remote sensing

MDPI

Article

Lite-BSSNet: A Lightweight Blueprint-Guided Visual State Space Network for Remote Sensing Imagery Segmentation

Jiaxin Yan, Yuxiang Xie *, Yan Chen, Yanming Guo and Wenzhe Liu

College of Systems Engineering, National University of Defense Technology, Changsha 410000, China; yanjiaxin2024@nudt.edu.cn (J.Y.); chenyan0702@nudt.edu.cn (Y.C.); guoyanming@nudt.edu.cn (Y.G.); liuwenzhe15@nudt.edu.cn (W.L.)

* Correspondence: yxxie@nudt.edu.cn

Highlights

What are the main findings?

- We propose Lite-BSSNet, a lightweight blueprint-guided visual state space network that reframes the decoding process into three stages: structural blueprint generation, cross-layer bridging, and structural repair. This approach effectively addresses the coexistence of multi-scale objects and the tendency of slender structures to break in remote sensing semantic segmentation.
- Extensive experiments demonstrate that this method achieves state-of-the-art performance, while ablation studies and qualitative visualizations validate the effectiveness and efficiency of this decoding paradigm.

What is the implication of the main finding?

- The results indicate that Lite-BSSNet has the potential to be applied to high-resolution remote sensing segmentation tasks in resource-constrained settings, as it improves accuracy without increasing the number of parameters while maintaining low computational cost.
- The proposed blueprint–bridge–repair design provides a generalizable decoder paradigm: structural priors, cross-layer alignment, and structure-repair refinement, which may be extended to broader Mamba-based architectures and other remote sensing conditions.

Abstract

Remote sensing image segmentation requires balancing global context and local detail across multi-scale objects. However, convolutional neural network (CNN)-based methods struggle to model long-range dependencies, while transformer-based approaches suffer from quadratic complexity and become inefficient for high-resolution remote sensing scenarios. In addition, the semantic gap between deep and shallow features can cause misalignment during cross-layer aggregation, and information loss in upsampling tends to break thin continuous structures, such as roads and roof edges, introducing pronounced structural noise. To address these issues, we propose lightweight Lite-BSSNet (Blueprint-Guided State Space Network). First, a Structural Blueprint Generator (SBG) converts high-level semantics into an edge-enhanced structural blueprint that provides a topological prior. Then, a Visual State Space Bridge (VSS-Bridge) aligns multi-level features and projects axially aggregated features into a linear-complexity visual state space, smoothing high-gradient edge signals for sequential scanning. Finally, a Structural Repair Block (SRB) enlarges the effective receptive field via dilated convolutions and uses spatial/channel gating to suppress upsampling artifacts and reconnect thin structures. Experiments on the ISPRS Vaihingen and Potsdam datasets show that Lite-BSSNet achieves the highest segmentation accuracy among the compared lightweight models, with mIoU of 83.9% and

Check for updates

Academic Editor: Xiangtao Zheng

Received: 15 December 2025
Revised: 22 January 2026
Accepted: 27 January 2026
Published: 30 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.

86.7%, respectively, while requiring only 45.4 GFLOPs, thus achieving a favorable trade-off between accuracy and efficiency.

Keywords: remote sensing; semantic segmentation; state space model; lightweight

1. Introduction

Semantic segmentation is a fundamental task in remote sensing and has emerged as a key tool for extracting urban spatial structures [1]. With the continuous advancement of imaging technology, the spatial resolution of remote sensing images has steadily improved. Leveraging their high-resolution surface observation capabilities, these images have been widely applied in various domains, including urban planning [2], environmental monitoring [3], precision agriculture [4], and national defense and security [5]. High-resolution scenes typically contain objects with large-scale variations, complex boundaries, and frequent occlusions, which place higher demands on multi-scale representation, long-range dependency modeling, and the delineation of thin structures. Accurate remote sensing image segmentation essentially requires a delicate balance between global context and local detail across multi-scale objects.

In recent years, the rapid development of deep learning has led to the replacement of traditional semantic segmentation methods by deep-neural-network-based approaches due to their superior segmentation accuracy. FCN [6] was the first model to transform a classification network into a fully convolutional network, paving the way for large-scale applications of deep learning in pixel-level visual tasks. However, it still exhibits notable limitations in fine-grained segmentation and boundary delineation. To overcome these issues, U-Net [7] introduced a symmetric encoder–decoder architecture in which the encoder extracts image features via convolutional layers (downsampling) and the decoder progressively restores spatial dimensions through transposed convolutions (upsampling). Consequently, extensive research has been conducted based on the U-Net framework in the field of remote sensing [8–10]. Meanwhile, remote sensing images pose unique challenges such as significant variations in object scales, complex terrain boundaries, and frequent occlusions. Such dense spatial dependencies [11] present substantial challenges for networks based on convolutional neural networks (CNNs) and vision transformers (ViTs). Although prior studies have sought to mitigate these issues by expanding the receptive field through large-kernel convolutions [12–14] and by employing ViTs [15,16] to enhance global context modeling, CNN-based models still lag behind ViTs in performance. However, ViTs incur high computational cost due to quadratic complexity when processing high-resolution remote sensing imagery.

Recently, Mamba-based derivative models [17,18] have shown great promise in modeling long-range dependencies with near-linear complexity in visual tasks. Building upon this, researchers have successfully incorporated Mamba as a decoder within U-Net architectures [19–22], achieving remarkable results. However, most of them directly feed multi-scale cross-layer features into the SSM module without carefully reorganizing heterogeneous and noisy information. In remote sensing, shallow features preserve fine textures and edges but are often noisy. Deep features are semantically rich but lack spatial details. This mismatch creates a pronounced semantic gap. Simple concatenation or dense connections across layers easily introduce redundancy and amplify noise propagation, which limits the full potential of state space models.

To alleviate cross-layer semantic misalignment, several studies have proposed dedicated cross-layer interaction modules. For example, CVMHUNet [23] introduces multi-

frequency information via MFMSBlock and a pointwise convolution branch to refine local details, while Sparx [24] enhances feature reuse through sparse cross-layer connections. Although these methods improve cross-layer information utilization, they either incur considerable extra computation or ignore the impact of cross-layer design on the complexity of subsequent decoders and thus do not address the problem from a "complexity bridging" perspective. At the same time, interpolation-based upsampling in U-Net-like decoders often leads to broken thin structures such as roads and roof edges, as illustrated in Figure 1. Existing models rarely address this issue from a repair perspective. Instead, they typically add an extra refinement head at the end of the decoder. For example, UNetFormer [9] employs the FRH module and improves performance. However, the refinement branch is loosely coupled to the noise introduced during upsampling, which limits its repair capability.

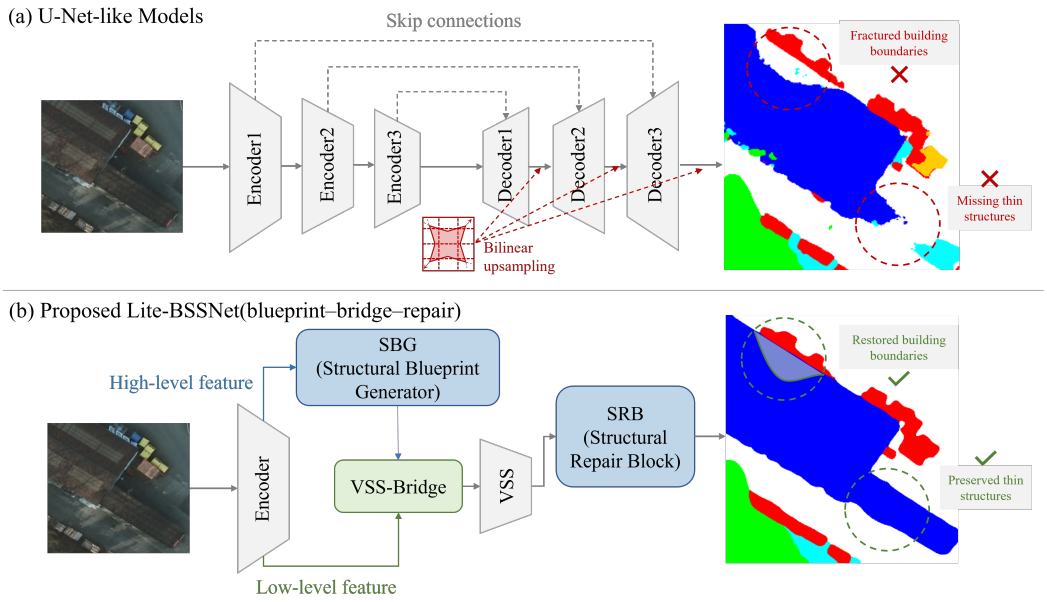


Figure 1. Motivation of the proposed Lite-BSSNet. (**a**) U-Net-like models with interpolation-based upsampling often produce fractured building boundaries and missing thin structures in high-resolution remote sensing scenes. (**b**) The proposed blueprint–bridge–repair decoding pipeline, consisting of the Structural Blueprint Generator (SBG), Visual State Space Bridge (VSS-Bridge), and Structural Repair Block (SRB), explicitly leverages structural priors and large-kernel refinement to restore building boundaries and preserve thin structures.

Structural priors and edge guidance are widely regarded as key cues for improving boundary quality and maintaining the continuity of fine structures in semantic segmentation. Classical methods incorporate contour constraints via active contours and energy minimization [25], or enforce spatial consistency using graphical models such as graph cuts [26]. In the deep learning era, edge-supervised mechanisms further strengthen boundary representations [27], and networks that explicitly model shape priors have demonstrated the importance of structural cues for accurate boundary localization and the preservation of thin, elongated objects [28]. However, remote sensing imagery is characterized by large-scale variations, cluttered textures, and frequent fragmentation of slender targets, making low-level edge responses prone to noise and instability. Motivated by these observations, we propose the SBG module to learn an edge-enhanced "structural blueprint" as a lightweight, learnable structural prior, providing stable structural constraints to guide the subsequent decoding process.

In this study, we revisit remote sensing semantic segmentation from a decoding perspective and propose Lite-BSSNet (Blueprint-guided State Space Network), a lightweight

model. We adopt a pre-trained RegNet-Y [29] as the encoder to extract multi-scale semantic features with low computational cost and redesign the decoder as a three-stage pipeline: structural blueprint generation, state space bridging, and structural repair. Specifically, the Structural Blueprint Generator (SBG) transforms high-level semantic features into an edge-enhanced structural blueprint that acts as a topological prior to provide stable global structural constraints and guide cross-layer interactions. Building on this prior, the Visual State Space Bridge (VSS-Bridge) aligns and aggregates multi-level features to reduce semantic mismatch and redundancy and then uses axial interactions to project the fused features into a linear-complexity visual state space; this projection smooths strong edge gradients into scan-friendly representations, effectively connecting transformer-like spatial interactions to streaming state space modeling. Finally, the Structural Repair Block (SRB) combines small kernels with dilated convolutions to approximate a large receptive field and incorporates spatial and channel-wise gating, which suppresses upsampling-induced noise and repairs breaks in thin structures such as roads and roof boundaries, thereby improving topological consistency with negligible additional parameters and computation. The main contributions of this paper are summarized as follows:

(1) We propose Lite-BSSNet, a blueprint-guided state space network that reformulates the decoding process of remote sensing semantic segmentation into three stages—structural blueprint generation, state space bridging, and structural repair—enabling efficient and robust segmentation in multi-scale, high-resolution scenarios.
(2) We design a Structural Blueprint Generator (SBG) and a Visual State Space Bridge (VSS-Bridge). SBG produces an edge-enhanced structural blueprint as a topological prior, while VSS-Bridge aligns and aggregates multi-level features and projects axially interacted features into a linear-complexity visual state space, jointly bridging cross-layer semantics and computational complexity.
(3) We propose a Structural Repair Block (SRB) that emulates a large receptive field through the combination of small and dilated convolutions and adopts spatial and channel-wise gating to suppress upsampling-induced noise and repair broken thin structures.

2. Related Work

2.1. CNN- and Transformer-Based Remote Sensing Image Segmentation Methods

With the advancement of deep learning technology, semantic segmentation techniques have found extensive applications in the field of remote sensing. Early research focused on CNNs, achieving efficient segmentation through local feature extraction. Among these, U-Net [7] significantly improved the accuracy of small-object segmentation by combining shallow high-resolution features with deep semantic representations, demonstrating outstanding performance in UAV-based building extraction [30,31] and high-resolution road segmentation [32]. This encoder–decoder framework subsequently became the standard architecture for remote sensing image segmentation networks [33–35]. Subsequent studies have refined this architecture through various improvements. For instance, Li [36] and Fu [37] introduced self-attention mechanisms to enhance the capability of capturing global contextual information. While CNN-based methods excel in extracting local details, their fixed receptive field size limits the model's ability to represent global context effectively. The vision transformer (ViT) [15] directly models pixel-wise correlations across entire images using self-attention, establishing a new paradigm for large-scale segmentation in remote sensing. TransUNet [38] integrates Transformer modules into the U-Net framework, combining the global modeling power of ViTs with U-Net's ability to preserve local details, thereby significantly improving segmentation performance. FTransUNet [39] further introduces an AdaMBA module to extract deep features and

employs an inter-correlation mechanism to guide multimodal feature fusion, effectively capturing the complex spatial structures of remote sensing data. However, ViT's global self-attention incurs high computational complexity, which limits its scalability to high-resolution and large-scale imagery. To address this, the Swin Transformer [40] proposes a "window self-attention + sliding window" mechanism that computes attention within local windows to reduce complexity while allowing cross-window information exchange through shifting operations. To satisfy the fine-grained requirements of remote sensing segmentation, ST-Unet [41] constructs a dual-encoder architecture that combines the Swin Transformer and CNN in parallel, integrating global dependencies into CNN features. Although transformer-based models provide superior global modeling capability, they remain computationally expensive, requiring significant hardware resources and posing challenges for deployment on low-power devices [42].

In this study, we designed a novel module by integrating the respective strengths of CNNs and Transformers, aiming to balance model efficiency and performance.

2.2. Mamba-Based Remote Sensing Image Segmentation Methods

To address the high computational cost of Transformers and the locality limitations of CNNs, the Mamba vision model based on state space models (SSMs) has been introduced into the field of remote sensing semantic segmentation [19,20,43]. Samba [44] is a semantic segmentation framework designed for high-resolution remote sensing imagery, which demonstrates the effectiveness of SSMs in this task. RS-Mamba [45] captures large-scale spatial features in remote sensing images through multidirectional global modeling using an omnidirectional selective scanning mechanism. Similarly, RS3Mamba [21] integrates Mamba into the U-Net architecture by constructing auxiliary branches based on the VSS module, providing additional global contextual information with near-linear computational complexity. UNetMamba [22] further proposes a pluggable Mamba-based segmentation decoder (MSD) for efficient semantic information decoding.

In this study, Lite-BSSNet explicitly restructures cross-layer information before state space decoding by generating an edge-enhanced structural blueprint and using a visual state space Bridge to filter, align, and project cross-layer features into the VSS domain.

2.3. Hybrid-Based Remote Sensing Image Segmentation Methods

Hybrid decoding architectures have recently become a research hotspot in remote sensing semantic segmentation. Their core concept is to utilize CNNs for efficient local feature extraction and Transformers or Mamba models for global feature modeling. Building upon this framework, different approaches emphasize different aspects. For example, STransFuse [8] employs dual branches to extract local and global features and fuses them adaptively through an attention mechanism. UNetFormer [9] introduces a global–local transformer block (GLTB) in the decoder stage, utilizing Transformer-based global modeling and CNN-based local feature extraction to enhance semantic associations and improve segmentation accuracy. UrbanSSF-S [20] integrates a feature state interaction mamba (FSI-Mamba) to model feature sequence relationships, combined with a global semantic enhancement (GSE) module and spatial interaction attention (SIA) mechanism to optimize feature representation. UMFormer [46] further reduces computational overhead through a lightweight attention mechanism and a Global–Local Visual State Space (GL-VSS) module.

In this study, Lite-BSSNet employs a blueprint-guided hybrid decoder: SBG generates edge-aware structural blueprints to suppress texture noise. Based on this blueprint, VSS-Bridge aligns cross-layer features and projects them into a state space representation for efficient long-range context aggregation; subsequently, SRB refines upsampled features via large receptive field convolutions and restores fine-grained structural connectivity.

3. Method

3.1. Overall

The overall architecture of the proposed Lite-BSSNet is illustrated in Figure 2. It follows an encoder–decoder framework. As RegNet has demonstrated a favorable balance between effectiveness and efficiency across a broad range of real-time semantic segmentation tasks, this study adopts a pre-trained RegNet-Y [29] as the encoder backbone. The encoder progressively extracts hierarchical feature maps, denoted as $\{E_1, E_2, E_3, E_4\}$. On top of the deepest feature E_4, a Structural Blueprint Generator (SBG) is applied to transform high-level semantics into an edge-enhanced structural blueprint, which serves as a topological prior to guide the subsequent decoding process.

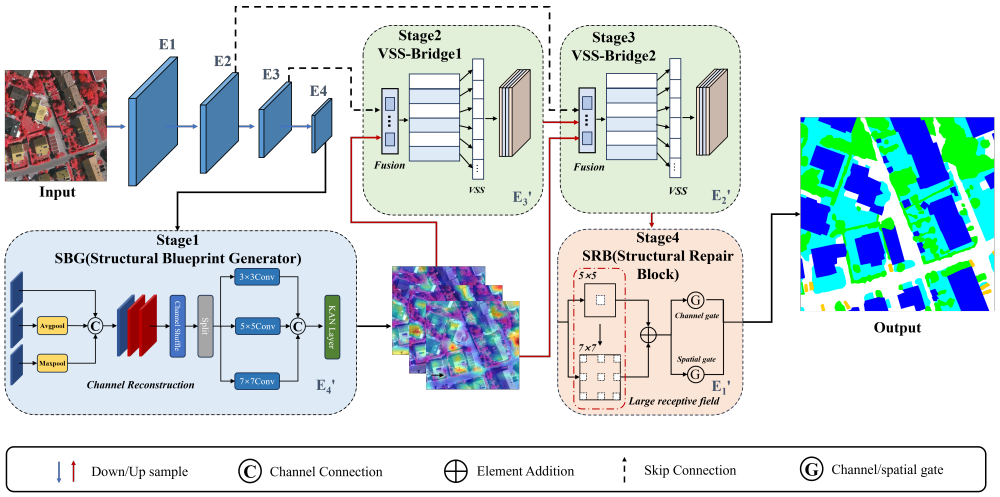


Figure 2. Architecture of our proposed Lite-BSSNet

The decoder is built around a Visual State Space Bridge (VSS-Bridge) and a Structural Repair Block (SRB). The VSS-Bridge receives the structural blueprint together with selected cross-layer features and first encodes positional information via a dynamic position encoding (DPE) module [47]. DPE adopts a 3×3 depthwise separable convolution (DW-Conv) [48], which is widely used in modern visual backbones, to dynamically inject spatial position cues into the feature maps. The position-aware features are then filtered and aligned across layers and fed into the Visual State Space (VSS) module from VMamba [18], which models long-range dependencies with near-linear complexity.

At the final decoding stage, the proposed SRB is employed to refine the high-resolution prediction. SRB combines several small convolution kernels with dilated convolutions to emulate the receptive field of a large convolutional kernel and further incorporates spatial and channel-wise gating. In this way, it suppresses noise accumulated during upsampling and explicitly repairs breaks in thin structures, enhancing topological consistency without incurring excessive computational cost.

Based on these components, Lite-BSSNet performs blueprint-guided cross-layer fusion in the early decoding stages through the VSS-Bridge, while the final stage focuses on structural refinement via SRB. The model is trained using the main loss function $\mathcal{L}_m$ and an auxiliary loss function $\mathcal{L}_a$.

3.2. Structural Blueprint Generator (SBG)

To address the issues of low parameter efficiency and limited interpretability in conventional MLPs, Liu et al. [49] proposed Kolmogorov–Arnold Networks (KANs), which replace fixed linear weights with learnable activation functions on edges. This design

provides strong interpretability and parameter efficiency and has recently demonstrated promising performance in semantic segmentation [50–52].

Compared with a conventional MLP, generating a boundary-oriented blueprint can be viewed as selective purification: it suppresses texture-like noise while enhancing class-transition cues and thin-structure continuity. KAN is well suited for this goal because it models cross-channel interactions with learnable univariate functions, enabling input-adaptive reweighting. This reweighting strengthens structure-consistent responses and suppresses spurious activations, producing a more stable structural prior. In contrast, convolutional alternatives mainly behave as local spatial filters and may blur or break thin structures.

Meanwhile, features extracted from different encoder stages exhibit distinct characteristics: lower-level features capture fine-grained textures and boundary cues but are more susceptible to noise, whereas higher-level features are more semantic and category-specific, containing richer global structure. Motivated by this observation, we design a Structural Blueprint Generator (SBG) to enhance the deepest encoder feature and produce an edge-enhanced structural prior that facilitates subsequent structure-aware decoding. The overall process is shown in Algorithm 1.

Algorithm 1: Structural Blueprint Generator (SBG)

Input: Deep encoder feature $E_4 \in \mathbb{R}^{B\times C_4\times H\times W}$

Output: Structural blueprint feature $O \in \mathbb{R}^{B\times C_d\times H\times W}$ and shortcut S

`/* Compute channel-statistics priors for structural enhancement */`

$P_{\text{avg}} \leftarrow \text{Mean}(E_4,\ \text{dim} = C)$ // $\in \mathbb{R}^{B\times 1\times H\times W}$

$P_{\max} \leftarrow \text{Max}(E_4,\ \text{dim} = C)$ // $\in \mathbb{R}^{B\times 1\times H\times W}$

`/* Channel reduction and feature reorganization */`

$X_r \leftarrow \text{Conv}_{1\times 1}(E_4)$ // $\in \mathbb{R}^{B\times C_d\times H\times W}$

$X_f \leftarrow \text{Concat}(X_r,\ P_{\text{avg}},\ P_{\max})$ `// concat along channel`

$X_f \leftarrow \text{ChannelShuffle}(X_f,\ g = 3)$ `// shuffle for efficient mixing`

`/* Multi-scale depthwise convolution fusion */`

Split X_f into three channel groups: $[X^{(1)}, X^{(2)}, X^{(3)}]$

$Y_3 \leftarrow \text{DWConv}_{3\times 3}(X^{(1)})$

$Y_5 \leftarrow \text{DWConv}_{5\times 5}(X^{(2)})$

$Y_7 \leftarrow \text{DWConv}_{7\times 7}(X^{(3)})$

$Y \leftarrow \text{Concat}(Y_3,\ Y_5,\ Y_7)$

`/* Dynamic positional encoding (DPE) */`

$Y \leftarrow Y + \text{DWConv}_{3\times 3}(Y)$

`/* Token-wise KAN enhancement for structural expressiveness */`

$T \leftarrow \text{Flatten2D}(Y)$ // $T \in \mathbb{R}^{B\times (HW)\times C_e}$

for $\ell = 1$ **to** L **do**

  $T \leftarrow \text{KANBlock}_\ell(T,\ H,\ W)$

$T \leftarrow \text{LayerNorm}(T)$

$Y_{\text{kan}} \leftarrow \text{Unflatten2D}(T,\ H,\ W)$ // $\in \mathbb{R}^{B\times C_e\times H\times W}$

`// Output projection with residual refinement`

$O \leftarrow \text{GELU}(\text{Conv}_{3\times 3}(Y_{\text{kan}})) + X_r$

$S \leftarrow O$

return $(O,\ S)$

KAN formulation. Given an input token sequence $Z \in \mathbb{R}^{(HW)\times d}$, a K-layer KAN is constructed by composing multiple KAN layers:

$$\mathrm{KAN}(Z) = (\Phi_{K-1} \circ \Phi_{K-2} \circ \cdots \circ \Phi_1 \circ \Phi_0)(Z), \tag{1}$$

where Φ_k denotes the k-th KAN layer and $\circ$ represents function composition.

Each layer Φ_k maps an n_{in}-dimensional input to an n_{out}-dimensional output and is parameterized by a matrix of learnable univariate activation functions on edges:

$$\Phi_k = \left[\phi_{p,q}^{(k)}(\cdot)\right]_{p=1,q=1}^{n_{\text{in}},\,n_{\text{out}}}, \qquad \phi_{p,q}^{(k)} : \mathbb{R} \to \mathbb{R}. \tag{2}$$

The forward computation within layer k is defined element-wise as

$$z_{t,q}^{k+1} = \sum_{p=1}^{n_{\text{in}}} \phi_{p,q}^{(k)}\left(z_{t,p}^{k}\right), \qquad q = 1, \ldots, n_{\text{out}}, \tag{3}$$

where $t \in \{1, \ldots, HW\}$ indexes tokens after flattening the 2D feature map and p and q index the input and output channels of the KAN layer, respectively. Each $\varphi_{p,q}^{(k)}(\cdot)$ is a learnable univariate function defined on the edge (p, q).

3.3. Visual State Space Bridge (VSS-Bridge)

VMamba [18] models long-range dependencies with near-linear complexity and performs well on high-resolution remote sensing imagery. Motivated by this, we adopt the Visual State Space (VSS) module from VMamba as the sequential modeling component in our decoder. To support efficient VSS-based decoding, we introduce VSS-Bridge. It restructures heterogeneous cross-layer features and explicitly projects them into the VSS domain, as shown Figure 3.

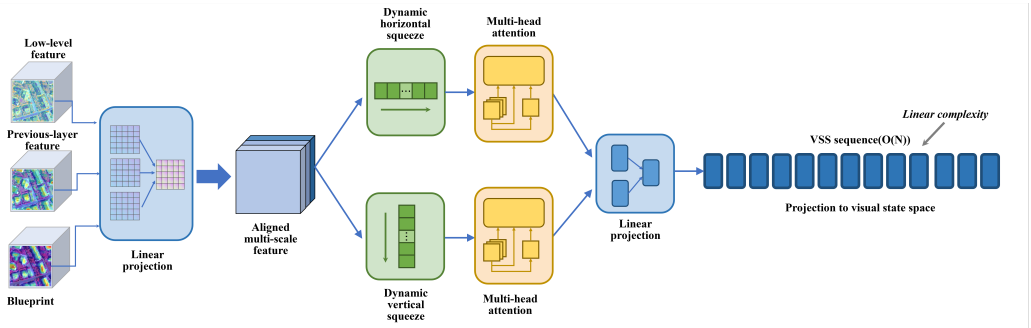


Figure 3. Architecture of the proposed Visual State Space Bridge (VSS-Bridge).

We implement the VSS-Bridge using a lightweight dynamic axial attention block with structural-guided gating. Specifically, x_0 denotes the current decoder feature, and $x_n = \{x_1, \ldots, x_t\}$ denotes the collected cross-layer features. C, H, and W represent the channel number and the spatial height and width, respectively. In addition, we denote by b the structural blueprint generated by the SBG module. To explicitly inject structural priors into the decoder node, we first modulate x_0 using a gating mechanism derived from b:

$$m = \sigma\big(\mathrm{Conv}_{1\times1}(\mathrm{Up}(b))\big), \tag{4}$$

$$\tilde{x}_0 = x_0 \odot m + x_0 = (1 + m) \odot x_0, \tag{5}$$

where $\mathrm{Up}(\cdot)$ resizes the blueprint b to the spatial resolution $H \times W$. The 1×1 convolution outputs C channels, so that $m \in \mathbb{R}^{B\times C\times H\times W}$ can gate x_0 via element-wise multiplication, and $\odot$ denotes the hadamard (element-wise) product.

We first concatenate the features in x_n along the channel dimension. The concatenated features are then projected to $2C$ channels through a linear layer and evenly split to form key and value candidates. In parallel, the current node feature $\tilde{x}_0$ is mapped to produce the query, key, and value features F_q, F_k, and F_v:

$$s = \text{Linear}(\text{LayerNorm}(\text{Concat}(x_n))), \tag{6}$$

$$k, v = \text{chunk}(s, 2, \text{dim} = 1), \tag{7}$$

$$F_q = Q(\tilde{x}_0), \quad F_k = K(k), \quad F_v = V(v), \tag{8}$$

where $F_q, F_k, F_v \in \mathbb{R}^{C\times H\times W}$, $\text{chunk}(\cdot)$ denotes splitting the tensor along the specified dimension, and $Q(\cdot)$, $K(\cdot)$, and $V(\cdot)$ denote linear projection functions for generating queries, keys, and values, respectively.

To reduce the computational and memory overhead associated with standard 2D self-attention, we employ a dynamic axial attention mechanism that performs interaction along one axis at a time. Specifically, we squeeze one spatial dimension via average pooling to obtain 1D axial tokens. For the row branch, we average over the width dimension:

$$\begin{aligned} q_{\text{row}}(c,h) &= \frac{1}{W}\sum_{w=1}^{W} F_q(c,h,w), \\ k_{\text{row}}(c,h) &= \frac{1}{W}\sum_{w=1}^{W} F_k(c,h,w), \\ v_{\text{row}}(c,h) &= \frac{1}{W}\sum_{w=1}^{W} F_v(c,h,w). \end{aligned} \tag{9}$$

where $q_{\text{row}}, k_{\text{row}}, v_{\text{row}} \in \mathbb{R}^{C\times H}$. Similarly, for the column branch, we average over the height dimension to obtain $q_{\text{col}}, k_{\text{col}}, v_{\text{col}} \in \mathbb{R}^{C\times W}$.

We then apply multi-head scaled dot-product attention (SDPA) to improve representation capacity. Let M be the number of attention heads and D the per-head dimensionality, such that $C = M \cdot D$. We reshape axial tokens into multi-head form:

$$\hat{Q}_{\text{row}}, \hat{K}_{\text{row}}, \hat{V}_{\text{row}} \in \mathbb{R}^{M\times H\times D}, \quad \hat{Q}_{\text{col}}, \hat{K}_{\text{col}}, \hat{V}_{\text{col}} \in \mathbb{R}^{M\times W\times D}. \tag{10}$$

To encode positional information in axial sequences, we introduce a learnable 1D positional embedding and interpolate it to the current length $N \in \{H, W\}$:

$$P_N = \text{Interp}(P, \text{size} = N), \tag{11}$$

where P is a fixed-size learnable embedding and $\text{Interp}(\cdot)$ denotes linear interpolation.

The row-wise and column-wise attentions are computed as

$$A_{\text{row}} = \text{softmax}\left(\frac{(\hat{Q}_{\text{row}} + P_H^q)(\hat{K}_{\text{row}} + P_H^k)^\top}{\sqrt{D}}\right)\hat{V}_{\text{row}}, \tag{12}$$

$$A_{\text{col}} = \text{softmax}\left(\frac{(\hat{Q}_{\text{col}} + P_W^q)(\hat{K}_{\text{col}} + P_W^k)^\top}{\sqrt{D}}\right)\hat{V}_{\text{col}}, \tag{13}$$

where the $\text{softmax}(\cdot)$ in axial attention is applied along the axial sequence length (i.e., H for row-wise attention and W for column-wise attention). $\sqrt{D}$ stabilizes the attention logits by the per-head dimensionality.

In the final fusion step, we broadcast the axial responses back to the 2D layout and introduce learnable mixing weights α and β to improve robustness:

$$\mathrm{Attn} = F_v + \alpha A_{\mathrm{row}}^{\mathrm{expand}} + \beta A_{\mathrm{col}}^{\mathrm{expand}} \in \mathbb{R}^{C\times H\times W}, \tag{14}$$

where $A_{\mathrm{row}}^{\mathrm{expand}}$ is broadcast from $\mathbb{R}^{C\times H}$ to $\mathbb{R}^{C\times H\times W}$ by replication along the width dimension and $A_{\mathrm{col}}^{\mathrm{expand}}$ is broadcast by replication along the height dimension. $\alpha, \beta \in \mathbb{R}^{C}$ are learnable per-channel weights and are broadcast spatially.

The gated current-node feature $\tilde{x}_0$ and the attention-enhanced feature Attn are concatenated along the channel dimension, followed by channel shuffling and a linear projection to obtain the aggregated output Y:

$$Y = \mathrm{Linear}\big(\mathrm{Shuffle}(\mathrm{Concat}[\tilde{x}_0, \mathrm{Attn}])\big) \in \mathbb{R}^{C\times H\times W}, \tag{15}$$

where $\mathrm{Shuffle}(\cdot)$ denotes channel reordering and $\mathrm{Concat}(\cdot)$ denotes feature concatenation.

Finally, to interface with the VSS module, we flatten the aggregated feature and reshape it back after VSS decoding:

$$Y^{\mathrm{vss}} = \mathrm{Unflatten}\big(\mathrm{VSS}\big(\mathrm{Flatten}(Y)\big)\big) \in \mathbb{R}^{C\times H\times W}, \tag{16}$$

where $Y \in \mathbb{R}^{C\times H\times W}$ and $\mathrm{Flatten}(\cdot)$ follows raster order.

In this way, the bridge mitigates semantic gaps and redundancy across layers via axial cross-layer filtering. It also converts spatial interactions into a sequence-compatible representation for efficient visual state space decoding.

3.4. Structural Repair Block (SRB)

In the final stage of the decoder, deeper layers provide features enriched with multi-scale contextual information; however, repeated upsampling and multi-scale feature fusion may also accumulate noise and introduce structural distortions. Among these factors, interpolation-based upsampling is a key contributor. As a fixed-kernel resampling operation, interpolation smooths sharp contours and attenuates boundary responses of fine structures. As shown in Figure 4, bilinear resampling reduces the number of connected components while increasing the dominance of the largest connected component, indicating a tendency to merge fragmented regions and suppress minute or marginal structures. This behavior broadens the low-confidence transition band near object boundaries and may alter the topology of fine-grained regions. In practical decoders, boundary ambiguity and potential sub-pixel inconsistencies introduced by successive resampling and fusion can be further amplified by subsequent convolutional operations, leading to unstable boundary predictions.

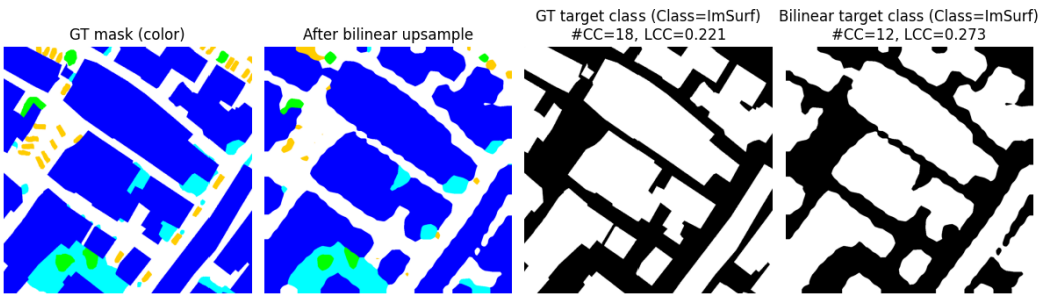


Figure 4. Effect of bilinear upsampling on structural connectivity.

To address this issue, we design a lightweight Structural Repair Block (SRB). Large-kernel attention expands the receptive field and provides sufficient structural context for

boundary decisions while preserving geometric continuity. Spatial and channel gating suppress unreliable boundary responses and selectively enhance geometrically coherent regions. As a result, SRB mitigates topology drift caused by over-smoothing and boundary blurring.

As illustrated in Figure 5, the Structural Repair Block (SRB) is designed to refine high-resolution decoder features by enlarging the receptive field while suppressing accumulated noise. Given an input feature map $X \in \mathbb{R}^{C\times H\times W}$, SRB first adopts a lightweight large-kernel attention implemented by cascading small-kernel deep convolutions [13]:

$$A = \phi_{5\times5}(X), \qquad B = \phi_{7\times7,d=3}(A), \qquad A, B \in \mathbb{R}^{C\times H\times W}. \tag{17}$$

where C denotes the channel number and H and W denote the spatial height and width, respectively; $\phi_{5\times5}(\cdot)$ denotes a 5×5 convolution, and $\phi_{7\times7,d=3}(\cdot)$ denotes a 7×7 dilated convolution with dilation rate $d=3$.

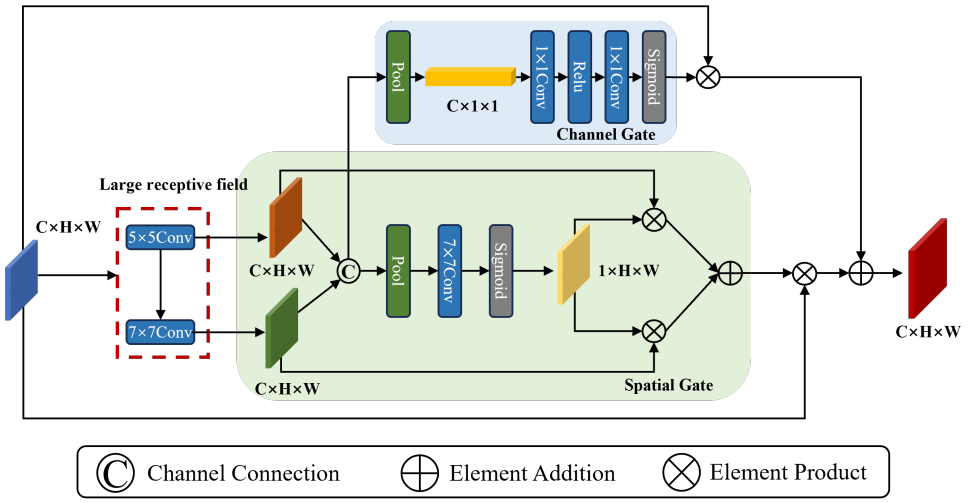


Figure 5. Structure of the Structural Repair Block (SRB).

For dilated convolution, the effective kernel size is

$$k_{\text{eff}} = k + (k-1)(d-1), \tag{18}$$

where k denotes the kernel size, d denotes the dilation rate, and k_{eff} is the resulting effective kernel size (receptive field) of the dilated convolution. This yields a substantially enlarged receptive field with $(k,d) = (7,3)$.

The two branches are then compressed and fused by 1×1 projections:

$$F = \text{Concat}\Big(\phi_{1\times1}^{C/2}(A),\ \phi_{1\times1}^{C/2}(B)\Big) \in \mathbb{R}^{C\times H\times W}. \tag{19}$$

where $\phi_{1\times1}^{C/2}(\cdot)$ denotes a 1×1 convolution that projects features to $C/2$ channels, and $\text{Concat}(\cdot)$ indicates concatenation along the channel dimension.

Next, SRB applies spatial and channel gating in parallel. The spatial attention is computed from channel-wise average/max pooled maps:

$$M_s = \sigma\Big(\phi_{7\times7}\big(\text{Concat}(\text{AvgPool}_c(F),\ \text{MaxPool}_c(F))\big)\Big) \in \mathbb{R}^{1\times H\times W}. \tag{20}$$

where $\text{AvgPool}_c(\cdot)$ and $\text{MaxPool}_c(\cdot)$ denote channel-wise average pooling and max pooling, respectively; $\sigma(\cdot)$ is the sigmoid activation; and $\phi_{7\times7}(\cdot)$ denotes a 7×7 convolution.

The channel attention is obtained by global average/max pooling followed by an MLP:

$$M_c = \sigma\Big(\mathrm{MLP}(\mathrm{GAP}(F)) + \mathrm{MLP}(\mathrm{GMP}(F))\Big) \in \mathbb{R}^C. \tag{21}$$

where $\mathrm{GAP}(\cdot)$ and $\mathrm{GMP}(\cdot)$ denote global average pooling and global max pooling, respectively; $\mathrm{MLP}(\cdot)$ denotes a multi-layer perceptron; and $\sigma(\cdot)$ is the sigmoid activation.

Finally, the gated features are fused and refined with a residual connection:

$$Y = X + \big(F \odot M_s + F \odot M_c\big), \qquad Y \in \mathbb{R}^{C\times H\times W}. \tag{22}$$

where $\odot$ denotes the Hadamard (element-wise) product. $M_s \in \mathbb{R}^{1\times H\times W}$ is broadcast along the channel dimension, and $M_c \in \mathbb{R}^{C\times 1\times 1}$ is broadcast spatially.

3.5. Loss Function

Lite-BSSNet adopts a combined loss strategy during training, consisting of two parts: the main loss function $\mathcal{L}_m$ for overall optimization and the auxiliary loss function $\mathcal{L}_a$ for local supervision. The main loss function $\mathcal{L}_m$ uses dice loss and cross-entropy loss, expressed as

$$\mathcal{L}_{ce} = -\frac{1}{N}\sum_{i=1}^{N}\sum_{c=1}^{C} y_{i,c}\log(\hat{y}_{i,c}) \tag{23}$$

$$\mathcal{L}_{dice} = 1 - \frac{2}{N}\sum_{i=1}^{N}\sum_{c=1}^{C}\frac{y_{i,c}\hat{y}_{i,c}}{y_{i,c}+\hat{y}_{i,c}} \tag{24}$$

$$\mathcal{L}_m = \mathcal{L}_{ce} + \mathcal{L}_{dice} \tag{25}$$

where N represents the number of pixels, C represents the number of sample classes, $y_{i,c}$ represents the ground truth, and $\hat{y}_{i,c}$ represents the probability that the i-th sample is predicted to be class c.

The auxiliary loss $\mathcal{L}_a$ is computed using a cross-entropy loss and applied in the auxiliary head. The auxiliary head takes as input the fused features from the SBG module and two sets of VSS modules. Its output is produced through a 3 × 3 convolutional layer followed by batch normalization and ReLU activation and then a 1 × 1 convolutional layer and an upsampling operation. During training, the auxiliary loss is scaled by a weighting coefficient λ to balance the main loss $\mathcal{L}_m$ and the auxiliary loss $\mathcal{L}_a$. Accordingly, the total loss $\mathcal{L}$ is defined as

$$\mathcal{L} = \mathcal{L}_m + \lambda\mathcal{L}_a \tag{26}$$

where λ is set to 0.4 [9] and $\mathcal{L}_a$ is applied to an intermediate prediction head to stabilize optimization.

4. Experiment

4.1. Datasets

(1) *ISPRS Vaihingen*: The Vaihingen dataset consists of 33 top image tiles with a spatial resolution of 9 cm, including RGB and near-infrared (NIR) bands, with an average size of 2500 × 2000 pixels. The images cover an urban area, and the scenes contain a variety of complex features, mainly divided into the following six classes: buildings, trees, low vegetation, vehicles, impervious surfaces, and miscellaneous objects. Following the same experimental protocol as UNetFormer [9], we select the following IDs: 2, 4, 6, 8, 10, 12, 14, 16, 20, 22, 24, 27, 29, 31, 33, 35, and 38 for testing, ID 30 for validation, and the remaining 15 images for training. The original image tiles are cropped to 1024 × 1024 pixels.

(2) *ISPRS Potsdam*: The Potsdam dataset contains 38 top image tiles with a spatial resolution of 5 cm, including RGB and NIR bands, with an average size of 6000 × 6000 pixels. The dataset classes are the same as those of the Vaihingen dataset. Following the same experimental protocol as UNetFormer [9], we select the following IDs: 2_13, 2_14, 3_13, 3_14, 4_13, 4_14, 4_15, 5_13, 5_14, 5_15, 6_13, 6_14, 6_15, and 7_13 for testing, and ID 2_10 is used for validation. The remaining 22 images (except ID 7_10 with error annotations) are used for training. The original image tiles are cropped to 1024 × 1024 pixels.

4.2. Implementation Details

The model was implemented and trained using PyTorch 2.0.0 on a single NVIDIA RTX 4090 GPU. For the ISPRS Vaihingen and Potsdam datasets, the AdamW optimizer was adopted to ensure stable and accelerated convergence. The learning rate followed a cosine annealing schedule, starting from an initial value of 6×10^{-4}. Images were cropped into 512 × 512-pixel patches and subjected to random scaling ([0.5, 0.75, 1.0, 1.25, 1.5]) and random horizontal and vertical flipping. Training was conducted for 100 epochs with a batch size of 8. During testing, test-time augmentation (TTA) strategies including multi-scale and random flipping were applied.

4.3. Evaluation Metrics

For quantitative analysis, three categories of evaluation metrics were used. The first class evaluates the segmentation performance. We used the overall accuracy (OA), mean F1 score (mF1), mean intersection over union (mIoU), mean boundary F1 score (mbF1), and mean centerline Dice score (mclDice). The meanings of these variables are provided by the following equations:

$$OA = \frac{\sum_{k=1}^{K} TP_k}{\sum_{k=1}^{K}(TP_k + FP_k + FN_k + TN_k)} \tag{27}$$

$$mF1 = \frac{1}{K}\sum_{k=1}^{K}\frac{2 \times TP_k}{2 \times TP_k + FP_k + FN_k} \tag{28}$$

$$mIoU = \frac{1}{K}\sum_{k=1}^{K}\frac{TP_k}{TP_k + FP_k + FN_k} \tag{29}$$

$$mbF1 = \frac{1}{K}\sum_{k=1}^{K}\frac{2 \times P_k^b \times R_k^b}{P_k^b + R_k^b} \tag{30}$$

where TP_k, FP_k, FN_k, and TN_k represent the true positive, true negative, false positive, and false negative samples of the class k, respectively. For the boundary metric, P_k^b and R_k^b denote the boundary precision and boundary recall of class k.

To evaluate topological consistency, we additionally adopt clDice [53]. For class k, let P_k and G_k denote the predicted and ground-truth binary masks, and let $\mathcal{S}(\cdot)$ be a skeletonization operator. The topological precision and recall are defined as

$$tPrec_k = \frac{|\mathcal{S}(P_k) \cap G_k|}{|\mathcal{S}(P_k)|}, \quad tRec_k = \frac{|\mathcal{S}(G_k) \cap P_k|}{|\mathcal{S}(G_k)|}, \tag{31}$$

Accordingly, the mclDice over K classes is

$$mclDice = \frac{1}{K}\sum_{k=1}^{K}\frac{2 \times tPrec_k \times tRec_k}{tPrec_k + tRec_k}. \tag{32}$$

The second class evaluates the scan-friendly properties of intermediate representations, which are important for Visual State Space (VSS) modeling. We adopt three indicators: total variation (TV), scan-wise variation (Scan), and scan-wise peak variation (Peak). Total variation measures spatial smoothness of feature maps and is defined as

$$TV = \mathbb{E}\big[|X_{c,h+1,w} - X_{c,h,w}|\big] + \mathbb{E}\big[|X_{c,h,w+1} - X_{c,h,w}|\big], \tag{33}$$

where $X \in \mathbb{R}^{C\times H\times W}$ is a feature map and $X_{c,h,w}$ denotes the value at channel c and location (h, w).

To assess suitability for sequential scanning, the feature map is flattened into a 1D sequence following a fixed raster scan order. The scan-wise variation and scan-wise peak variation are defined as

$$Scan = \mathbb{E}\big[|x_{t+1} - x_t|_2\big], \tag{34}$$

$$Peak = \max_t |x_{t+1} - x_t|_2, \tag{35}$$

where $E[\cdot]$ denotes the empirical mean over all valid indices. X is flattened into a 1D sequence $\{x_t\}_{t=1}^{HW}$ in row-major order, where $x_t \in \mathbb{R}^C$.

For comparison, we present the reduction ratio (after/before) of these metrics, where lower values indicate smoother and more scan-friendly representations.

The third class evaluates the memory requirements and running speed of the model. Memory requirements are measured by the number of model parameters (P), while running speed is evaluated by the number of floating point operations (FLOPs) and the frames per second (FPS).

4.4. Performance Comparison and Visualization

In this experiment, we selected several SOTA semantic segmentation methods and conducted a comprehensive comparison on the ISPRS Vaihingen and Potsdam datasets. The compared methods include CNN-based models: ABCNet [36], SwiftNet [54], and PACSCNet [55]; Transformer-based models: Segmenter [56], BANet [57], and HST-UNet [58]; Mamba-based models: RS3Mamba [21], UNetMamba [22], Samba [44], PPMamba [59], and PyramidMamba [60]; hybrid-based models: UNetFormer [9], CMLFormer [10], UrbanSSF-T [20], and UMFormer [46].

(1) *Results on the ISPRS Vaihingen dataset*

The ISPRS Vaihingen dataset has been widely adopted for semantic segmentation tasks owing to its high-resolution imagery, diverse object categories, and high-quality annotations. As shown in Table 1, Lite-BSSNet achieves the highest mIoU (83.9%) and mF1 (91.1%) on the Vaihingen test set, surpassing existing CNN-, ViT-, Mamba-, and hybrid-based models. Notably, Lite-BSSNet maintains high segmentation accuracy while achieving competitive performance in terms of memory and computational efficiency. The total number of parameters (12.3 M) is second only to the lightweight model UNetFormer, whereas the FLOPs (45.4 G) are the lowest among all compared models. These results indicate that Lite-BSSNet not only delivers superior classification performance but also achieves an optimized balance between memory consumption and computational efficiency. Furthermore, the lightweight design of Lite-BSSNet enables efficient operation under resource-constrained and real-time conditions, leading to improvements in processing efficiency and reductions in inference latency, suggesting the possibility of practical deployment.

However, the overall accuracy (OA) of Lite-BSSNet did not reach the highest level, which may be because OA primarily reflects the overall proportion of correctly classified pixels and is heavily influenced by dominant classes. In contrast, Lite-BSSNet achieves relatively balanced performance across all classes, maintaining consistent accuracy rather

than favoring dominant categories. By comparison, UrbanSSF-T exhibits exceptionally high accuracy in the Impervious Surface (ImSurf) class, which disproportionately elevates its overall OA score.

Table 1. Results on the ISPRS Vaihingen test dataset, where "—" denotes missing values. bold and underlined text represent the best and second-best results, respectively. "↑" indicates better results with higher values.

Type	Method	Backbone	ImSurf ↑	Building ↑	LowVeg ↑	Tree ↑	Car ↑	mF1 ↑	OA ↑	mIoU ↑
CNN	ABCNet [36]	ResNet-18	92.7	95.2	84.5	89.7	85.3	89.5	90.7	81.3
	SwiftNet [54]	ResNet-18	92.2	94.8	84.1	89.3	81.2	90.2	88.3	79.6
	PACSCNet [55]	ResNet-50	—	—	—	—	—	—	90.0	82.3
Transformer	Segmenter [56]	ViT-Tiny	89.8	93.0	81.2	88.9	67.6	84.1	88.1	73.6
	BANet [57]	ResT-Lite	92.2	95.2	83.8	89.9	86.8	89.6	90.5	81.4
	HST-UNet [58]	—	77.9	85.4	60.3	72.8	62.0	83.0	—	71.4
Mamba	RS3Mamba [21]	R18-VMamba-T	92.8	**96.8**	80.8	**91.1**	90.1	90.3	—	82.8
	UNetMamba [22]	ResT-Lite	92.3	95.3	83.9	89.8	87.6	89.8	90.5	81.7
	Samba [44]	Samba	—	—	—	—	—	84.2	—	73.6
	PPMamba [59]	ResNet-18	93.0	95.9	84.1	89.8	87.2	90.0	90.9	82.0
	PyramidMamba [60]	ResNet-18	96.6	95.3	83.9	89.7	86.5	90.4	93.0	82.9
Hybrid	UNetFormer [9]	ResNet-18	92.9	95.5	84.5	89.8	87.3	90.0	90.9	82.1
	CMLFormer [10]	ResNet-18	88.0	91.9	77.3	84.9	78.8	84.2	—	73.1
	UrbanSSF-T [20]	MobileNet-V3	96.6	95.5	84.4	90.0	87.2	90.7	**93.1**	83.3
	UMFormer [46]	ResNet-18	**96.7**	95.2	83.8	89.5	88.1	90.7	93.0	83.3
Hybrid	Lite-BSSNet	RegNet-Y	93.5	96.1	**85.4**	90.3	**90.3**	**91.1**	91.6	**83.9**

(2) *Results on the ISPRS Potsdam dataset*

To further evaluate the effectiveness of Lite-BSSNet, we also performed experiments on the ISPRS Potsdam dataset, using a processing method similar to that of the ISPRS Vaihingen dataset. Unlike the ISPRS Vaihingen dataset, the ISPRS Potsdam dataset has higher image resolution and point-cloud density, making it suitable for finer-grained feature classification. As shown in Table 2, Lite-BSSNet achieved the highest mIoU (86.7%) and mF1 (92.7%) on the Potsdam test dateset. It should be noted that our method achieved an F1 score of 88.3% in the low vegetation class, exceeding other networks by more than 0.3%.

Table 2. Results on the ISPRS Potsdam test dataset, where "—" denotes missing values, bold and underlined text represent the best and second-best results, respectively. "↑" indicates better results with higher values.

Type	Method	Backbone	ImSurf ↑	Building ↑	LowVeg ↑	Tree ↑	Car ↑	mF1 ↑	OA ↑	mIoU ↑
CNN	ABCNet [36]	ResNet-18	93.0	96.1	87.0	88.4	94.9	91.9	90.7	85.2
	SwiftNet [54]	ResNet-18	91.8	95.9	85.7	86.8	94.5	91.0	89.3	83.8
	PACSCNet [55]	ResNet-50	—	—	—	—	—	—	85.2	76.0
Transformer	Segmenter [56]	ViT-Tiny	91.5	95.3	85.4	85.0	88.5	88.7	89.2	80.7
	BANet [57]	ResT-Lite	93.3	96.7	87.4	89.1	**96.0**	92.5	91.0	86.3
	HST-UNet [58]	—	80.3	87.2	69.7	71.1	78.6	87.1	—	77.4
Mamba	RS3Mamba [21]	R18-VMamba-T	—	—	—	—	—	—	—	—
	UNetMamba [22]	ResT-Lite	92.6	96.3	87.4	88.7	95.2	92.0	90.7	85.5
	Samba [44]	Samba	—	—	—	—	—	—	—	82.3
	PPMamba [59]	ResNet-18	93.3	96.7	88.0	89.1	95.9	92.6	91.3	86.4
	PyramidMamba [60]	ResNet-18	**93.8**	96.4	87.3	88.4	96.0	92.4	91.2	86.2
Hybrid	UNetFormer [9]	ResNet-18	92.3	95.9	86.7	87.6	95.0	91.5	90.2	85.3
	CMLFormer [10]	ResNet-18	90.8	95.0	84.0	83.9	90.3	88.8	—	80.1
	UrbanSSF-T [20]	MobileNet-V3	93.3	95.9	87.2	88.4	95.4	90.9	**92.0**	85.4
	UMFormer [46]	ResNet-18	93.7	96.4	86.7	87.8	95.5	92.0	90.9	85.5
Hybrid	Lite-BSSNet	RegNet-Y	93.2	**96.9**	**88.3**	**89.4**	95.9	**92.7**	91.4	**86.7**

(3) *Visualization of segmentation results*

To enhance the interpretability of Lite-BSSNet, the segmentation results on the Vaihingen and Potsdam datasets are visualized, as shown in Figures 6 and 7. The red areas highlight challenging object regions. By comparing the visualizations, it is evident that Lite-BSSNet produces more accurate segmentation results in challenging regions and generates outputs closer to the ground truth than other models. These results suggest that Lite-BSSNet exhibits a stronger capability to capture fine details and accurately delineate complex object boundaries, even in the presence of occlusions and high-reflectance surfaces.

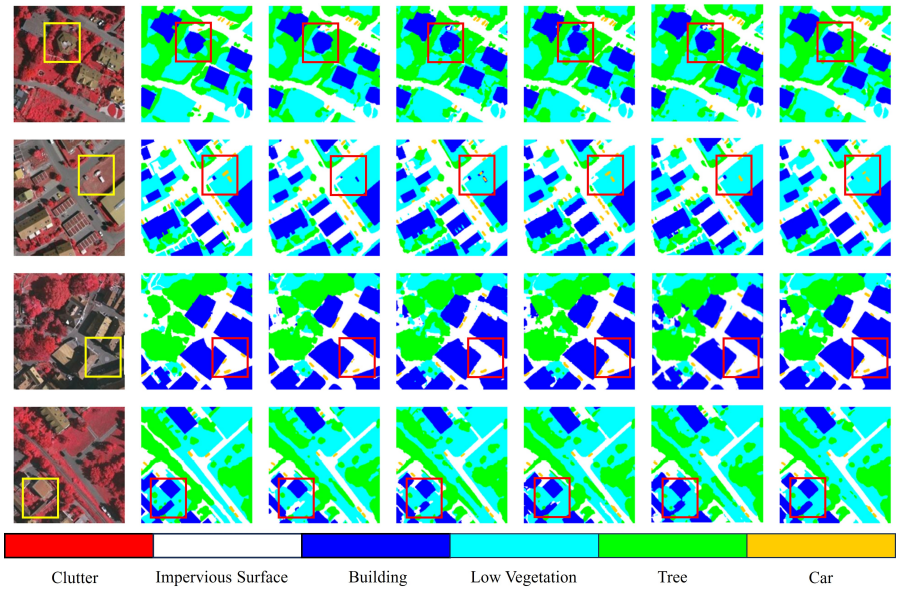


Figure 6. Semantic segmentation results on the ISPRS Vaihingen test set, from left to right: original image, ground truth, ABCNet [36], UNetFormer [9], UNetMamba [22], UMFormer [46], and Lite-BSSNet. The colored bounding boxes highlight representative regions for visual comparison (yellow: input images; red: segmentation maps).

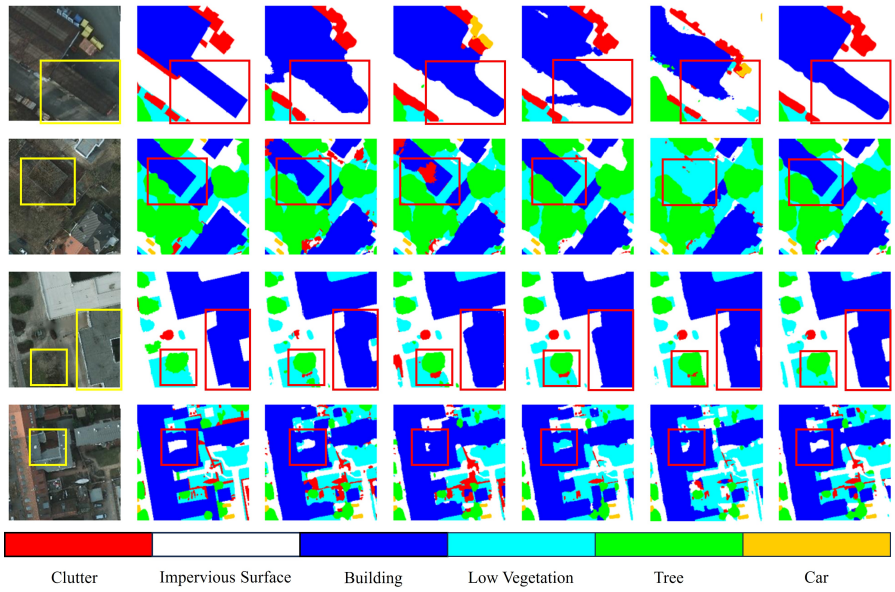


Figure 7. Semantic segmentation results of the ISPRS Potsdam test set, from left to right: original image, ground truth labels, ABCNet [36], UNetFormer [9], UNetMamba [22], UMFormer [46], and Lite-BSSNet. The colored bounding boxes highlight representative regions for visual comparison (yellow: input images; red: segmentation maps).

4.5. Efficiency Analysis

Model complexity and runtime speed are crucial metrics for evaluating segmentation performance, particularly in resource-constrained environments where accurate real-time or near-real-time performance is essential. The performance of Lite-BSSNet on the Vaihingen dataset is summarized in Figure 8. The results show that Lite-BSSNet improves both accuracy and computational efficiency without increasing the parameter count, achieving the highest mIoU (83.9%) and the lowest 45.4 GFLOPs among compared models. Specifically, compared with single-decoder architectures, Lite-BSSNet achieves an average mIoU improvement of 2.2% and a FLOPs reduction of 37.4 G. When compared with existing lightweight hybrid decoder models, it achieves an additional 1.2% improvement in mIoU while maintaining a similar computational cost. The results suggest that Lite-BSSNet achieves a reasonable balance between segmentation accuracy and computational efficiency, showing potential for real-time applications.

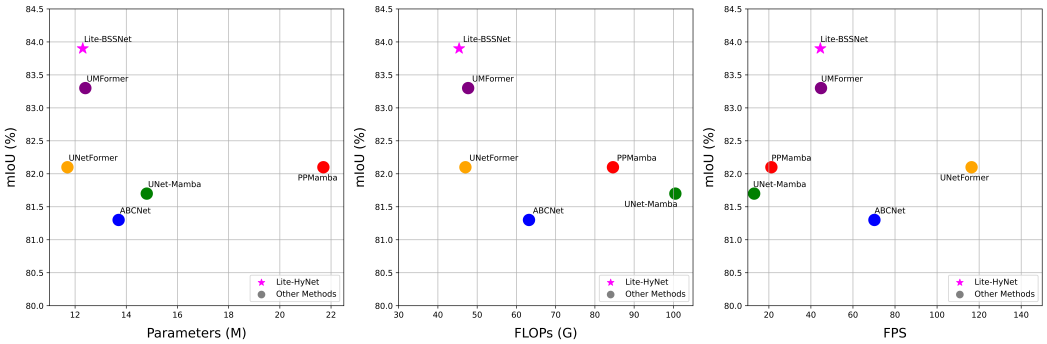


Figure 8. Visualization of comparison results against SOTA lightweight networks. Complexity and speed are measured based on performance on a single NVIDIA RTX 4090 GPU processing with an input size of 1024 × 1024.

However, Lite-BSSNet does not achieve the highest performance in terms of frames per second (FPS). This limitation may result from Mamba's selective state mechanism, which dynamically adjusts parameters based on input characteristics. Although this mechanism enhances the model's adaptability and feature representation capability, it inevitably constrains real-time processing efficiency.

4.6. Ablation Experiment

(1) *The role of each component*

To evaluate the contribution of each core component in Lite-BSSNet, we take the full model as the baseline and conduct a series of ablation experiments on the ISPRS Vaihingen and Potsdam test sets. Specifically, we examine the effectiveness of the three key modules in the decoder: the Structural Blueprint Generator (SBG), the Visual State Space Bridge (VSS-Bridge), and the Structural Repair Block (SRB). By selectively enabling or disabling these modules, we obtain eight model variants that cover all possible combinations, as reported in Table 3.

As shown in Table 3, introducing any single module leads to consistent improvements on both datasets. In particular, enabling VSS-Bridge alone yields a larger gain than using SBG or SRB alone. Specifically, incorporating VSS-Bridge independently increases mIoU by 1.0% and 0.4% across the Vaihingen and Potsdam datasets, highlighting the importance of cross-layer alignment and state-space-aware feature projection. SBG and SRB also deliver significant benefits by enhancing high-level structural priors and optimizing structural refinement, respectively. Combining two modules consistently yields additional gains. For example, using VSS-Bridge and SRB together increases mIoU from 82.1% to 83.6%

on Vaihingen and from 84.9% to 86.6% on Potsdam. The best results are obtained when all three components are enabled: Lite-BSSNet achieves mIoU scores of 83.9% and 86.7% on the Vaihingen and Potsdam datasets, respectively, along with the highest mF1 and overall accuracy (OA). Overall, these results validate the effectiveness of the proposed blueprint–bridge–repair decoding pipeline and confirm that each module contributes to the final performance.

Table 3. Ablation experiment results on the ISPRS Vaihingen and Potsdam test datasets; the best result is shown in bold. Symbols "×" and "✓" indicate the exclusion and inclusion of specific modules, respectively. "↑" indicates better results with higher values.

SBG	VSS-Bridge	SRB	Vaihingen			Potsdam		
			mF1 ↑	OA ↑	mIoU ↑	mF1 ↑	OA ↑	mIoU ↑
×	×	×	90.0	90.6	82.1	91.3	89.9	84.9
✓	×	×	90.3	90.8	82.5	91.3	90.1	85.0
×	×	✓	90.6	91.0	82.9	91.5	90.3	85.2
×	✓	×	90.7	91.3	83.1	91.5	90.4	85.3
✓	×	✓	90.7	91.1	83.1	92.0	90.5	85.5
✓	✓	×	90.5	91.3	83.2	92.3	91.0	86.0
×	✓	✓	90.9	91.3	83.6	92.5	91.3	86.6
✓	✓	✓	**91.1**	**91.6**	**83.9**	**92.7**	**91.4**	**86.7**

(2) *Effectiveness of the blueprint–bridge–repair hybrid decoding structure*

To evaluate the effectiveness of the proposed Lite-BSSNet blueprint–bridge–repair hybrid decoding structure, we replaced its decoder with different single-architecture decoders and conducted experiments on the Vaihingen dataset. Specifically, three types of decoders were compared: a CNN-based decoder (U-Net [7]), a Transformer-based decoder (Swin-UNet [61]), and a Mamba-based decoder (UNetMamba [22]). As shown in Table 4, Lite-BSSNet's hybrid decoder achieves the highest segmentation accuracy, with an mIoU value of 83.9%, averaging approximately 1.2% higher than the three single-architecture decoders. Notably, this improvement does not incur a significant computational burden: Lite-BSSNet requires only 45.4 GFLOPs and attains an inference speed of 44.5 fps, indicating a favorable balance between performance and efficiency.

Table 4. Results of different decoders on the Lite-BSSNet model for the ISPRS Vaihingen test dataset. "↑" indicates higher is better, while "↓" indicates lower is better.

Decoder	P (M) ↓	FLOPs (G) ↓	FPS ↑	mIoU ↑
CNN-based	10.9	39.2	70.2	81.6
Transformer-based	11.7	50.9	43.6	83.2
Mamba-based	11.5	46.4	40.6	83.3
Lite-BSSNet hybrid decoding	12.3	45.4	44.5	83.9

Furthermore, the visual comparison of segmentation results (as shown in Figure 9) shows that Lite-BSSNet more accurately identifies small and partially occluded objects compared with single-architecture decoders. This suggests that the blueprint–bridge–repair decoding pipeline is effective. It combines CNN-style local operators, visual state space modeling, and structural repair. As a result, the model captures global context while preserving local sensitivity, enabling precise recognition of fine-grained targets in complex scenes.

In summary, the proposed Lite-BSSNet hybrid decoder significantly improves segmentation accuracy while maintaining high computational efficiency, validating the rationality and effectiveness of its design for remote sensing semantic segmentation.

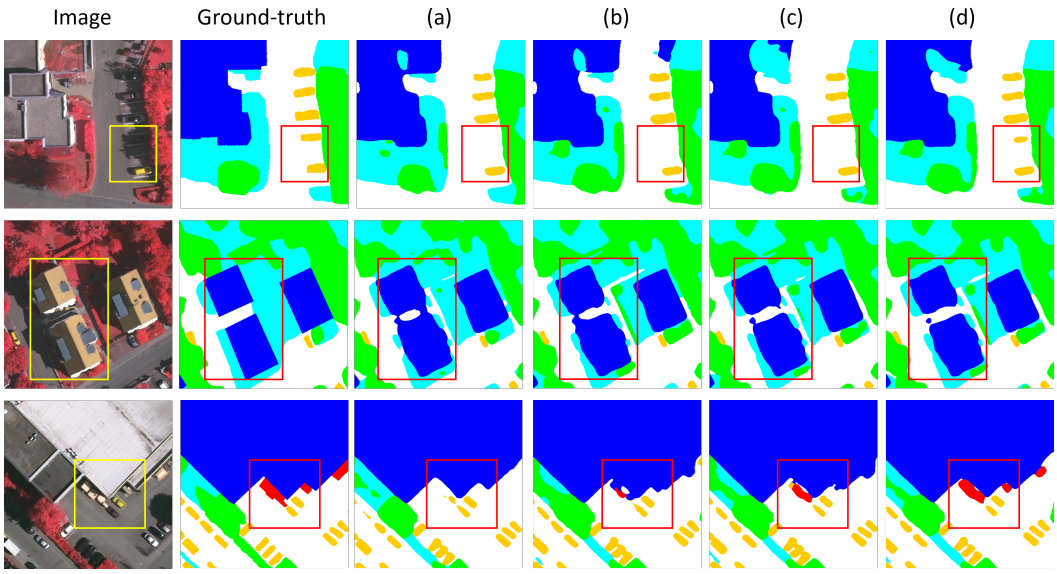


Figure 9. Semantic segmentation results of the Lite-BSSNet model using different decoders. (**a**) CNN based. (**b**) Transformer based. (**c**) Mamba based. (**d**) Lite-BSSNet hybrid decoding. The colored bounding boxes highlight representative regions for visual comparison (yellow: input images; red: segmentation maps).

(3) *Comparison of bridge feature interaction modules*

To further evaluate the effectiveness of the proposed Visual State Space Bridge (VSS-Bridge), we compare it with several alternative cross-layer feature interaction mechanisms: (1) directly concatenating cross-layer features ("Concat"); (2) replacing VSS-Bridge with the SEA module from SeaFormer [62] ("SEA"); (3) replacing it with the WS module from UNetFormer [9] ("WS"); and (4) replacing it with the DMCA module from SparX [24] ("DMCA"). As shown in Table 5, VSS-Bridge achieves the best overall segmentation performance on both the Vaihingen and Potsdam test sets. All interaction-based alternatives outperform the "Concat" baseline, supporting the benefit of explicit cross-layer interaction over naive feature aggregation.

Table 5. Experimental results of the bridge feature interaction module on the ISPRS Vaihingen and Potsdam test datasets; the best result is shown in bold. "↑" indicates higher is better, while "↓" indicates lower is better.

Method	P (M) ↓	FLOPs (G) ↓	Vaihingen			Potsdam		
			mF1 ↑	OA ↑	mIoU ↑	mF1 ↑	OA ↑	mIoU ↑
Concat	11.3	44.2	90.7	91.1	83.1	92.0	90.5	85.5
SEA [62]	12.0	48.7	90.7	91.2	83.2	92.3	90.7	85.8
WS [9]	11.3	44.2	90.8	91.3	83.3	92.6	91.2	86.4
DMCA [24]	11.4	45.1	90.9	91.3	83.5	92.3	91.0	85.9
VSS-Bridge	12.3	45.4	**91.1**	**91.6**	**83.9**	**92.7**	**91.4**	**86.7**

Beyond segmentation accuracy, we further examine the scan-friendly properties of the resulting representations using three indicators: total variation (TV), scan-wise variation, and scan-wise peak variation. As shown in Table 6, "Concat" yields the largest ratios across all three metrics, reflecting substantial spatial and sequential discontinuities. In contrast, all interaction modules reduce these ratios, with VSS-Bridge consistently achieving the lowest values. This suggests that VSS-Bridge suppresses both average and worst-case

token-to-token variations along the scan order. As a result, the representations are better conditioned for the streaming nature of visual state space modeling.

Table 6. Comparison of scan-friendly properties for different interaction modules; ratios are computed as after/before, and lower is better; the best result is shown in bold. "↓" indicates lower is better.

Method	TV Ratio ↓	Scan Ratio ↓	Peak Ratio ↓
Concat	0.2901	0.2797	0.2805
WS [9]	0.2280	0.2235	0.2176
DMCA [24]	0.1690	0.1687	0.1819
SEA [62]	0.1712	0.1709	0.1990
VSS-Bridge	**0.1610**	**0.1619**	**0.1772**

Notably, DMCA attains competitive reductions but remains less effective on the peak metric, indicating weaker control over worst-case transitions. A plausible explanation is that DMCA relies on fixed spatial interaction patterns, whereas VSS-Bridge leverages dynamic axial attention to adapt cross-layer alignment to diverse structural layouts in remote sensing scenes, improving robustness without introducing significant additional parameters or computational overhead.

(4) *Selection of SRB receptive field size*

ConvNet [12] demonstrated that enlarging convolutional kernel sizes can effectively strengthen long-range dependency modeling and improve a model's expressive capability. Following this principle, we evaluate how different receptive field configurations within the proposed Structural Repair Block (SRB) influence the performance of Lite-BSSNet.

We, therefore, compare models configured with SRB receptive fields equivalent to kernel sizes of 5, 11, 23, and 29, along with a baseline in which SRB is removed entirely. Experiments are conducted on both the ISPRS Vaihingen and Potsdam test sets. As reported in Table 7, a receptive field of 23 achieves the best overall performance. Enlarging the receptive field generally enhances segmentation accuracy because wider spatial support enables SRB to better correct upsampling-induced distortions and preserve long, thin structures. However, excessively increasing the receptive field yields diminishing performance gains while introducing additional computational overhead. This result suggests that extremely large kernels do not further improve the structural repair process and may even reduce computational efficiency. Overall, these findings confirm that a moderate large-kernel configuration, such as a receptive field of 23, provides the most effective balance between structural repair strength, gradient smoothing for VSS modeling, and computational efficiency.

Table 7. Experimental results of SRB with different receptive fields on the ISPRS Vaihingen and Potsdam test datasets; the best result is shown in bold. "—" indicates a missing value. "√" indicates inclusion of specific modules. "↑" indicates higher is better, while "↓" indicates lower is better.

Receptive Field	Kernel Size				P (M) ↓	FLOPs (G) ↓	Vaihingen			Potsdam		
	1 × 1	3 × 3	5 × 5	7 × 7			mF1 ↑	OA ↑	mIoU ↑	mF1 ↑	OA ↑	mIoU ↑
Baseline	—	—	—	—	12.3	42.4	90.5	91.3	83.1	92.3	91.0	86.0
5	√	√			12.3	45.1	90.8	91.2	83.3	92.6	91.2	86.4
11		√	√		12.3	45.2	90.9	91.4	83.7	**92.7**	**91.4**	86.6
23			√	√	12.3	45.4	**91.1**	**91.6**	**83.9**	**92.7**	**91.4**	**86.7**
29		√	√	√	12.3	45.9	90.9	91.3	83.5	92.6	91.2	86.4

(5) *Interpretability of the SBG*

To further enhance the interpretability of the proposed Lite-BSSNet, we visualize the outputs of the Structural Blueprint Generator (SBG). Figure 10 presents the Grad-CAM++ [63] heatmaps corresponding to SBG. When the high-level feature extractor is

implemented using a standard MLP (second row), the activations are relatively diffuse and often fail to delineate structure-sensitive regions, leading to inaccurate localization of elongated or boundary-dominant areas such as roads and tree contours. In contrast, after introducing the KAN-based SBG (third row), the activation maps exhibit clearer structural responses, and the highlighted regions become more aligned with the ground truth (first row), especially along class-transition boundaries and thin structures.

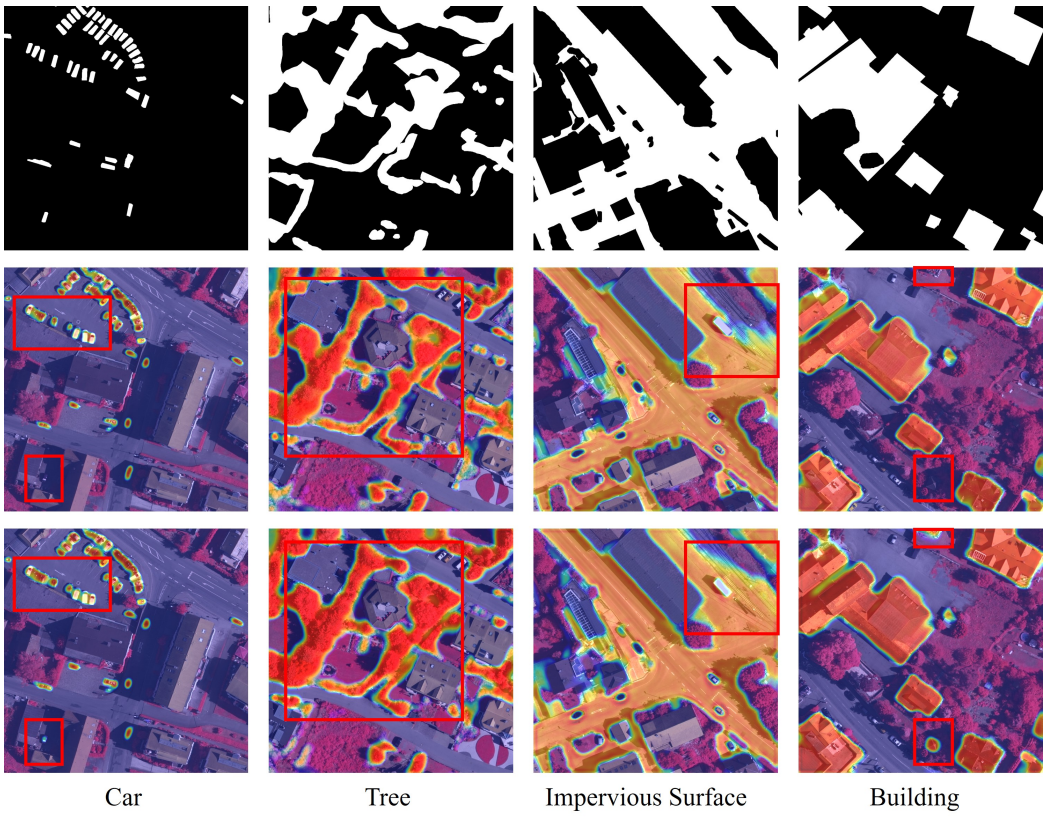


Figure 10. Heatmaps of features output by KAN and MLP on the ISPRS Vaihingen test dataset, where the first row is the ground truth, the second row is the heatmap after MLP processing, and the third row is the heatmap after KAN processing. The red bounding boxes highlight representative regions for visual comparison.

To further evaluate the effectiveness of integrating KAN into SBG, we replaced the MLP in SBG with KAN while keeping all other components unchanged and measured boundary quality using the bF1 score. As shown in Table 8, the KAN variant achieved higher bF1 scores in all categories, indicating enhanced boundary delineation capabilities in the model's output layer. Combined with the visualizations from Grad-CAM++ in Figure 10, these quantitative improvements align with qualitative observations, demonstrating that KAN promotes more discernible cross-channel interactions. This generates structurally enhanced blueprints that better preserve structural continuity.

Table 8. Experimental results of the Structural Blueprint Generator (SBG) instantiated with KAN and a conventional MLP on the ISPRS Vaihingen dataset; the best result is shown in bold.

Method	ImSurf	Building	LowVeg	Tree	Car	mbF1	mIoU
SBG-KAN	**65.1**	**71.9**	**51.8**	**57.6**	**71.2**	**63.5**	**83.9**
SBG-MLP	64.5	71.3	51.1	57.4	70.3	62.9	83.6

(6) *Topology-Consistency Evaluation of SRB*

To verify the effectiveness of the proposed Structural Repair Block (SRB), we conduct an ablation study on SRB. Specifically, we construct two comparative variants: (1) w/o SRB, where SRB is removed from the end of the decoder and only the original upsampling and multi-scale fusion structure is retained; (2) w/ SRB, where SRB is inserted at the same position to perform structured refinement and repair on high-resolution features.

Quantitative results are summarized in Table 9. Compared with w/o SRB, adding SRB yields consistent gains, with larger improvements on categories prone to interpolation smoothing and fusion noise. The overall topology-consistency metric mclDice also increases from 75.9% to 76.6% (+0.7%), indicating improved connectivity and structural integrity of thin structures.

Table 9. Ablation of SRB on the ISPRS Vaihingen test dataset. clDice and its topology-aware decomposition (tPrec/tRec) are reported. Δ denotes (w/ SRB − w/o SRB). Bold values indicate performance improvements, and the improvement magnitude equals the bold number itself.

Method	Segmentation (clDice)						Topology-Aware Connectivity (tPrec/tRec)									
	ImSurf	Building	LowVeg	Tree	Car	mclDice	ImSurf		Building		LowVeg		Tree		Car	
							tPrec	tRec	tPrec	tRec	tPrec	tRec	tPrec	tRec	tPrec	tRec
w/o SRB	79.1	84.7	66.5	74.7	74.4	75.9	73.8	85.3	77.8	94.2	57.7	81.5	68.8	85.7	68.2	84.6
w/ SRB	79.5	85.0	67.6	74.7	76.2	76.6	73.6	87.6	79.1	93.1	62.3	76.4	66.0	90.3	71.4	84.2
Δ	**+0.4**	**+0.3**	**+1.1**	**+0.0**	**+1.8**	**+0.7**	−0.2	**+2.3**	**+1.3**	−1.1	**+4.6**	−5.1	−2.8	**+4.6**	**+3.2**	−0.4

To clarify the source of this improvement, we report the clDice decomposition terms tPrec and tRec. SRB notably increases tPrec on structure-sensitive classes (Building +1.3%, LowVeg +4.6%, Car +3.2%), suggesting fewer spurious or misaligned centerlines near boundaries. It also improves tRec on ImSurf and Tree (+2.3% and +4.6%), implying its ability to effectively suppress unreliable responses near the boundary. Although tRec decreases on LowVeg (−5.1%), the overall gain in tPrec still results in improved clDice and IoU. These results align with SRB's design: large-kernel attention provides long-range structural context, while spatial/channel gating suppresses unreliable boundary responses, jointly reducing topology drift and fragmentation.

5. Discussion

This section further analyzes the experimental results to better understand the advantages and limitations of the proposed Lite-BSSNet. We focus on the "blueprint–bridge–repair" decoding paradigm and discuss its effectiveness from three perspectives: performance on urban benchmark datasets, structural consistency, and computational efficiency. Specifically, we first analyze the segmentation performance and generalization behavior of Lite-BSSNet on different urban remote sensing benchmarks; then, based on a series of ablation studies, we examine the effectiveness of the three core components; finally, we summarize the trade-off between accuracy and efficiency and discuss limitations and future directions.

5.1. Performance on Remote Sensing Benchmarks

The experimental results show that Lite-BSSNet achieves high segmentation accuracy while maintaining a lightweight architectural design. The model obtains competitive mIoU and mF1 scores on both the ISPRS Vaihingen and Potsdam datasets, indicating stable performance trends across scenes with different spatial resolutions, object scales, and texture distributions. Specifically, Lite-BSSNet achieves 83.9% mIoU on the Vaihingen

dataset and 86.7% mIoU on the Potsdam dataset while keeping the parameter scale and computational cost at a relatively low level.

Qualitative visualizations further verify the generalization ability of Lite-BSSNet. As shown in Figures 6 and 7, in complex scenes such as dense building areas and narrow roads, Lite-BSSNet produces clearer boundaries and more continuous predictions and preserves more details in slender structures and complex contours.

5.2. Effectiveness of the Blueprint–Bridge–Repair Pipeline

Ablation studies validate the effectiveness of the three-stage decoding pipeline: enabling any single component (SBG, VSS-Bridge, or SRB) brings consistent gains while enabling all three achieves the best results on both datasets.

Among the three components, VSS-Bridge provides the most prominent standalone improvement: when enabled alone, it increases mIoU by 1.0% on the Vaihingen dataset and by 0.4% on the Potsdam dataset. This indicates that explicit cross-layer alignment before state space decoding is critical. Without this process, heterogeneous features from different encoder stages can introduce redundancy and semantic inconsistency, which may affect the stability of subsequent sequence modeling. Moreover, Table 6 shows that VSS-Bridge achieves the lowest ratios on the scanability metrics (TV, Scan, and Peak). This indicates that it better suppresses both average and worst-case token-to-token variations along the scan order. As a result, the features better match the streaming characteristics of visual state space models.

SBG provides structural constraints for decoding by generating a stable, edge-enhanced structural prior. The ablation results show that introducing SBG yields consistent performance improvements. In addition, interpretability analysis further supports its structural awareness: Grad-CAM++ visualizations show that the KAN-based SBG activations are more focused on structural boundary regions; boundary quality evaluation also shows that SBG-KAN improves the overall mbF1 and brings a slight mIoU gain. These results suggest that SBG helps suppress texture noise interference and enhances boundary-related responses, thereby improving the representation quality of structural details.

SRB is designed to alleviate boundary blurring and structural distortions introduced by interpolation-based upsampling and multi-scale fusion, thereby improving the structural quality of high-resolution outputs. The ablation results indicate that introducing SRB brings consistent gains on both datasets: on Vaihingen, mIoU increases from 82.1% to 82.9%, and on Potsdam, from 84.9% to 85.2%, showing a positive contribution to the final prediction. Further topology consistency evaluation shows that adding SRB increases the overall mclDice from 75.9% to 76.6%, with relatively clear improvements in structural connectivity for structure-sensitive categories such as buildings, low vegetation, and cars. Meanwhile, receptive field configuration experiments show that a moderate effective large receptive field achieves the best balance between performance and overhead: when the effective receptive field continues to increase, performance gains tend to saturate, while computational cost rises. These results are consistent with the design goal of SRB, namely, to provide richer structural context through large-receptive-field modeling and to suppress unreliable responses via spatial/channel gating, thereby mitigating structure breaks and noise accumulation caused by upsampling.

5.3. Balance Between Accuracy and Efficiency

From a deployment perspective, Lite-BSSNet achieves a good balance between segmentation accuracy and computational efficiency. On the Vaihingen dataset, the model achieves the highest mIoU among lightweight networks while maintaining low FLOPs. Compared with single-architecture decoders, the hybrid blueprint–bridge–repair decoding

paradigm improves mIoU by approximately 1.2% without significantly increasing computational burden. This indicates that, compared with simply increasing model capacity or stacking heavier attention modules, restructuring the decoding process in a structured manner can improve performance more efficiently.

Therefore, Lite-BSSNet shows potential for practical remote sensing applications under resource-constrained conditions, such as UAV mapping, urban monitoring, and near-real-time semantic analysis in typical remote sensing task scenarios.

5.4. Limitations and Future Perspectives

Although Lite-BSSNet achieves certain performance improvements, it still has limitations. First, the current implementation does not fully realize the potential inference speed: the selective state mechanism in the visual state space module involves dynamic parameter updates, which may limit frame rate under real-time requirements. Future work could further improve runtime efficiency through state space execution optimization and hardware-aware acceleration.

Second, although the framework performs well on urban remote sensing benchmarks, its effectiveness on non-urban scenes or datasets with substantially different target morphologies remains to be verified. Future work may focus on extending the blueprint–bridge–repair paradigm to broader Mamba architectures, enhancing the interpretability of interactions among the three stages, and improving adaptability to diverse sensing modalities and complex environmental conditions.

6. Conclusions

The proposed framework follows an encoder–decoder architecture. The encoder extracts hierarchical features, and the decoder consists of three components: SBG, VSS-Bridge, and SRB. SBG transforms high-level features into an edge-enhanced structural blueprint that provides a global topological prior. VSS-Bridge aligns and filters multi-scale cross-layer features and projects them into a linear-complexity visual state space, enabling efficient long-range modeling. SRB restores thin structures via large-receptive-field modeling and spatial–channel gating, which reduces upsampling-induced distortions and improves geometric continuity. Extensive experiments on the ISPRS Vaihingen and Potsdam datasets, together with comprehensive ablation studies, demonstrate that Lite-BSSNet achieves state-of-the-art performance among lightweight CNN-, Transformer-, Mamba-, and hybrid-based models. The results confirm that Lite-BSSNet provides a favorable balance between segmentation accuracy, structural consistency, and computational efficiency.

Although Lite-BSSNet exhibits strong performance, future work may focus on improving the interpretability of interactions within the blueprint–bridge–repair decoding pipeline, extending the state space bridging mechanism to broader Mamba-based architectures, and enhancing the model's adaptability to diverse remote sensing conditions and sensor modalities.

Author Contributions: Conceptualization, J.Y.; methodology, J.Y.; software, Y.C.; validation, J.Y.; formal analysis, J.Y.; investigation, J.Y. and Y.X.; resources, J.Y.; data curation, Y.C.; writing—original draft preparation, J.Y.; writing—review and editing, J.Y., Y.X., Y.C., Y.G. and W.L.; visualization, J.Y.; supervision, Y.X., Y.G. and W.L.; funding acquisition, Y.X. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the General Program of National Natural Science Foundation of China (Grant No. 72571281); the Hunan Excellent Young Scientists Fund (Grant No. 2025JJ40066); training Program for Excellent Young Innovators of Changsha (Grant No. kq2209001); the Fund of the National Key Laboratory of Space Target Awareness, and the Fund of Laboratory for Big Data and Decision.

Data Availability Statement: The code will be released at https://github.com/hm-hm53/Lite-BSSNet (accessed on 26 January 2026).

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

References

1. Ding, H.; Xia, B.; Liu, W.; Zhang, Z.; Zhang, J.; Wang, X.; Xu, S. A Novel Mamba Architecture with a Semantic Transformer for Efficient Real-Time Remote Sensing Semantic Segmentation. *Remote Sens.* **2024**, *16*, 2620. [CrossRef]
2. Chen, Y.; Zhang, S.; Han, T.; Du, Y.; Zhang, W.; Li, J. Chat3D: Interactive understanding 3D scene-level point clouds by chatting with foundation model for urban ecological construction. *ISPRS J. Photogramm. Remote Sens.* **2024**, *212*, 181–192. [CrossRef]
3. Yang, H.; Jiang, Z.; Zhang, Y.; Wu, Y.; Luo, H.; Zhang, P.; Wang, B. A high-resolution remote sensing land use/land cover classification method based on multi-level features adaptation of segment anything model. *Int. J. Appl. Earth Obs. Geoinf.* **2025**, *141*, 104659. [CrossRef]
4. Wang, H.; Dai, Y.; Yao, Q.; Ma, L.; Zhang, Z.; Lv, X. Multi-task learning model driven by climate and remote sensing data collaboration for mid-season cotton yield prediction. *Field Crop. Res.* **2025**, *333*, 110070. [CrossRef]
5. Li, Y.; Fu, K.; Sun, H.; Sun, X. An Aircraft Detection Framework Based on Reinforcement Learning and Convolutional Neural Networks in Remote Sensing Images. *Remote Sens.* **2018**, *10*, 243. [CrossRef]
6. Long, J.; Shelhamer, E.; Darrell, T. Fully convolutional networks for semantic segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA, 7–12 June 2015; pp. 3431–3440.
7. Ronneberger, O.; Fischer, P.; Brox, T. U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015*; Springer: Cham, Switzerland, 2015; pp. 234–241.
8. Gao, L.; Liu, H.; Yang, M.; Chen, L.; Wan, Y.; Xiao, Z.; Qian, Y. STransFuse: Fusing Swin Transformer and Convolutional Neural Network for Remote Sensing Image Semantic Segmentation. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2021**, *14*, 10990–11003. [CrossRef]
9. Wang, L.; Li, R.; Zhang, C.; Fang, S.; Duan, C.; Meng, X.; Atkinson, P.M. UNetFormer: A UNet-like transformer for efficient semantic segmentation of remote sensing urban scene imagery. *ISPRS J. Photogramm. Remote Sens.* **2022**, *190*, 196–214. [CrossRef]
10. Wu, H.; Zhang, M.; Huang, P.; Tang, W. CMLFormer: CNN and Multiscale Local-Context Transformer Network for Remote Sensing Images Semantic Segmentation. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2024**, *17*, 7233–7241. [CrossRef]
11. Bao, M.; Lyu, S.; Xu, Z.; Zhou, H.; Ren, J.; Xiang, S.; Li, X.; Cheng, G. Vision Mamba in Remote Sensing: A Comprehensive Survey of Techniques, Applications and Outlook. *arXiv* **2025**, arXiv: 2505.00630. [CrossRef]
12. Liu, Z.; Mao, H.; Wu, C.-Y.; Feichtenhofer, C.; Darrell, T.; Xie, S. A ConvNet for the 2020s. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), New Orleans, LA, USA, 18–24 June 2022; pp. 11966–11976.
13. Li, Y.; Li, X.; Dai, Y.; Hou, Q.; Liu, L.; Liu, Y.; Cheng, M.M.; Yang, J. LSKNet: A Foundation Lightweight Backbone for Remote Sensing. *Int. J. Comput. Vis.* **2025**, *133*, 1410–1431. [CrossRef]
14. Li, D.; Li, L.; Chen, Z.; Li, J. ShiftwiseConv: Small Convolutional Kernel with Large Kernel Effect. In Proceedings of the Computer Vision and Pattern Recognition Conference, Nashville, TN, USA, 11–15 June 2025; pp. 25281–25291.
15. Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; et al. An image is worth 16x16 words: Transformers for image recognition at scale. In Proceedings of the International Conference on Learning Representations, ICLR 2021, Vienna, Austria, 4 May 2021.
16. Wang, W.; Xie, E.; Li, X.; Fan, D.P.; Song, K.; Liang, D.; Lu, T.; Luo, P.; Shao, L. Pyramid vision transformer: A versatile backbone for dense prediction without convolutions. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, QC, Canada, 10–17 October 2021; pp. 568–578.
17. Zhu, L.; Liao, B.; Zhang, Q.; Wang, X.; Liu, W.; Wang, X. Vision mamba: Efficient visual representation learning with bidirectional state space model. In Proceedings of the International Conference on Machine Learning, Vienna, Austria, 21–27 July 2024.
18. Liu, Y.; Tian, Y.; Zhao, Y.; Yu, H.; Xie, L.; Wang, Y.; Ye, Q.; Jiao, J.; Liu, Y. VMamba: Visual State Space Model. *Adv. Neural Inf. Process. Syst.* **2024**, *37*, 103031–103063.
19. Liu, M.; Dan, J.; Lu, Z.; Yu, Y.; Li, Y.; Li, X. CM-UNet: Hybrid CNN-Mamba UNet for Remote Sensing Image Semantic Segmentation. *arXiv* **2024**, arXiv:2405.10530.
20. Wang, Z.; Yi, J.; Chen, A.; Chen, L.; Lin, H.; Xu, K. Accurate semantic segmentation of very high-resolution remote sensing images considering feature state sequences: From benchmark datasets to urban applications. *ISPRS J. Photogramm. Remote Sens.* **2025**, *220*, 824–840. [CrossRef]
21. Ma, X.; Zhang, X.; Pun, M.-O. RS3Mamba: Visual State Space Model for Remote Sensing Image Semantic Segmentation. *IEEE Geosci. Remote Sens. Lett.* **2024**, *21*, 6011405. [CrossRef]
22. Zhu, E.; Chen, Z.; Wang, D.; Shi, H.; Liu, X.; Wang, L. UNetMamba: An Efficient UNet-Like Mamba for Semantic Segmentation of High-Resolution Remote Sensing Images. *IEEE Geosci. Remote Sens. Lett.* **2025**, *22*, 6001205. [CrossRef]

23. Cao, Y.; Liu, C.; Wu, Z.; Zhang, L.; Yang, L. Remote Sensing Image Segmentation Using Vision Mamba and Multi-Scale Multi-Frequency Feature Fusion. *Remote Sens.* **2025**, *17*, 1390. [CrossRef]
24. Lou, M.; Fu, Y.; Yu, Y. Sparx: A sparse cross-layer connection mechanism for hierarchical vision mamba and transformer networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*; AAAI Press: Washington, DC, USA, 2025.
25. Kass, M.; Witkin, A.; Terzopoulos, D. Snakes: Active contour models. *Int. J. Comput. Vis.* **1988**, *1*, 321–331. [CrossRef]
26. Boykov, Y.Y.; Jolly, M.-P. Interactive graph cuts for optimal boundary & region segmentation of objects in N-D images. In Proceedings the Eighth IEEE International Conference on Computer Vision, ICCV 2001, Vancouver, BC, Canada, 7–14 July 2001; pp. 105–112.
27. Xie, S.; Tu, Z. Holistically-Nested Edge Detection. In Proceedings of the 2015 IEEE International Conference on Computer Vision, ICCV 2015, Santiago, Chile, 7–13 December 2015; pp. 1395–1403.
28. You, X.; He, J.; Yang, J.; Gu, Y. Learning With Explicit Shape Priors for Medical Image Segmentation. *IEEE Trans. Med Imaging* **2025**, *44*, 927–940. [CrossRef]
29. Radosavovic, I.; Kosaraju, R.P.; Girshick, R.; He, K.; Dollár, P. Designing Network Design Spaces. In Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR 2020), Seattle, WA, USA, 14–19 June 2020; pp. 10425–10433.
30. Guo, M.; Liu, H.; Xu, Y.; Huang, Y. Building Extraction Based on U-Net with an Attention Block and Multiple Losses. *Remote Sens.* **2020**, *12*, 1400. [CrossRef]
31. Sharma, S.K.; Sharma, V.; Mishra, V. Building Extraction for Urban Infrastructure Mapping Using Deep Neural Networks and High-Resolution Remote Sensing Data. *J. Indian Soc. Remote Sens.* **2025**, *53*, 2223–2237. [CrossRef]
32. Yang, J.; Liu, T.; Zhang, L.; Zhang, Y. Effective Road Segmentation With Selective State-Space Model and Frequency Feature Compensation. *IEEE Trans. Geosci. Remote Sens.* **2025**, *63*, 5603813. [CrossRef]
33. Badrinarayanan, V.; Kendall, A.; Cipolla, R. SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *39*, 2481–2495. [CrossRef] [PubMed]
34. Sun, Y.; Tian, Y.; Xu, Y. Problems of encoder-decoder frameworks for high-resolution remote sensing image segmentation: Structural stereotype and insufficient learning. *Neurocomputing* **2019**, *330*, 297–304. [CrossRef]
35. Teng, J.; Zhu, Z.; Pu, D.; Wang, T.; Zeng, X.; Snoussi, H.; Shi, R.; Sheng, Q.Z.; He, H.; Li, J. MSCSCNet: Multi-Scale Convolution Selective Context Network With Boundary Supervision for Semantic Segmentation of Remote Sensing Images. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2025**, 18, 25892–25908. [CrossRef]
36. Li, R.; Zheng, S.; Zhang, C.; Duan, C.; Wang, L.; Atkinson, P.M. ABCNet: Attentive bilateral contextual network for efficient semantic segmentation of Fine-Resolution remotely sensed imagery. *ISPRS J. Photogramm. Remote Sens.* **2021**, *181*, 84–98. [CrossRef]
37. Fu, J.; Liu, J.; Tian, H.; Li, Y.; Bao, Y.; Fang, Z.; Lu, H. Dual Attention Network for Scene Segmentation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–20 June 2019; pp. 3141–3149.
38. Chen, J.; Lu, Y.; Yu, Q.; Luo, X.; Adeli, E.; Wang, Y.; Lu, L.; Yuille, A.L.; Zhou, Y. TransUNet: Transformers Make Strong Encoders for Medical Image Segmentation. *arXiv* **2021**, arXiv:2102.04306. [CrossRef]
39. Ma, X.; Zhang, X.; Pun, M.-O.; Liu, M. A Multilevel Multimodal Fusion Transformer for Remote Sensing Semantic Segmentation. *IEEE Trans. Geosci. Remote Sens.* **2024**, *62*, 5403215. [CrossRef]
40. Liu, Z.; Lin, Y.; Cao, Y.; Hu, H.; Wei, Y.; Zhang, Z.; Lin, S.; Guo, B. Swin Transformer: Hierarchical Vision Transformer using Shifted Windows. In Proceedings of the 2021 IEEE/CVF International Conference on Computer Vision (ICCV), Montreal, QC, Canada, 11–17 October 2021.
41. He, X.; Zhou, Y.; Zhao, J.; Zhang, D.; Yao, R; Xue, Y. Swin Transformer Embedding UNet for Remote Sensing Image Semantic Segmentation. *IEEE Trans. Geosci. Remote Sens.* **2022**, *60*, 4408715. [CrossRef]
42. Huang, L.; Jiang, B.; Lv, S.; Liu, Y.; Fu, Y. Deep-Learning-Based Semantic Segmentation of Remote Sensing Images: A Survey. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2024**, *17*, 8370–8396. [CrossRef]
43. Chen, H.; Luo, H.; Wang, C. AfaMamba: Adaptive Feature Aggregation With Visual State Space Model for Remote Sensing Images Semantic Segmentation. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2025**, *18*, 8965–8983. [CrossRef]
44. Zhu, Q.; Cai, Y.; Fang, Y.; Yang, Y.; Chen, C.; Fan, L.; Nguyen, A. Samba: Semantic segmentation of remotely sensed images with state space model. *Heliyon* **2024**, *10*, e38495. [CrossRef]
45. Zhao, S.; Chen, H.; Zhang, X.; Xiao, P.; Bai, L.; Ouyang, W. RS-Mamba for Large Remote Sensing Image Dense Prediction. *IEEE Trans. Geosci. Remote Sens.* **2024**, *62*, 5633314. [CrossRef]
46. Li, L.; Yi, J.; Fan, H.; Lin, H. A Lightweight Semantic Segmentation Network Based on Self-Attention Mechanism and State Space Model for Efficient Urban Scene Segmentation. *IEEE Trans. Geosci. Remote Sens.* **2025**, *63*, 4703215. [CrossRef]
47. Chu, X.; Tian, Z.; Zhang, B.; Wang, X.; Shen, C. Conditional Positional Encodings for Vision Transformers. In Proceedings of the 10th International Conference on Learning Representations (ICLR 2022), Online, 25–29 April 2022.

48. Cao, J.; Li, Y.; Sun, M.; Chen, Y.; Lischinski, D.; Cohen-Or, D.; Chen, B.; Tu, C. DO-Conv: Depthwise Over-Parameterized Convolutional Layer. *IEEE Trans. Image Process.* **2022**, *31*, 3726–3736. [CrossRef]
49. Liu, Z.; Wang, Y.; Vaidya, S.; Ruehle, F.; Halverson, J.; Soljacic, M.; Hou, T.Y.; Tegmark, M. KAN: Kolmogorov–Arnold Networks. In Proceedings of the Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, 24–28 April 2025.
50. Li, C.; Liu, X.; Li, W.; Wang, C.; Liu, H.; Liu, Y.; Chen, Z.; Yuan, Y. U-kan makes strong backbone for medical image segmentation and generation. In Proceedings of the AAAI Conference on Artificial Intelligence, Philadelphia, PA, USA, 25 February–4 March 2025.
51. Zhang, S.; Liu, J.; Qian, T.; Li, X. Prompt-Guided Dual-Path UNet with Mamba for Medical Image Segmentation. *arXiv* **2025**, arXiv:2503.19589.
52. Zhang, Y. KM-UNet: KAN Mamba UNet for Medical Image Segmentation. *arXiv* **2025**, arXiv:2501.02559.
53. Shit, S.; Paetzold, J.C.; Sekuboyina, A.; Ezhov, I.; Unger, A.; Zhylka, A.; Pluim, J.P.; Bauer, U.; Menze, B.H. clDice-A Novel Topology-Preserving Loss Function for Tubular Structure Segmentation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 16555–16564.
54. Oršić, M.; Šegvić, S. Efficient semantic segmentation with pyramidal fusion. *Pattern Recognit.* **2021**, *110*, 107611. [CrossRef]
55. Fan, X.; Zhou, W.; Qian, X.; Yan, W. Progressive Adjacent-Layer coordination symmetric cascade network for semantic segmentation of Multimodal remote sensing images. *Expert Syst. Appl.* **2024**, *238*, 121999. [CrossRef]
56. Strudel, R.; Garcia, R.; Laptev, I.; Schmid, C. Segmenter: Transformer for Semantic Segmentation. In Proceedings of the 2021 IEEE/CVF International Conference on Computer Vision (ICCV), Montreal, QC, Canada, 11–17 October 2021; pp. 7242–7252.
57. Wang, L.; Li, R.; Wang, D.; Duan, C.; Wang, T.; Meng, X. Transformer Meets Convolution: A Bilateral Awareness Network for Semantic Segmentation of Very Fine Resolution Urban Scene Images. *Remote Sens.* **2021**, *13*, 3065. [CrossRef]
58. Zhou, H.; Xiao, X.; Li, H.; Liu, X.; Liang, P. Hybrid shunted transformer embedding unet for remote sensing image semantic segmentation. *Neural Comput. Appl.* **2024**, *36*, 15705–15720. [CrossRef]
59. Mu, J.; Zhou, S.; Sun, X. PPMamba: Enhancing Semantic Segmentation in Remote Sensing Imagery by SS2D. *IEEE Geosci. Remote Sens. Lett.* **2025**, *22*, 6001705. [CrossRef]
60. Wang, L.; Li, D.; Dong, S.; Meng, X.; Zhang, X.; Hong, D. PyramidMamba: Rethinking pyramid feature fusion with selective space state model for semantic segmentation of remote sensing imagery. *Int. J. Appl. Earth Obs. Geoinf.* **2025**, *144*, 104884. [CrossRef]
61. Cao, H.; Wang, Y.; Chen, J.; Jiang, D.; Zhang, X.; Tian, Q.; Wang, M. Swin-Unet: Unet-Like Pure Transformer for Medical Image Segmentation. In Proceedings of the Computer Vision–ECCV 2022 Workshops, Tel Aviv, Israel, 23–27 October 2022; pp. 205–218.
62. Wan, Q.; Huang, Z.; Lu, J.; Yu, G.; Zhang, L. SeaFormer: Squeeze-enhanced Axial Transformer for Mobile Semantic Segmentation. In Proceedings of the Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, 1–5 May 2023.
63. Chattopadhay, A.; Sarkar, A.; Howlader, P.; Balasubramanian, V.N. Grad-CAM++: Generalized Gradient-Based Visual Explanations for Deep Convolutional Networks. In Proceedings of the 2018 IEEE Winter Conference on Applications of Computer (WACV), Lake Tahoe, NV, USA, 12–15 March 2018.

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.