	✓	✓	**99.13±0.14**	**96.88±0.58**	**98.85±0.18**	0.48	3.74

Experimental settings and evaluation metrics follow those of the comparison experiments. On the WHU-Hi-LongKou dataset, the training ratio remains 0.2%, and all eight configurations report ten-run averages. The Params and FLOPs of all ablation models are nearly identical to DESDA-Mamba, with only occasional fluctuations (≈0.01), confirming the lightweight nature of the proposed mechanisms. Hence, these two metrics are omitted in the following analysis.

Table 7 reports the ablation results on four HSI datasets. Each proposed mechanism consistently improves the baseline configuration across all datasets, demonstrating the effectiveness of direction

© 2026 by the author(s). Distributed under a Creative Commons CC BY license.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 28 May 2026 doi:10.20944/preprints202605.1990.v1

adaptation, strided scanning, and the proposed diagonal-enabled new direction combination. Among them, direction adaptation provides the most consistent individual improvement, yielding OA gains of 0.13%, 0.08%, 0.16%, and 0.08% on University of Pavia, Salinas, Indian Pines, and WHU-Hi-LongKou, respectively. This indicates that dynamically adjusting scanning directions according to learned feature responses effectively enhances anisotropic spatial modeling. The proposed new direction combination further improves performance by introducing diagonal spatial dependency modeling beyond conventional horizontal and vertical scanning. Meanwhile, strided scanning achieves competitive accuracy gains while reducing redundant neighboring sampling and enlarging the effective receptive field.

More importantly, the combination of multiple mechanisms consistently produces larger improvements than individual components alone, indicating strong complementarity among the three designs. The complete DESDA-Mamba achieves the best OA, AA, and Kappa on nearly all datasets, surpassing the baseline by 0.20%, 0.19%, 0.25%, and 0.15% OA on University of Pavia, Salinas, Indian Pines, and WHU-Hi-LongKou, respectively, demonstrating that the three mechanisms jointly provide more comprehensive spatial structure modeling than any individual component alone. These results further suggest that the performance gains mainly originate from more effective directional modeling and sequence construction rather than increased model complexity, since Params and FLOPs remain nearly unchanged across all configurations.

In terms of efficiency, the baseline achieves the lowest computational cost because it avoids online direction prediction and strided serialization. DESDA-Mamba introduces moderate additional overhead due to adaptive direction estimation and sparse scanning operations. Nevertheless, the increase remains limited relative to the obtained performance improvements. For example, on University of Pavia, the training time increases from 0.34 s to 0.47 s, while OA improves from 98.67% to 98.87%. Similar trends can also be observed on the other three datasets. Overall, the proposed design achieves a favorable balance between classification performance, robustness, and computational efficiency.

3.5. Comparison with Model Variants

During the development of DESDA-Mamba, several model variants were explored. To further verify the effectiveness of the proposed framework, comparative experiments were conducted on six representative variants across four datasets under the same experimental settings as previous sections. Each experiment was repeated 10 times and the averaged results were reported. The training ratio on the WHU-Hi-LongKou dataset was still 0.2%. The variants mainly differ in four aspects: scanning direction design (V1, V2), direction selection strategy (V3, V4), direction prediction network (V5), and network depth (V6). The details are summarized as follows.

1. Scanning-direction variant V1: Row major scanning. All feature maps are serialized using only horizontal (left-to-right) scanning with stride 2, without direction adaptation or additional scanning directions. This Scanning-direction setting is the same as MambaHSI [35], while DESDA-Mamba and the ablation counterparts both employ four scanning directions.
2. Scanning-direction variant V2: Eight-direction scanning. The direction adaptation module selects scanning direction for each feature group from eight independent directions: horizontal and horizontal reverse ($\rightarrow, \leftarrow$), vertical and vertical reverse ($\downarrow, \uparrow$), main-diagonal and main-diagonal reverse ($\searrow, \nwarrow$), and anti-diagonal and anti-diagonal reverse ($\swarrow, \nearrow$).
3. Direction-selection variant V3: Fully random selection. The scanning direction of each feature group is randomly selected from the four directions in the new direction combination used in DESDA-Mamba without employing the direction adaptation module.
4. Direction-selection variant V4: Patch regrouping. Each patch first predicts its preferred direction using the direction adaptation network used in DESDA-Mamba. Patches with the same predicted direction are then regrouped into four sub-groups and separately processed by the corresponding directional Mamba modules.

© 2026 by the author(s). Distributed under a Creative Commons CC BY license.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 28 May 2026 doi:10.20944/preprints202605.1990.v1

Table 8. Comparison of different model variants on four HSI datasets.

Dataset	Metric	Scanning Direction		Selection Strategy		Extraction	Depth	Prototype
		V1 (Row-wise)	V2 (Eight-dir)	V3 (Random)	V4 (Regrouping)	V5 (Explicit)	V6 (Two-layer)	DESDA-Mamba
PaviaU	OA	98.79±0.18	98.74±0.23	98.78±0.12	98.77±0.12	98.79±0.15	98.69±0.30	**98.87±0.18**
	AA	98.07±0.33	98.05±0.33	98.04±0.23	97.96±0.27	**98.11±0.21**	97.89±0.41	98.06±0.24
	Kappa	98.40±0.24	98.33±0.31	98.38±0.16	98.37±0.16	98.39±0.20	98.26±0.40	**98.51±0.24**
	Params/M	0.03	0.03	0.03	0.03	0.03	0.04	0.03
	FLOPs/M	**5.15**	5.17	5.17	5.17	5.26	5.8	5.17
	Train Time/s	0.41	0.44	0.42	1.2	0.46	0.78	0.47
	Test Time/s	0.69	0.75	**0.66**	1.21	0.84	1.18	0.85
Salinas	OA	99.19±0.09	99.17±0.10	99.23±0.06	99.17±0.09	99.17±0.11	99.14±0.12	**99.27±0.11**
	AA	99.39±0.10	99.37±0.09	99.40±0.09	**99.42±0.10**	99.41±0.09	99.37±0.09	99.39±0.18
	Kappa	99.10±0.11	99.08±0.11	99.15±0.07	99.08±0.10	99.08±0.12	99.04±0.14	**99.19±0.12**
	Params/M	0.05	0.06	**0.05**	**0.05**	**0.05**	0.07	**0.05**
	FLOPs/M	**8.11**	8.21	8.20	8.21	8.39	10.29	8.21
	Train Time/s	0.55	0.57	0.55	1.59	0.62	1.02	**0.53**
	Test Time/s	0.88	1.06	0.89	1.65	1.04	1.42	**0.87**
IP	OA	98.55±0.23	98.67±0.17	98.56±0.15	98.63±0.25	98.49±0.21	98.51±0.21	**98.71±0.15**
	AA	97.72±0.79	97.73±0.75	97.52±0.92	**98.07±0.65**	97.28±1.58	97.71±0.86	98.05±0.63
	Kappa	98.35±0.26	98.48±0.19	98.35±0.18	98.44±0.28	98.28±0.24	98.30±0.24	**98.53±0.17**
	Params/M	0.05	0.06	**0.05**	**0.05**	**0.05**	0.07	**0.05**
	FLOPs/M	**8.11**	8.21	8.20	8.21	8.39	10.29	8.21
	Train Time/s	**0.91**	1.08	0.99	3.00	1.12	1.81	1.08
	Test Time/s	0.25	0.27	**0.24**	0.39	0.29	0.37	0.27
LongKou	OA	99.03±0.16	99.03±0.22	98.99±0.17	99.05±0.14	98.97±0.13	99.01±0.15	**99.13±0.14**
	AA	96.56±0.75	96.49±0.94	96.34±0.68	96.53±0.69	96.21±0.56	96.58±0.68	**96.88±0.58**
	Kappa	98.72±0.21	98.72±0.30	98.66±0.23	98.75±0.19	98.65±0.17	98.70±0.20	**98.85±0.18**
	Params/M	0.04	0.04	0.04	0.04	0.04	0.04	0.04
	FLOPs/M	**5.87**	5.91	5.90	5.91	6.02	6.82	5.91
	Train Time/s	0.52	0.49	**0.42**	1.25	0.48	0.86	0.48
	Test Time/s	**3.01**	3.81	3.07	6.02	3.86	6.29	3.74

5. Direction-prediction-network variant V5: Explicit directional feature extraction. Unlike the implicit direction adaptation in DESDA-Mamba, which directly pools feature maps to learn direction preferences from high-level semantic features, V5 explicitly extracts geometric directional responses using four predefined directional convolution kernels (horizontal, vertical, main-diagonal, anti-diagonal). Specifically, for each input channel, the four kernels are applied via depthwise convolution, producing four directional response sub-maps per channel. These sub-maps are then aggregated via global average pooling across both spatial dimensions and all channels, yielding a directional strength vector of shape $[4]$ for each feature map, forming an output of shape $[B, 4]$, which is subsequently fed into a lightweight MLP for direction prediction. The same batch-level averaging strategy as DESDA-Mamba is then applied to obtain a unified direction per batch.
6. Network-depth variant V6: Two-layer structure. The original DESDA-Mamba employs only a single core DESDA-Mamba block. In V6, an additional DESDA-Mamba block is stacked after the first, allowing the output features to undergo a second round of direction adaptation, strided scanning, and long-range dependency modeling.

The results in Table 8 reveal an important observation that directional modeling in HSI classification does not necessarily benefit from increasingly fine-grained or complex designs. Instead, an appropriate balance between directional diversity, statistical robustness, semantic adaptivity, and computational efficiency appears to be more effective. Across all four datasets, DESDA-Mamba consistently achieves the best OA, improving OA by approximately 0.08%–0.20% over most corresponding variants while maintaining competitive computational cost, demonstrating the robustness of the proposed directional adaptation strategy.

For the scanning-direction variants, V1 (row major scanning) consistently yields lower OA and Kappa than DESDA-Mamba on all datasets. Specifically, DESDA-Mamba improves OA over V1 by 0.08% on PaviaU, 0.08% on Salinas, 0.16% on Indian Pines, and 0.10% on LongKou. These results suggest that using only a single horizontal scanning direction limits the modeling capability for diverse spatial structures. Nevertheless, V1 achieves slightly lower FLOPs and generally faster inference speed, showing that simple row-wise scanning remains computationally efficient. V2 (eight-direction scanning) also underperforms DESDA-Mamba despite introducing more candidate directions. Although V2 slightly improves OA over V1 on Indian Pines (98.67% vs. 98.55%), its overall performance remains inferior to DESDA-Mamba. This suggests that simply increasing the number of scanning directions does not necessarily improve performance, since several reverse-order directions provide highly correlated directional information and enlarge the direction-selection space unnecessarily. The

© 2026 by the author(s). Distributed under a Creative Commons CC BY license.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 28 May 2026 doi:10.20944/preprints202605.1990.v1

resulting increase in optimization difficulty may weaken the stability of direction adaptation during training. Overall, these results validate the effectiveness of the proposed four-direction design, which preserves the four principal spatial orientations (horizontal, vertical, main-diagonal, and anti-diagonal) while avoiding unnecessary directional redundancy.

For the direction-selection strategy variants, V3 (random selection) achieves lower accuracy on all datasets, confirming the necessity of adaptive direction selection. Although random selection slightly reduces training and inference time, it sacrifices the spatial modeling capability brought by learned directional adaptation. V4 (patch regrouping) further demonstrates that finer-grained directional routing is not necessarily more effective. Despite explicitly assigning each patch to its predicted direction, V4 fails to improve OA or Kappa on most datasets, while its computational overhead increases substantially. For example, on PaviaU, the training and testing time of V4 increase from 0.47 s and 0.85 s to 1.20 s and 1.21 s, respectively, while OA decreases slightly from 98.87% to 98.77%. Similar trends can also be observed on Indian Pines, where V4 requires nearly three times the training time of DESDA-Mamba (3.00 s vs. 1.08 s) without achieving superior OA. These results suggest that patch-level routing amplifies prediction noise and introduces considerable regrouping overhead. In contrast, the proposed batch-level unified direction strategy suppresses unstable patch-wise predictions through batch averaging, leading to more robust direction selection and significantly better computational efficiency. Moreover, DESDA-Mamba generally exhibits smaller or comparable standard deviations than V4 across multiple datasets, further indicating improved optimization stability.

For the direction-prediction-network variant, V5 (explicit directional feature extraction) also fails to outperform DESDA-Mamba. Although V5 explicitly extracts directional responses using handcrafted directional convolution kernels, its OA remains consistently lower across all datasets. For instance, on Indian Pines and LongKou, V5 reduces OA from 98.71% to 98.49% and from 99.13% to 98.97%, respectively. This suggests that explicit low-level geometric direction estimation is less effective than the proposed implicit direction adaptation strategy. In DESDA-Mamba, directional preferences are inferred directly from high-level learned feature-channel response patterns, which are more semantically discriminative and robust than handcrafted directional responses. Moreover, not all feature channels exhibit clear directional sensitivity, making explicit per-pixel directional estimation more susceptible to noisy or ambiguous responses.

For the network-depth variant, V6 (two-layer structure) further shows that blindly increasing network depth does not necessarily improve classification performance. Although V6 introduces an additional DESDA-Mamba block for deeper feature extraction, its OA, AA, and Kappa remain consistently lower than those of DESDA-Mamba on most datasets, while the training and inference cost increases substantially. For example, on Salinas, V6 increases FLOPs from 8.21 M to 10.29 M and nearly doubles testing time (1.42 s vs. 0.87 s), yet OA still decreases from 99.27% to 99.14%. Similar phenomena can also be observed on PaviaU and Indian Pines. These results suggest that the proposed single-block architecture is already sufficient for effective spatial dependency modeling, whereas excessive stacking may introduce feature redundancy and increase the risk of overfitting.

Overall, the variant experiments reveal an important insight: directional modeling in HSI classification is not necessarily more effective when performed at excessively fine granularity or with increasingly complex directional decomposition. Increasing directional complexity, routing granularity, explicit geometric constraints, or network depth does not consistently improve performance, and may instead introduce redundant directional dependencies, unstable optimization, and amplified prediction noise. Although several variants slightly reduce FLOPs or inference time, their classification performance consistently degrades. Conversely, more complex variants substantially increase computational overhead without yielding corresponding accuracy gains. In contrast, DESDA-Mamba achieves a better balance between directional diversity, statistical robustness, semantic adaptivity, and computational efficiency through implicit batch-level direction adaptation and a compact non-redundant four-direction scanning design. Interestingly, these results suggest that directional modeling

© 2026 by the author(s). Distributed under a Creative Commons CC BY license.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 28 May 2026 doi:10.20944/preprints202605.1990.v1

in HSI classification may benefit more from statistically stable and semantically adaptive direction selection than from increasingly fine-grained directional decomposition.

4. Conclusions and Future Research

This paper proposed DESDA-Mamba, a direction-adaptive Mamba framework with diagonal-enabled strided scanning for hyperspectral image classification. By introducing implicit direction adaptation, strided sparse scanning, and efficient diagonal scanning, the proposed method improves anisotropic spatial modeling while reducing redundant computation. Extensive experiments on four public HSI datasets demonstrated that DESDA-Mamba achieves superior or highly competitive classification performance with favorable computational efficiency. Furthermore, the ablation and model-variant experiments revealed an important finding that increasingly complex directional modeling does not necessarily lead to better HSI classification performance. Instead, effective HSI directional modeling requires an appropriate balance between directional diversity, robustness, and computational simplicity. The proposed diagonal-enabled four-direction design with batch-level implicit direction adaptation demonstrates that moderate and well-constrained directional flexibility can be more effective than both overly simplified fixed-direction modeling and excessively fine-grained directional routing strategies.

In future work, more flexible adaptive scanning strategies, such as differentiable direction selection or dynamic stride adjustment, may be explored to further improve the adaptability of the framework under more complex large-scale remote sensing scenarios.

Author Contributions: Conceptualization, C. Liu; methodology, C. Liu; software, C. Liu and H. Jin; visualization, C. Liu; formal analysis, C. Liu and H. Jin; validation, C. Liu and H. Jin; writing—original draft preparation, C. Liu; writing—review and editing, C. Liu and H. Jin; resources, H. Jin; funding acquisition, H. Jin. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported in part by the National Natural Science Foundation of China under Grant 62173171.

Data Availability Statement: Publicly available datasets were analyzed in this study.

Acknowledgments: We would like to thank anonymous editor and reviewers.

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript:

AA	Average Accuracy
CNN	Convolutional Neural Network
DWConv	Depthwise Convolution
FFN	Feed-Forward Network
GAP	Global Average Pooling
HSI	Hyperspectral Image
OA	Overall Accuracy
PCA	Principal Component Analysis
SPD	Symmetric Positive Definite
SSM	State Space Model
SVM	Support Vector Machine
ViT	Vision Transformer
1D-SS	One-Dimensional Selective Scan

© 2026 by the author(s). Distributed under a Creative Commons CC BY license.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 28 May 2026 doi:10.20944/preprints202605.1990.v1

References

1. Du B, Zhang Y, Zhang L, et al. Beyond the sparsity-based target detector: A hybrid sparsity and statistics-based detector for hyperspectral images. *IEEE Trans. Image Process.* **2016**, *25*, 5345–5357.
2. Pande S, Banerjee B. HyperLoopNet: Hyperspectral image classification using multiscale self-looping convolutional networks. *ISPRS J. Photogramm. Remote Sens.* **2022**, *183*, 422–438.
3. Xu Y, Du B, Zhang L. Beyond the patchwise classification: Spectral-spatial fully convolutional networks for hyperspectral image classification. *IEEE Trans. Big Data* **2020**, *6*, 492–506.
4. Wang Z, Du B, Zhang L, et al. A novel semisupervised active-learning algorithm for hyperspectral image classification. *IEEE Trans. Geosci. Remote Sens.* **2017**, *55*, 3071–3083.
5. Paoletti M E, Haut J M, Plaza J, et al. Deep learning classifiers for hyperspectral imaging: A review. *ISPRS J. Photogramm. Remote Sens.* **2019**, *158*, 279–317.
6. Zhong Y, Hu X, Luo C, et al. WHU-Hi: UAV-borne hyperspectral with high spatial resolution (H2) benchmark datasets and classifier for precise crop identification based on deep convolutional neural network with CRF. *Remote Sens. Environ.* **2020**, *250*, Art. no. 112012.
7. Lee M A, Huang Y, Yao H, et al. Determining the effects of storage on cotton and soybean leaf samples for hyperspectral analysis. *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.* **2014**, *7*, 2562–2570.
8. Yuan J, Wang S, Wu C, et al. Fine-grained classification of urban functional zones and landscape pattern analysis using hyperspectral satellite imagery: A case study of Wuhan. *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.* **2022**, *15*, 3972–3991.
9. Weber C, et al. Hyperspectral imagery for environmental urban planning. In Proceedings of the IEEE International Geoscience and Remote Sensing Symposium (IGARSS), Valencia, Spain, 22–27 July 2018; pp. 1628–1631.
10. Kramer S J, Siegel D A, Maritorena S, et al. Modeling surface ocean phytoplankton pigments from hyperspectral remote sensing reflectance on global scales. *Remote Sens. Environ.* **2022**, *270*, Art. no. 112879.
11. Brook A, Dor E B. Quantitative detection of settled dust over green canopy using sparse unmixing of airborne hyperspectral data. *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.* **2016**, *9*, 884–897.
12. Melgani F, Bruzzone L. Classification of hyperspectral remote sensing images with support vector machines. *IEEE Trans. Geosci. Remote Sens.* **2004**, *42*, 1778–1790.
13. Ahmad M, et al. Hyperspectral image classification—Traditional to deep models: A survey for future prospects. *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.* **2022**, *15*, 968–999.
14. Ham J, Chen Y, Crawford M M, et al. Investigation of the random forest framework for classification of hyperspectral data. *IEEE Trans. Geosci. Remote Sens.* **2005**, *43*, 492–501.
15. Gualtieri J A, Chettri S. Support vector machines for classification of hyperspectral data. In Proceedings of the IEEE International Geoscience and Remote Sensing Symposium (IGARSS), Honolulu, HI, USA, 24–28 July 2000; Volume 2, pp. 813–815.
16. Feng J, Jiao L, Liu F, et al. Unsupervised feature selection based on maximum information and minimum redundancy for hyperspectral images. *Pattern Recognit.* **2016**, *51*, 295–309.
17. Wang Z-Y, Xia Q-M, Yan J-W, et al. Hyperspectral image classification based on spectral and spatial information using multi-scale ResNet. *Appl. Sci.* **2019**, *9*, 4890.
18. Gao L, Xiao S, Hu C, et al. Hyperspectral image classification based on fusion of convolutional neural network and graph network. *Appl. Sci.* **2023**, *13*, 7143.
19. Lee H, Kwon H. Going deeper with contextual CNN for hyperspectral image classification. *IEEE Trans. Image Process.* **2017**, *26*, 4843–4855.
20. Hamida A B, Benoit A, Lambert P, et al. 3-D deep learning approach for remote sensing image classification. *IEEE Trans. Geosci. Remote Sens.* **2018**, *56*, 4420–4434.
21. Ahmad M, Khan A M, Mazzara M, et al. A fast and compact 3-D CNN for hyperspectral image classification. *IEEE Geosci. Remote Sens. Lett.* **2022**, *19*, 1–5.
22. Roy S K, Krishna G, Dubey S R, et al. HybridSN: Exploring 3-D-2-D CNN feature hierarchy for hyperspectral image classification. *IEEE Geosci. Remote Sens. Lett.* **2020**, *17*, 277–281.
23. Dai H, Yue Y, Liu Q. Hyperspectral image classification based on hybrid depth-wise separable convolution and dual-branch feature fusion network. *Appl. Sci.* **2023**, *13*, 9667.
24. Yang J, Du B, Zhang L. From center to surrounding: An interactive learning framework for hyperspectral image classification. *ISPRS J. Photogramm. Remote Sens.* **2023**, *197*, 145–166.
25. Vaswani A, et al. Attention is all you need. In Proceedings of the Advances in Neural Information Processing Systems (NeurIPS), Long Beach, CA, USA, 4–9 December 2017; Volume 30.

© 2026 by the author(s). Distributed under a Creative Commons CC BY license.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 28 May 2026 doi:10.20944/preprints202605.1990.v1

26. Li Z, Xue Z, Xu Q, et al. SPFormer: Self-pooling transformer for few-shot hyperspectral image classification. *IEEE Trans. Geosci. Remote Sens.* **2024**, *62*, Art. no. 5502019.
27. Hong D, et al. SpectralFormer: Rethinking hyperspectral image classification with transformers. *IEEE Trans. Geosci. Remote Sens.* **2021**, *60*, 1–15.
28. Peng Y, Zhang Y, Tu B, et al. Spatial-spectral transformer with cross-attention for hyperspectral image classification. *IEEE Trans. Geosci. Remote Sens.* **2022**, *60*, Art. no. 5537415.
29. Mei S, Song C, Ma M, et al. Hyperspectral image classification using group-aware hierarchical transformer. *IEEE Trans. Geosci. Remote Sens.* **2022**, *60*, Art. no. 5539014.
30. Qi W, Huang C, Wang Y, et al. Global-local 3-D convolutional transformer network for hyperspectral image classification. *IEEE Trans. Geosci. Remote Sens.* **2023**, *61*, Art. no. 5510820.
31. Zhao Z, Xu X, Li S, et al. Hyperspectral image classification using groupwise separable convolutional vision transformer network. *IEEE Trans. Geosci. Remote Sens.* **2024**, *62*, Art. no. 5511817.
32. Gu A, Goel K, Re C. Efficiently modeling long sequences with structured state spaces. In Proceedings of the International Conference on Learning Representations (ICLR), Virtual Event, 25–29 April 2022.
33. Gu A, Dao T. Mamba: Linear-time sequence modeling with selective state spaces. In Proceedings of the International Conference on Learning Representations (ICLR), Vienna, Austria, 7–11 May 2024.
34. Liu Z, et al. VMamba: Visual state space model. *IEEE Trans. Pattern Anal. Mach. Intell.* **2025**, (early access).
35. Li Y, Luo Y, Zhang L, et al. MambaHSI: Spatial-spectral Mamba for hyperspectral image classification. *IEEE Trans. Geosci. Remote Sens.* **2024**, *62*, Art. no. 5524216.
36. He Y, Tu B, Liu B, et al. 3DSS-Mamba: 3D-spectral-spatial Mamba for hyperspectral image classification. *IEEE Trans. Geosci. Remote Sens.* **2024**, *62*, Art. no. 5535514.
37. Sheng J, Zhou J, Wang J, et al. DualMamba: A lightweight spectral-spatial Mamba-convolution network for hyperspectral image classification. *IEEE Trans. Geosci. Remote Sens.* **2025**, *63*, Art. no. 5502914.
38. Chen K, Chen B, Liu C, et al. RSMamba: Remote sensing image classification with state space model. *IEEE Geosci. Remote Sens. Lett.* **2024**, *21*, 1–5.
39. Cao H, et al. DBMCMamba: Few-Shot hyperspectral image classification with mamba and manifold convolution fusion network. *Knowl.-Based Syst.* **2025**, *330*, Art. no. 114531.
40. Sun L, Zhao G, Zheng Y, et al. Spectral-spatial feature tokenization transformer for hyperspectral image classification. *IEEE Trans. Geosci. Remote Sens.* **2022**, *60*, Art. no. 5522214.
41. Wang X, Fan Y. Multiscale densely connected attention network for hyperspectral image classification. *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.* **2022**, *15*, 1–12.
42. Cui Y, Xia J, Wang Z, et al. Lightweight spectral-spatial attention network for hyperspectral image classification. *IEEE Trans. Geosci. Remote Sens.* **2022**, *60*, Art. no. 5524814.
43. Ma X, Kang X, Qin H, et al. A lightweight hybrid convolutional neural network for hyperspectral image classification. *IEEE Trans. Geosci. Remote Sens.* **2023**, *61*, Art. no. 5512614.
44. Roy S K, Jamali A, Chanussot J, et al. SimPoolFormer: A two-stream vision transformer for hyperspectral image classification. *Remote Sens. Appl. Soc. Environ.* **2025**, *37*, Art. no. 101450.

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.

© 2026 by the author(s). Distributed under a Creative Commons CC BY license.