


## Article

# LDMNet: Enhancing the Segmentation Capabilities of Unmanned Surface Vehicles in Complex Waterway Scenarios

Tongyang Dai, Huiyu Xiang , Chongjie Leng <sup>\*</sup>, Song Huang, Guanghui He and Shishuo Han

School of Computer and Artificial Intelligence, Beijing Technology and Business University, Beijing 100048, China; 2230601040@st.btbu.edu.cn (T.D.); 2230601007@st.btbu.edu.cn (S.H.); 2230602094@st.btbu.edu.cn (G.H.); 2230601013@st.btbu.edu.cn (S.H.)

<sup>\*</sup> Correspondence: xianghy@th.btbu.edu.cn (H.X.); lcj@th.btbu.edu.cn (C.L.)

**Abstract:** Semantic segmentation-based Complex Waterway Scene Understanding has shown great promise in the environmental perception of Unmanned Surface Vehicles. Existing methods struggle with estimating the edges of obstacles under conditions of blurred water surfaces. To address this, we propose the Lightweight Dual-branch Mamba Network (LDMNet), which includes a CNN-based Deep Dual-branch Network for extracting image features and a Mamba-based fusion module for aggregating and integrating global information. Specifically, we improve the Deep Dual-branch Network structure by incorporating multiple Atrous branches for local fusion; we design a Convolution-based Recombine Attention Module, which serves as the gate activation condition for Mamba-2 to enhance feature interaction and global information fusion from both spatial and channel dimensions. Moreover, to tackle the directional sensitivity of image serialization and the impact of the State Space Model's forgetting strategy on non-causal data modeling, we introduce a Hilbert curve scanning mechanism to achieve multi-scale feature serialization. By stacking feature sequences, we alleviate the local bias of Mamba-2 towards image sequence data. LDMNet integrates the Deep Dual-branch Network, Recombine Attention, and Mamba-2 blocks, effectively capturing the long-range dependencies and multi-scale global context information of Complex Waterway Scene images. The experimental results on four benchmarks show that the proposed LDMNet significantly improves obstacle edge segmentation performance and outperforms existing methods across various performance metrics.

**Keywords:** Deep Dual-branch Network; Mamba-2; complex waterway scenes; attention fusion strategies; obstacle detection



**Citation:** Dai, T.; Xiang, H.; Leng, C.; Huang, S.; He, G.; Han, S. LDMNet: Enhancing the Segmentation Capabilities of Unmanned Surface Vehicles in Complex Waterway Scenarios. *Appl. Sci.* **2024**, *14*, 7706. <https://doi.org/10.3390/app14177706>

Academic Editor: Byung-Gyu Kim

Received: 25 June 2024

Revised: 11 August 2024

Accepted: 22 August 2024

Published: 31 August 2024



**Copyright:** © 2024 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 (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

The rapid development of computer vision and autonomous driving technology has driven the advancement of Unmanned Surface Vehicles (USVs) and has also led to an increasing demand for image analysis in complex environments. Similar to autonomous driving, semantic segmentation in different waterway scenes is a key component of USVs' environmental perception tasks. However, in some scenes, the uncertainty in the size and quantity of obstacles, as well as the impact of weather and reflection conditions on camera imaging, can blur the edges of obstacles in the line of sight. This makes it difficult for traditional semantic segmentation algorithms to achieve good results in such complex waterway scenes. The accuracy of the segmentation algorithm determines whether the USV can detect and avoid nearby obstacles in a timely manner to achieve a high level of autonomous navigation, which is crucial for ensuring the safe navigation of vessels and improving navigation efficiency. Thanks to the exploration in the field of autonomous driving, some works [1–3] have introduced the integration of various sensors such as RADAR and LIDAR and achieved certain results. However, considering the aspect of lightweight, cameras [4–6] still show great potential as portable and flexible sensors.

In recent years, semantic segmentation networks have demonstrated the ability to obtain rich feature information in complex scenes and achieved significant segmentation

results [7–9]. At the same time, attention mechanisms have been proven to significantly enhance the model's global capturing ability regarding key information [10]. However, the encoder–decoder structure based on the Transformer architecture [11] has difficulties in real-time or resource-constrained scenarios like USVs due to hardware utilization defects during attention operations and quadratic time complexity issues. Mamba [12] addresses the two major issues of the Transformer architecture by using hardware-parallel optimization and selective state space to efficiently capture feature information of long sequences, achieving a powerful global context modeling ability with linear complexity. Benefiting from the successful practice of Transformers in the field of vision, Mamba has been rapidly applied to image tasks, especially in the field of image segmentation, such as VMamba [13], VM-UNet [14], VM-UNetV2 [15], CM-UNet [16], etc. They have proven that high-precision image segmentation can be achieved by stacking improved Mamba blocks combined with UNet [17,18]. However, Mamba uses forget gates [19] to provide positional information and local bias, forcing the model to adopt a recursive approach for training and inference. This recursive computation inevitably reduces the model's throughput. Therefore, using a deep dual-branch backbone for feature extraction combined with Mamba-2 [11] for global context information extraction has become a simple and practical strategy.

To enhance the scene-parsing capabilities of USVs operating in complex waterway environments under various weather conditions and reduce the misidentification rate of edges, this paper proposes a Lightweight Dual-branch Mamba Network based on image processing. Unlike the blocky U-Net architecture [14–16,20,21] and encoder–decoder frameworks [22–25], we are the first to attempt the use of a deep dual-branch backbone combined with Mamba-2 [11] to efficiently obtain the final feature map. Additionally, we have identified that directly fusing feature maps with high-resolution discrepancies can lead to imbalances and mismatches in features. To address this, we designed a Convolution-based Recombine Attention Module to apply weighted attention to both spatial and channel dimensions of the features, refining edge information. In consideration of the spatial information loss and directional sensitivity when images are unfolded into one-dimensional sequences, a Hilbert curve scanning mechanism is proposed to achieve multi-scale feature serialization. By stacking the feature sequences, this approach mitigates the local bias of Mamba-2 for non-causal data [19], thereby enabling more accurate global modeling and water edge segmentation.

The main contributions are summarized as follows:

- We have improved the Deep Dual-branch Network by adding an Atrous branch to increase the receptive field and enhance the information complementarity between the detail branch and the context branch. Furthermore, we are the first to integrate the deep dual-branch backbone with Mamba-2, thereby enabling Mamba-2 to more effectively acquire global contextual information.
- We designed a Recombine Attention Module based on convolution, which serves as a gate activation condition for Mamba-2 to enhance feature interaction and global-local information fusion from both spatial and channel dimensions.
- We proposed a Hilbert curve-scanning mechanism to realize the serialization of multi-scale features, and through feature sequence stacking, we alleviated the local bias of Mamba-2 toward non-causal data.
- The experimental results of four datasets show that the LDMNet network demonstrated strong applicability and edge prediction capabilities in both waterway environments and urban road environments.

## 2. Relative Work

Accurate and efficient obstacle segmentation in waterway environments is crucial for the safe navigation of Unmanned Surface Vehicles (USVs). In this section, we primarily introduce two types of work: efficient network architectures and attention fusion strategies.

### 2.1. Efficient Network Architectures

Efficient neural network architectures play a significant role in handling large amounts of data and complex tasks, addressing the challenge of maintaining model performance with limited computational resources. In mobile devices and embedded systems, lightweight network structures such as [26–28] improve computational efficiency by altering convolution structures and reducing complexity. However, this often comes at the expense of greater precision, and their effectiveness may not be as pronounced in more complex datasets. In recent years, the architectures represented by [9,29,30], which feature encoder–decoder structures, have emerged. These designs reduce spatial dimensions in the encoder and recover details in the decoder, with their symmetrical architecture aiding in the precise localization and segmentation of target objects. Nevertheless, such network architectures tend to suffer from more information loss, especially during the encoding process and particularly for high-resolution images. Overcoming this loss of information is a significant challenge in current research. Concurrently, multi-branch network architectures, represented by [7,8,31], have also seen rapid development in various fields. They typically excel in parallel processing capabilities and offer flexible methods of information fusion. By incorporating advanced design principles and technologies, they can maintain high performance while reducing the consumption of computational resources, making them suitable for a variety of complex visual tasks. Unlike [8], which employs frequent bilateral feature fusion, LDMNet achieves efficient complementary feature information through special cross-stage fusion, making it an efficient backbone designed specifically for dense prediction tasks.

### 2.2. Attention Fusion Strategies

Attention fusion strategies have been widely adopted in fields such as computer vision, natural language processing, and speech recognition. In the realm of computer vision, early fusion methods primarily involved Concatenation and Sum, which were widely used due to their fast computation speed and high interpretability. Subsequently, in order to capture both detailed information and high-level semantic information simultaneously, multi-scale feature fusion schemes [32,33] were developed based on these methods. These schemes innovated the fusion mechanism significantly and made the fusion structure more flexible. However, simply aggregating multi-level information does not guarantee effective information propagation. To achieve better feature fusion, works like [34–36] utilize learned attention weights for different branches to aggregate multi-level information. Nevertheless, traditional attention paradigms based on the Transformer structure are often characterized by high complexity and large computational requirements, which are not suitable for lightweight model intentions. In recent years, with the introduction of [37], several attention calculation methods specifically for images have begun to gain attention. Many outstanding works have focused on enhancing the feature representation of specific branches before fusion, such as [38,39], which emphasized exploring new feature enhancement methods with good portability using attention. However, this also leads to a fixed application scenario for the modules, making it difficult to tap into deeper performance capabilities. Other works focus on feature selection during fusion, such as [7,40,41], which carefully designs combinations of convolutions and nonlinear functions. However, these often concentrate on fully learning feature information while neglecting the astonishing capability of feature attention itself to enhance feature expression.

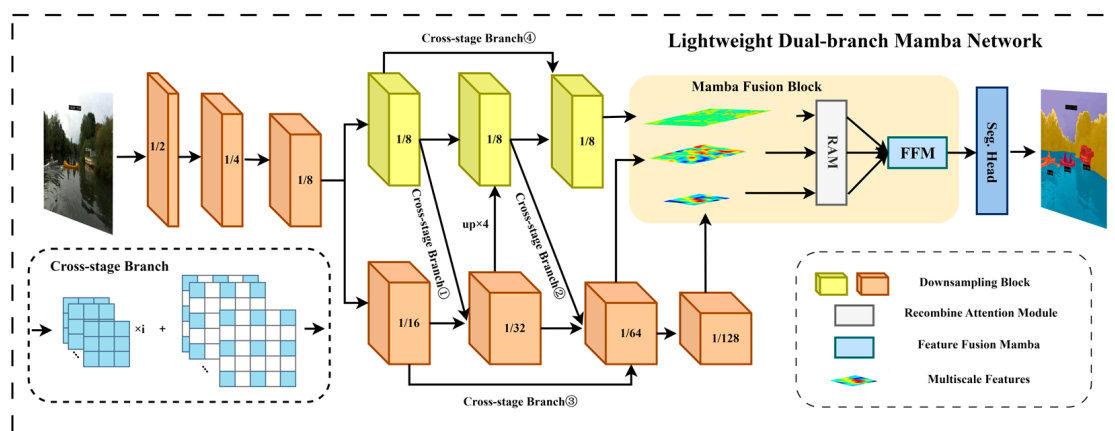
### 2.3. State Space Model

The State Space Model (SSM) is a mathematical model that describes the behavior of dynamic systems, characterized by its inherent cyclic linear representation [19]. Some works [42,43] have achieved satisfactory results based on SSM. However, Mamba [12] points out that SSM models are unable to perform context-based reasoning and proposes the use of the Selection Mechanism and Hardware-aware Algorithm to achieve efficient long-sequence modeling. Based on this, Bi-Mamba [44] introduces a bidirectional Mamba for

DNA sequence modeling, and Mamba-2 [11] transforms SSM from a single-head sequence to a multi-head sequence conversion. Subsequently, researchers have attempted to apply Mamba to visual tasks, such as Vision Mamba [45] introducing bidirectional SSM for global context modeling of non-causal data, VM-UNet series [14,15] combining Mamba with UNet to achieve precise segmentation of gastrointestinal and skin lesion semantic segmentation samples, and VMamba [13] proposing a 2D Selective Scan (SS2D) to bridge the gap between 1D array scanning and 2D plane traversal.

### 3. Methods

The overall network architecture of LDMNet is shown in Figure 1. The network backbone consists of an improved Deep Dual-branch Network, with the neck being the Mamba Fusion Block, which is composed of the Recombine Attention Module (RAM) and Feature Fusion Mamba (FFM). In this section, we will provide a detailed introduction to the entire pipeline of LDMNet.



**Figure 1.** The overall architecture of LDMNet, including the Mamba Fusion Block and the detailed structure of the cross-stage fusion branches. The notations such as  $\frac{1}{2}$  and  $\frac{1}{4}$  represent the downsampling multiples of the feature maps by the residual basic blocks. RAM denotes the Recombine Attention Module, FFM stands for Feature Fusion Mamba, and Seg. Head refers to the segmentation head. The detailed structure of the Cross-stage Branch is depicted within the dashed box.

#### 3.1. Deep Dual-Branch Network

We aim to achieve efficient feature extraction for specific scenes by leveraging multi-scale information and complementing multi-level information. To this end, we have redesigned the deep dual-branch backbone structure. Firstly, starting from the first downsampling, the network is divided into 7 stages (labeled as stages 0 to 6). After stage 2, an additional high-resolution branch is introduced. When the input image is  $1024 \times 1024$ , the high-resolution branch is fixed at  $128 \times 128$ , while the low-resolution branch extracts deep feature information through multiple downsamplings, ultimately resulting in an  $8 \times 8$  feature map. Secondly, to enhance the information exchange between the two branches, spatial and semantic information is shared once more among them through 4 Cross-stage Branches. The detailed structure is shown in Table 1.

Regarding the design philosophy of the network architecture, as illustrated in Figure 1, the approach can be broken down into specific steps. Firstly, to reduce computational complexity, a Downsampling Block is constructed by stacking two consecutive  $3 \times 3$  convolutional layers, and a bottleneck block is added at the end of each branch to expand the output dimensions. Secondly, the cross-stage fusion of the Atrous branch includes bilateral Cross-stage Branches ①~② (encompassing high-to-low and low-to-high fusion), the Cross-stage Branch ③ on the low-resolution branch (high-to-low fusion), and the Cross-stage Branch ④ on the high-resolution branch (channel fusion). For high-to-low fusion, there are two cases depending on the stage:

- (a) Between stage 3 and stage 4, bilateral feature fusion is conducted using a convolutional block consisting of a “ $3 \times 3$  convolution + BN + ReLU” sequence, followed by a downsampling operation with an Atrous convolution with a  $3 \times 3$  kernel and a dilation rate of 2. This is then followed by a  $1 \times 1$  convolution for channel reduction and finally fusion with the context branch.
- (b) Between stage 4 and stage 5, the convolutional block is doubled in sequence, with the rest of the settings remaining the same as in case a.

**Table 1.** Parameters of the Cross-stage Branch in LDMNet. It can be observed that we employ four different fusion methods, and each one concludes with a  $1 \times 1$  convolution for channel compression. To increase the receptive field, we incorporate Atrous convolutions in the first three types of cross-stage fusions.

Branch Name	Cross-Stage Branch in LDMNet Setting			
	Kernel (Size/Stride/Padding)	Channels	Dilation Rate	Repeating Times
Cross-stage Branch ①	$3 \times 3/2/1$	64	1	$x = 1$
	$3 \times 3/2/2$	256	2	$x = 1$
	$1 \times 1$	512	-	$x = 1$
Cross-stage Branch ②	$3 \times 3/2/1$	64/256	1	$x = 2$
	$3 \times 3/2/2$	512	2	$x = 1$
	$1 \times 1$	1024	-	$x = 1$
Cross-stage Branch ③	$3 \times 3/2/1$	128	s1	$x = 1$
	$3 \times 3/2/2$	512	2	$x = 1$
	$1 \times 1$	1024	-	$x = 1$
Cross-stage Branch ④	$3 \times 3/1/1$	64	1	$x = 1$
	$1 \times 1$	256	-	$x = 1$

For low-to-high fusion at the stage 4 position, the output features of the context branch are first compressed through  $1 \times 1$  convolution. They are then upsampled to  $1/8$  feature size using bilinear interpolation before being merged into the high-resolution branch. For channel fusion, only  $1 \times 1$  convolution is used to compress the feature channels.

If  $X_{Hi}$  and  $X_{Li}$  represent the high-resolution and low-resolution feature maps of the  $i$ -th stage, respectively, then the cross-stage fusion branches of the 4th and 5th stages can be represented as follows:

$$\begin{cases} X_{H4} = T_{L-H}F_LX_{L4} + R(F_HX_{H3}) \\ X_{H5} = T_{H-H}F_HX_{H3} + R(F_HX_{H4}) \\ X_{L4} = T_{H-L}F_HX_{H3} + R(F_LX_{L3}) \\ X_{L5} = T_{L-L}F_LX_{L3} + T_{H-L}F_HX_{H4} + R(F_LX_{L4}) \end{cases} \quad (1)$$

where  $F_H$  and  $F_L$  correspond to sequences of residual basic blocks with high and low resolutions, respectively, and  $T_{L-H}$ ,  $T_{H-L}$ ,  $T_{H-H}$ , and  $T_{L-L}$  represent the transformation functions from low to high, from high to low, from high to high, and from low to low, respectively.  $R(\cdot)$  denotes the ReLU function. Finally, the context branch has one additional stage 6 compared to the high-resolution branch, which is used to downsample the feature map to  $1/128$  of the original feature map. The Mamba Fusion Block then performs further multi-scale fusion using the stage 5 feature map from the high-resolution branch and the stage 5 and stage 6 feature maps from the context branch.

### 3.2. Mamba Fusion Block

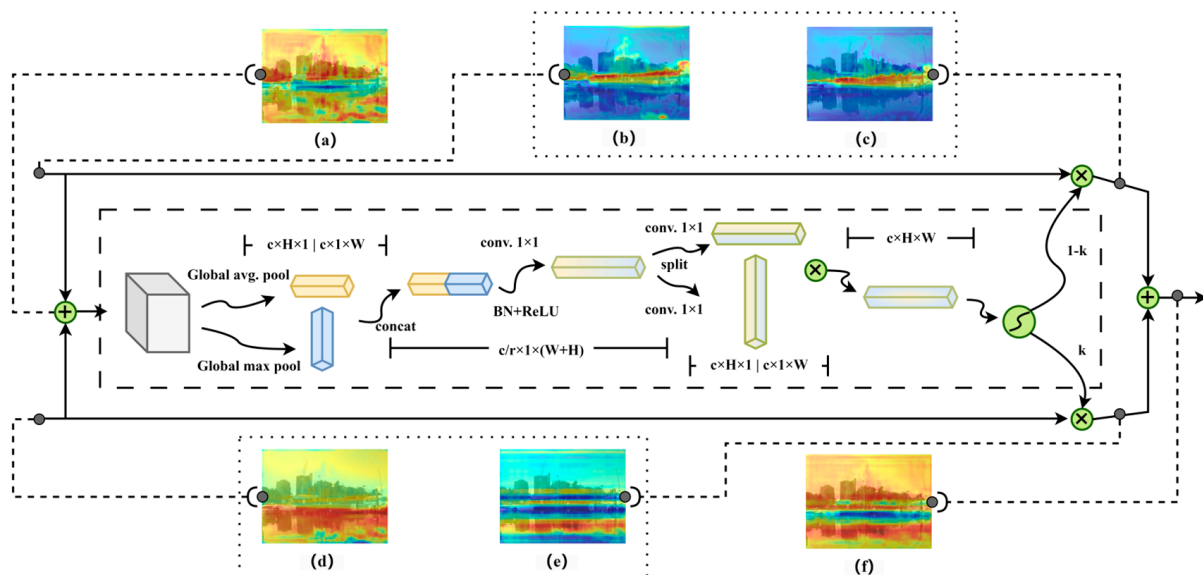
The Mamba Fusion Block consists of the Recombine Attention Module (RAM) and the Feature Fusion Mamba (FFM). The RAM can be regarded as a computational unit designed to enhance the expressive power of features while learning the most suitable fusion strategy between branches. The FFM is aimed at helping the model delve deeper

into contextual information. Below, we will introduce the detailed structures and functions of each component.

### 3.2.1. Recombine Attention Module

Inspired by convolutional attention mechanisms such as Squeeze-and-Excitation Attention [37] and Coordinate Attention Blocks [39], we propose a Recombine Attention Module (RAM) to address the issues of information conflict and inconsistency that arise during the fusion of feature maps with high-resolution differences. This module enables each pixel to focus on more specific contextual information from multi-level features during the aggregation phase, as illustrated in Figure 2. To explain in detail, given two input feature maps with different resolutions, corresponding to Figure 2b,d, let us assume  $Y$  is the feature map from a deeper level. In our network, this can be specifically represented as:

- Cross-branch connection scenario:  $X$  is the detail feature map output from stage 5 of the high-resolution branch, and  $Y$  is the context feature map output from stage 5 of the low-resolution branch.
- Cross-stage connection scenario:  $X$  is the output feature map from the cross-branch connection, and  $Y$  is the deep feature map output from stage 6 of the low-resolution branch.



**Figure 2.** Schematic diagram of the Recombine Attention Module (RAM) structure. In the diagram, the spatial dimensions of the feature maps at each stage are indicated. The images within the dashed boxes represent intermediate process feature maps, which are connected to the corresponding positions in the RAM with dashed lines. The two input branches on the left represent the input feature maps with high-resolution differences, corresponding to feature maps (b,d). After processing, the feature maps of each branch can be compared with (b) and (d) by (c) and (e), respectively. The feature maps after passing through the RAM can be compared with (a,f).

To enhance the RAM's overall grasp of spatial and texture information, as well as its sensitivity to the spatial location of information, the input features  $X$  and  $Y$  are first fused to obtain feature (a), which is then input into a carefully designed attention calculation unit to acquire attention weights, as shown in the dashed box in Figure 2. Specifically, the fused feature map (a) of  $X + Y$  is input into global average pooling and global max pooling layers in parallel, obtaining a pair of location-aware feature encodings along the  $H$  and  $W$  dimensions. At this stage, the transformation output of the global average pooling at height  $H$  in channel  $c$  can be represented as:

$$z_c^h(h) = \frac{1}{W} \sum_{0 \leq i \leq W} f_c^{X+Y}(h, i). \quad (2)$$



where  $z_c^h \in \mathbb{R}^{c \times H \times 1}$  and  $f_c^{X+Y}$  represent the fused feature map of  $X + Y$  in channel  $c$ . Similarly, if we let  $Max(\cdot)$  denote the global max pooling transformation function, then the transformation output of the global max pooling at width  $W$  in channel  $c$  can be represented as:

$$z_c^w(w) = Max(f_c^{X+Y}(h, w)). \quad (3)$$

Next, the feature sequences obtained from the aforementioned transformations are projected into the same spatial dimension for concatenation. The concatenated feature map is then processed by a combined transformation function  $F$ , which consists of “ $1 \times 1$  convolution + BN + ReLU”, to encode the spatial information of the feature map in both the horizontal and vertical directions, resulting in:

$$f = F([z^h, z^w]) \quad (4)$$

where  $f \in \mathbb{R}^{c/r \times (H+W) \times 1}$  represents the intermediate feature map generated and  $r$  denotes the channel reduction ratio, which is used to control the number of parameters in the computation process. The notation  $[\cdot, \cdot]$  represents the concatenation operation.

Finally, the  $f$  is split into two tensors, and each is processed through a  $1 \times 1$  convolution transformation. The width and height of the two convolutions are restored to match those of  $z^w$  and  $z^h$ , respectively. The outputs of the two convolutions are then multiplied and transformed to match the spatial dimensions of the input  $X$ , thus producing:

$$g = \sigma(F_h(f^h) \times F_w(f^w)) \quad (5)$$

where  $f^h$  and  $f^w$  represent the two tensors that are split,  $F_h$  and  $F_w$  represent the  $1 \times 1$  convolution transformation functions, and  $\sigma$  is the sigmoid function.

In this way, the weights learned by the attention calculation unit will further refine and modulate the (b) branch and the (d) branch, ultimately achieving the most suitable fusion state. As shown in the dashed box in Figure 2, if we view the above calculation unit as a transformation function  $M$ , then the overall operation scenario can be summarized as:

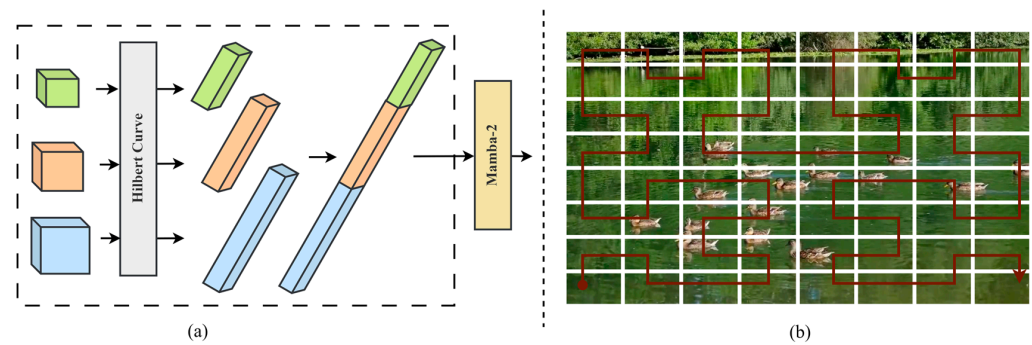
$$M(X + Y) \times X + (1 - M(X + Y)) \times Y \quad (6)$$

### 3.2.2. Feature Fusion Mamba

In image segmentation tasks, contextual information plays a crucial role in helping models resolve ambiguities and improve edge detection. To effectively capture contextual information, it is necessary to have a comprehensive grasp of features enhanced by attention at various scales. As shown in Figure 3, the data processing in the Feature Fusion Mamba (FFM) involves two steps: Hilbert curve scanning and Mamba-2 [11] processing. Firstly, the features at various scales are encoded in parallel using the Hilbert curve, which expands them into a one-dimensional sequence based on coordinate encoding. Then, the one-dimensional sequences from each scale are stacked to form a new sequence. Finally, this new sequence is input into Mamba-2 for selective scanning.

During the process of transforming two-dimensional features into a one-dimensional sequence, simple row-first or column-first interleaved scanning [13] is prone to causing information loss, especially for spatial proximity information. The Feature Fusion Mamba (FFM) adopts the Hilbert curve as the traversal path, which can maintain a certain degree of spatial proximity. Moreover, the Hilbert curve can reduce the distortion of information caused by serialization, allowing Mamba-2 to better simulate the way the human visual system pays attention to high-dimensional data when scanning the one-dimensional sequence. This will help the model to better capture contextual information.

Additionally, the non-causal nature of image data constrains the effective reasoning of the Mamba series framework [13,14,19]. The FFM achieves single-pass multiple selective scanning by stacking multi-scale features, effectively integrating information at different scales and establishing a global receptive field in two-dimensional space.



**Figure 3.** Illustrates the operation of the Feature Fusion Mamba (FFM). In (a), the detailed structure of the FFM is shown, where Hilbert Curve represents the use of the Hilbert curve as the traversal path for serializing the spatial arrangement of patch sequences. (b) The Hilbert curve traversal path.

#### 4. Experiment

In this section, we begin by introducing the training details, including the dataset, training setup, hardware, and evaluation metrics. We then assess the performance of LDMNet by comparing it with benchmark networks on four datasets: MaSTr1325 [46], LaRS [47], Water Segmentation in the USVInland [48], and Cityscapes [49], as shown in Table 2. Finally, we conduct ablation experiments on LDMNet from two perspectives, the Mamba Fusion Block and the overall architecture, and analyze and discuss the results.

**Table 2.** Important parameters of the datasets used.

Dataset	Type	Train	Val.	Resolution	Class
MaSTr1325	Marine Obstacle Segmentation	1060	265	$512 \times 512$	3
LaRS	Marine Obstacle Segmentation	2605	198	$1024 \times 1024$	3
Water Segmentation	Distinguishing between Reflections and the Actual Division of Water and Land	1166	234	$640 \times 640$	2
Cityscapes	Semantic Segmentation of Cityscapes	2975	500	$1024 \times 1024$	19

##### 4.1. Dataset

###### 4.1.1. Waterway Environment Dataset

**MaSTr1325:** MaSTr1325 [46] is a large-scale marine semantic segmentation dataset specifically designed for developing obstacle detection methods for small Unmanned Surface Vehicles (USVs). It contains 1325 different images captured by real USVs and annotated with pixel-level semantic labels. The labels are categorized into three classes: ocean, sky, and environment. To facilitate the training and comparison of various models, the dataset is divided into training and validation sets in a ratio of 8:2.

**LaRS:** LaRS [47] is currently the most diverse dataset for marine obstacle detection, capable of performing both semantic segmentation and panoramic segmentation tasks. All images are annotated with 20 scene-level attributes and categorized into three object classes and eight obstacle (dynamic obstacle) classes. The semantic segmentation dataset includes 2605 training images, 1203 test images, and 198 validation images.

**Water Segmentation in the USVInland:** Water Segmentation is a subset of the USVInland [48] series dataset, used to distinguish reflections from real objects. The dataset consists of 364 high-resolution images ( $1280 \times 640$ ) and 1036 low-resolution images ( $640 \times 320$ ). We have reorganized the dataset by incorporating some validation set images into the training set, resulting in an ultimate training set of 1166 images and a validation set of 234 images.

###### 4.1.2. Classic Datasets for Autonomous Driving

**Cityscapes:** Cityscapes [49] is a classic dataset in the field of autonomous driving, focusing on the semantic understanding of urban street scenes. It contains 5000 finely annotated images, with 2975 of them being used for training, 500 for validation, and 1525



for testing. The images in the dataset have a resolution of  $2048 \times 1024$  and are categorized into 19 classes. During training, no additional 20,000 roughly labeled images are used.

#### 4.2. Train Setting

All experiments are based on the MMSegmentation [50] framework. The specific training setup is as follows: using the Poly strategy to update the learning rate, AdamW optimizer, initial learning rate of 0.00006, exponential decay rates for the first and second moments of the estimate set to 0.9 and 0.999, respectively, and weight decay set to 0.01. For LaRS and Cityscapes, images are randomly cropped to  $1024 \times 1024$ , while for MaStr1325 and Water Segmentation in the USVInland, they are randomly cropped to  $512 \times 512$  and  $640 \times 640$ , respectively. During training, data augmentation operations are added, including random cropping of images, random horizontal flipping, and random scaling within the range of 0.5 to 2.0. An online hard example mining pixel sampler (OHEM Pixel Sampler) [51] is used, and during training, pixel values with confidence scores below 0.7 are sampled, with at least 100,000 pixel values retained. OHEM cross-entropy loss [51] is also used. To accelerate training, gradient accumulation is performed, with parameters updated four times before each update. To ensure thorough learning, all models are trained for 300 K iterations, with validation performed every 30 K iterations. The batch size is 2, and the training is conducted on a single 3070Ti GPU.

#### 4.3. Evaluation Metrics

The most commonly used evaluation metrics in the semantic segmentation field are adopted, which are Mean Intersection over Union (mIoU), Mean Pixel Accuracy (mPA), Mean Dice Coefficient (mDice), Recall (Re), F1 Score (F1), Frames Per Second (FPS), and in terms of model complexity, Giga Floating-point Operations Per Second (GFLOPs) and Params. mIoU measures the average of the ratio of the intersection and union of the predicted mask and the ground truth mask for all types, mPA is the average pixel accuracy for all types, and mDice is used to measure the average overlap between the predicted segmentation and the true segmentation for each category. Re represents recall, F1 represents the F1-Score, FPS is the number of frames processed per second by the model, GFLOPs is the number of floating-point operations per second, and Params is the number of parameters produced by the model.

#### 4.4. Speed and Accuracy Comparisons

During inference, the batch size is set to 1, using cuda 11.1, CUDNN 8004, and Pytorch 1.9.1 + cu111. Through fair comparison with other benchmarks on datasets such as MaStr1325, LaRS, Water Segmentation in the USVInland, and Cityscapes, the results are presented in Tables 3–6.

**Table 3.** Accuracy comparison of LDMNet with other advanced methods on MaStr1325. Among them, "-" indicates a lack of relevant data.

Model	Type	GPU	Resolution	Params ↓	mIoU (%) ↑	Speed (FPS) ↑
Deeplab V3+ [22]	CNN	V100	$512 \times 512$	-	85.4	0.56
SegNet [23]	CNN	V100	$512 \times 512$	-	81.8	0.85
WODIS [52]	CNN	V100	$512 \times 384$	89.5 M	91.3	43.2
Fast SCNN [25]	CNN	3070Ti	$512 \times 512$	1.36 M	93.5	67.5
DDRNet-s [8]	CNN	3070Ti	$512 \times 512$	17.05 M	94.5	79
Segmenter(vit-s) [24]	Transformer	3070Ti	$512 \times 512$	-	94.8	53.4
TransNeXt-t [53]	Transformer	4090Ti	$512 \times 512$	28.2 M	95.4	10.3
UNetformer(R18) [20]	Transformer	3070Ti	$512 \times 512$	11.69 M	94.2	5.6
VMamba-s [13]	Mamba	3070Ti	$512 \times 512$	70 M	93.6	52
VM-UNet [14]	Mamba	3070Ti	$512 \times 512$	34.62 M	93.4	21.1
VM-UNetV2 [15]	Mamba	3070Ti	$512 \times 512$	17.91 M	94.8	32
CM-UNet [16]	Mamba	3070Ti	$512 \times 512$	12.89 M	93.7	8.5
LDMNet	CNN&Mamba	3070Ti	$512 \times 512$	12.53 M	96.2	80

**Table 4.** Accuracy comparison of LDMNet with other advanced methods on LaRS.

Model	Type	mIoU (%) ↑	Re (%) ↑	F1 ↑
ICNet [54]	CNN	93.3	49.7	44.9
STDC2 [55]	CNN	93.5	54.3	54.3
PiDNet-s [7]	CNN	94.1	61.8	52.2
SFNet [56]	CNN	95.2	62.4	58.1
Segmenter [24]	Transformer	95.1	59.5	55.2
MLLA [19]	Transformer	95.3	63.5	59.4
UNetformer [20]	Transformer	93.5	61.5	54.3
TransNeXt-t [53]	Transformer	94.9	61.3	53.1
RSMamba [57]	Mamba	94.2	60.1	52.9
VM-UNet [14]	Mamba	95.1	64.2	61.3
CM-UNet [16]	Mamba	95.4	65.4	62.8
VMamba-s [13]	Mamba	94.2	60.5	54.2
LDMNet	CNN&Mamba	95.6	78.6	75.2

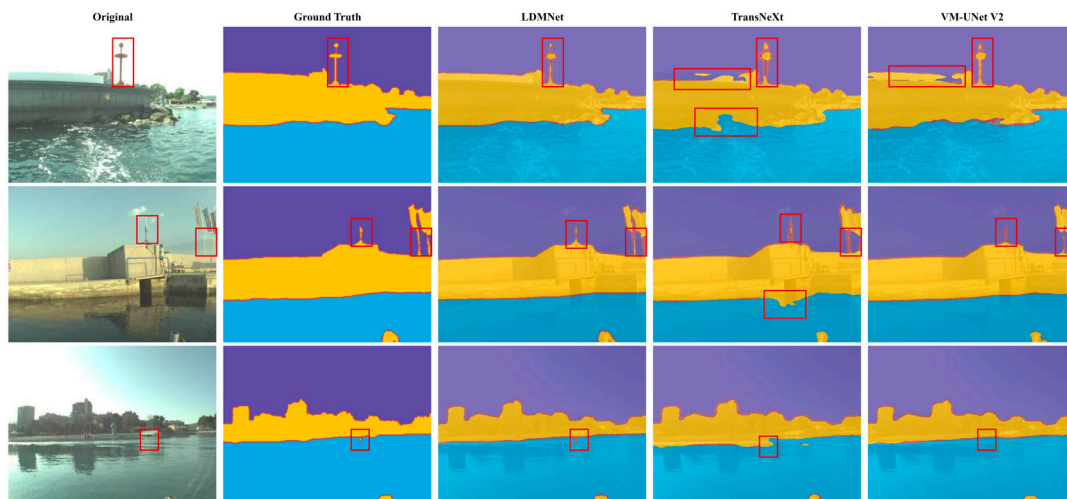
**Table 5.** Accuracy comparison of LDMNet with other advanced methods on Water Segmentation in the USVInland.

Model	Resolution	mIoU (%) ↑	mDice (%) ↑
BisenetV2 [58]	640 × 640	97.68	98.46
DDRNet-s [8]	640 × 640	98.64	99.15
Segmenter(vit-s) [24]	640 × 640	98.80	99.23
LDMNet	640 × 640	99.02	99.51

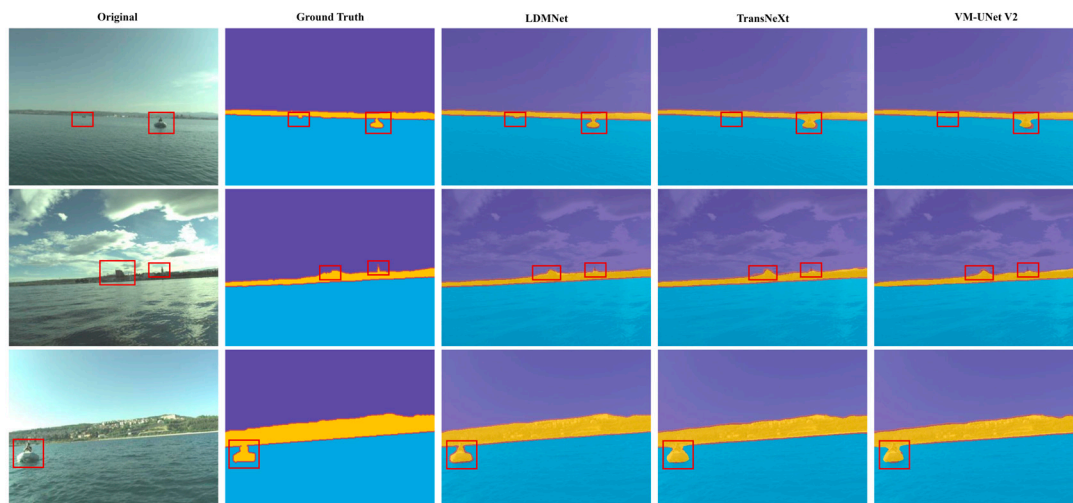
**Table 6.** Accuracy comparison of LDMNet with other advanced methods on Cityscapes. Among them, "-" indicates a lack of relevant data.

Model	Resolution	mIoU (%) ↑	GFLOPs ↓
SFNet(DF1) [56]	2048 × 1024	74.5	24.7
STDC2-Seg75 [55]	1536 × 768	76.8	-
PP-LiteSeg-T2 [9]	1536 × 768	74.9	-
HyperSeg-M [30]	1024 × 512	75.8	7.5
PiDNet-S [7]	2048 × 1024	77.1	46.3
DDRNet-s [8]	2048 × 1024	77.2	34.2
LDMNet	2048 × 1024	80.7	32.9

MaStr1325: From Table 3, it can be observed that LDMNet achieves a good compromise between accuracy and real-time performance. On the MaStr1325 test set, LDMNet reaches a mIoU of 96.2% at a speed of 80 FPS under the aforementioned hardware environment, which is 1.7% higher than DDRNet-s with a similar speed. Other methods, such as Segmenter and Fast SCNN, are outperformed by LDMNet on the test set with mIoU improvements of 1.4% and 2.7%, respectively, and LDMNet also operates faster than both of them. WODIS, proposed specifically for the detection of surface obstacles on maritime autonomous surface vehicles, achieves an mIoU of 91.3% at 43.2 FPS on the MaStr1325 test set, but this result is 4.9% lower than our proposed method and 36.8% slower in terms of speed. In comparison with many models based on the Transformer architecture and Mamba architecture, LDMNet outperforms TransNeXt-t by 0.8 mIoU and improves over other methods by 1.4% to 2.8% mIoU. We visualize the inference results of LDMNet and other methods on MaStr1325 in Figures 4 and 5. By comparing the segmentation effects of near and distant targets, it is evident that LDMNet can handle the edges of target objects more finely and demonstrates its strong capability in capturing details for elongated and small target edges.



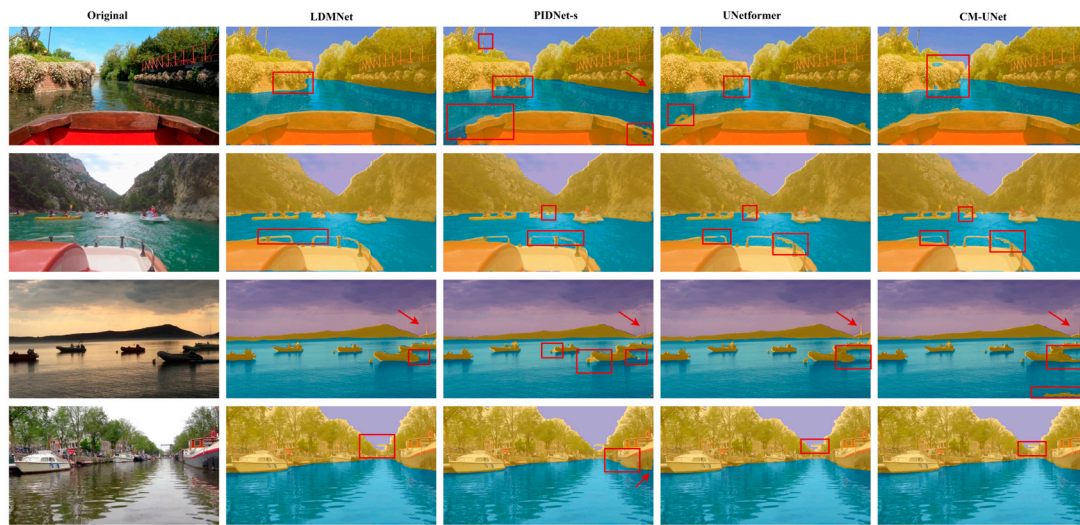
**Figure 4.** Visualization of the inference of LDMNet compared to TransNeXt [53] and VM-UNet V2 [15] on the MaStr1325 for close-up targets. We have marked the detailed differences with red boxes.



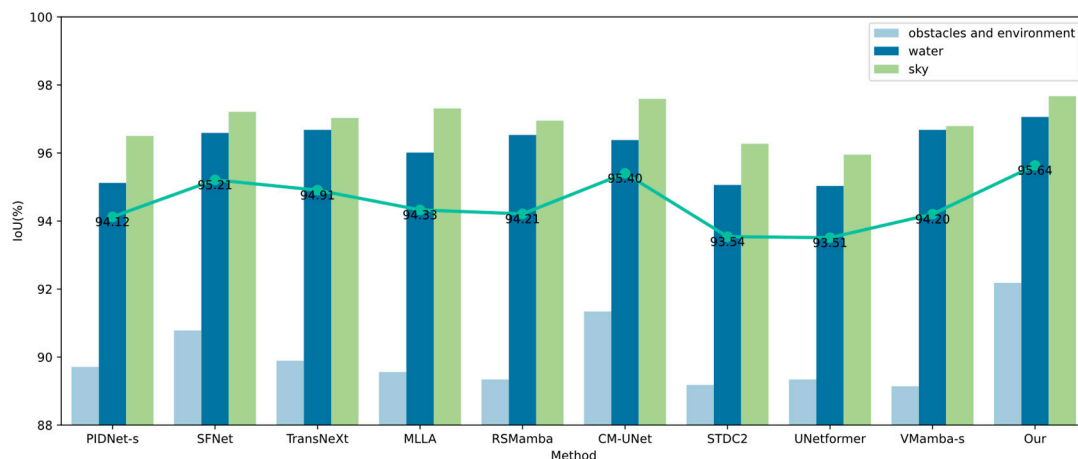
**Figure 5.** Visualization of the inference of LDMNet compared to TransNeXt and VM-UNet V2 on the MaStr1325 for distant targets. We have marked the detailed differences with red boxes.

**LaRS:** As shown in Table 4, in order to fully utilize the information retention capability of the Deep Dual-branch Network for high-resolution images, the images on the LaRS are cropped to  $1024 \times 1024$ . The experimental results show that networks based on the Mamba architecture generally perform well, with the highest mIoU achieved by CMU-Net at 95.4%, and other methods also exceed 94.2% mIoU, demonstrating the advanced nature of Mamba in handling long-sequence data. It is worth noting that CMU-Net also uses CNN attention modules in conjunction with Mamba blocks, which confirms the compatibility and effectiveness of CNN and Mamba in long-sequence modeling. The MLLA based on Transformer also achieves a score of 95.3%, leveraging the advantages of the Mamba architecture to improve the shortcomings of linear attention, showing great potential, but its accuracy is still 0.3% mIoU lower than LDMNet. Methods solely based on CNN networks, except for SFNet, do not show competitiveness, which is related to the lack of global information modeling capability in CNN networks, which is also one of the reasons we tried to integrate Mamba into global modeling. Figure 6 demonstrates LDMNet's strong capability in detail and edge processing. We also analyzed the accuracy of various prediction results of LDMNet and baseline methods on the LaRS, as shown in Figure 7. It can be seen that the segmentation accuracy of all methods tends to stabilize for the “water”

and “sky” categories, while the segmentation accuracy of the “obstacles and environment” category determines whether the overall segmentation accuracy can be improved.



**Figure 6.** Visualization of the inference of PIDNet-s, UNetformer, CM-UNet, and LDMNet on the LaRS. We have marked the detailed differences with red boxes and red arrows.



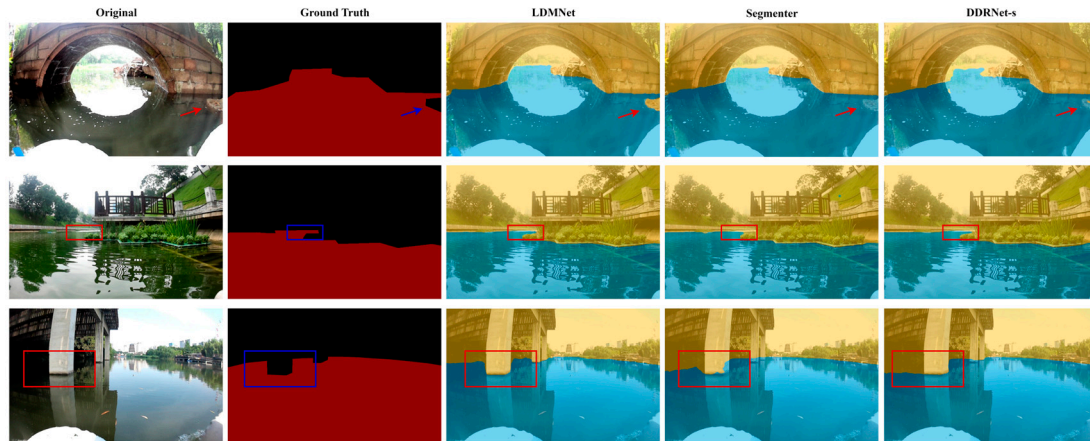
**Figure 7.** Comparison of the IoU of different methods in predicting various categories on the LaRS. Among them, the green solid line with circles indicates the mIoU difference between different models on LaRS.

**Water Segmentation in the USVInland:** To test LDMNet’s performance in resisting light reflection on the water’s surface, we trained LDMNet on the Water Segmentation in the USVInland and compared it with BiseNetV2, DDRNet-s, and Segmenter. The results are shown in Table 5. LDMNet significantly outperforms the benchmarks, with a mDice of 99.51% and a mIoU of 99.02%. Compared to Segmenter (vit-s) and DDRNet-s, the mDice is 0.28% and 0.36% higher, respectively, and the mIoU is 0.22% and 0.38% higher, respectively. We visualized the inference results of LDMNet and the benchmark methods BiseNetV2, DDRNet-s, and Segmenter in Figure 8. It is evident that LDMNet pays more attention to details in the treatment of boundaries compared to other methods, is more sensitive to edge features, and demonstrates strong fitting ability.

**Citiescapes:** To verify LDMNet’s applicability on different types of datasets, we applied LDMNet to Citiescapes in the autonomous driving field for validation. The experimental results are shown in Table 6. LDMNet still outperforms the methods listed in the table, achieving real-time detection with an mIoU of 80.7% on the Citiescapes validation set, outperforming SFNet and PP-LiteSeg-T2 by more than 5.8%. The speed is close to



STDC2-Seg75 and HyperSeg-M, but the mIoU is 3.9% and 4.9% higher, respectively. We also compared it with the current most advanced multi-branch network PIDNet-s. Although the inference speed is slightly behind, LDMNet's mIoU is 3.6% higher than PIDNet-s, indicating the strong applicability of the Dual-branch Network across different types of datasets and its potential to excel in the field of water and land unmanned driving.



**Figure 8.** Visualizes the inference of LDMNet compared to BisenetV2 and DDRNet-s on the Water Segmentation in the USVInland for water–shore segmentation. We have marked the detailed differences with boxes and arrows. It can be seen that in complex environments where the reflection of the water surface is overcome, LDMNet still demonstrates high accuracy in demarcating the boundaries between the water and the shore.

#### 4.5. Comparison of Performance with Advanced Counterparts in Similar Modules

To compare the performance of our designed module with advanced counterparts in similar modules, we conducted a comparative experiment. In this experiment, the backbone was replaced with DDRNet-s and LDMNet, and different Mamba scanning mechanisms were tried at the Neck position, each paired with RAM and AFF in a cross combination. The experimental results are presented in Tables 7 and 8.

**Table 7.** Comparison of the number of parameters between RAM and AFF [28], showing that RAM has fewer parameters than AFF.

Method	Params
AFF [28]	39.71 k
RAM	39.58 k

Regarding the Recombine Attention Module, in LDMNet, we introduce the Recombine Attention Module (RAM) to address the issues of feature imbalance and mismatch that arise during the fusion of branches with high-resolution differences. As indicated by the data in Tables 7 and 8, when all other module configurations remain the same, substituting the fusion strategy with RAM yields an increase in mIoU of 0.69% to 1.34% and in mPA of 0.38% to 0.66%, with a reduction in the number of parameters of 0.13 k. Specifically, when using the DDRNet-s-Mamba-AFF combination as the baseline, under the same experimental conditions, replacing AFF with RAM results in an increase in mIoU of 1.34% and mPA of 0.59%; if the backbone remains unchanged while both the Mamba scanning mechanism and RAM are replaced, mIoU and mPA increase by 1.43% and 0.71%, respectively; finally, when the Backbone is replaced with LDMNet, comparing LDMNet-Mamba-RAM with DDRNet-s-Mamba-AFF, mIoU and mPA increase by 1.63% and 0.95%, respectively. This demonstrates that RAM is capable of better adaptive selection and retention of more image features during feature fusion, and it exhibits greater applicability in tasks that require important detailed information such as edge textures.



**Table 8.** Comparison of the enhancement effects of RAM and AFF [28] on different Backbones on MaSTr1325, where SS represents the Sequential Scan Module, SS2D [53] denotes the 2D Selective Scan Module, and OSSM [57] signifies the Omnidirectional Selective Scan Module.  $\delta$  is the change obtained by subtracting the corresponding items of AFF from RAM, with DDRNet + Mamba (SS) as the comparison object. "-" indicates that the current row is used as the baseline.

Method		mIoU (%) $\uparrow$	mPA (%) $\uparrow$	$\delta$	
DDRNet + Mamba				$\delta$ (mIoU) (%) $\uparrow$	$\delta$ (mPA) (%) $\uparrow$
+SS					
+SS2D [53]	+AFF	92.53	94.97	-	-
	+RAM	93.87 (+1.34)	95.56 (+0.59)	+1.34	+0.59
	+AFF	93.09	94.68	-0.56	-0.29
	+RAM	94.23 (+1.14)	95.34 (+0.66)	+0.36	-0.22
+OSSM [57]					
+HCS	+AFF	93.04	94.37	+0.51	-0.60
	+RAM	94.42 (+1.38)	95.48 (+1.11)	+0.55	-0.08
	+AFF	94.68	97.02	+2.15	+2.05
	+RAM	95.96 (+1.28)	97.68 (+0.66)	+2.09	+2.12
<hr/>					
LDMNet + Mamba					
+SS					
+SS2D [53]	+AFF	93.43	95.42	+0.90	+0.45
	+RAM	94.12 (+0.69)	95.80 (+0.38)	+0.25	+0.24
	+AFF	94.17	95.94	+1.64	+0.97
	+RAM	95.34 (+1.17)	96.84 (+0.90)	+1.47	+1.28
+OSSM [57]					
+HCS	+AFF	94.47	96.02	+1.94	+1.05
	+RAM	95.06 (+0.59)	96.51 (+0.49)	+1.19	+0.95
	+AFF	95.32	97.38	+2.79	+2.41
	+RAM	96.16 (+0.84)	97.92 (+0.54)	+2.29	+2.36

Regarding the Hilbert Curve Scan, we designed four sets of experiments in conjunction with different backbones, as shown in Table 8. SS denotes sequential scanning, SS2D [53] represents the 2D Selective Scan Module, OSSM [57] represents the Omnidirectional Selective Scan Module, and HCS denotes the Hilbert Curve Scan. The results indicate that using SS as the baseline, SS2D improves mIoU by 0.36% and 1.22% through sequential scanning from four directions of the image, while OSSM, using eight-directional sequential scanning, increases mIoU by 0.55% and 0.94%. It is evident that merely increasing the number of directions not only boosts computational load but also does not necessarily lead to enhanced precision. Our proposed method of processing multi-scale images through HCS and stacking them results in an improvement in mIoU of 2.09% and 2.04%. Therefore, to enhance the model's effective modeling of non-causal data, it is crucial to select an appropriate scanning method. Our scanning method yielded more convincing results when combined with the Deep Dual-branch Network and Mamba.

#### 4.6. Ablative Experiments on MaSTr1325

We fixed the Backbone as LDMNet and reorganized the modules at the Neck position for retraining. The Recombine Attention Module (RAM) focuses on better fusion of feature maps with high differences, while the combination of Mamba and Hilbert Curve Scan (HCS) enables effective extraction of contextual information from deep features. The experimental results presented in Table 9 show that incrementally adding RAM and Mamba-2 (with HCS) can improve the model's mIoU by 0.87% and 1.22%, respectively. Moreover, using HCS with Mamba can enhance the model's mIoU by an additional 0.26%. Simultaneously

incorporating all three modules results in an improvement of 1.6% in mIoU and 1.15% in mPA, validating the effectiveness of these modules.

**Table 9.** Ablation experiments of LDMNet on MaSTr1325. Among them, "✓" indicates that this module is retained, while "✗" indicates the opposite.

+RAM	+HCS	+Mamba-2	mIoU (%) ↑	mPA (%) ↑
✗	✗	✗	94.56	96.77
✓	✗	✗	95.43	97.04
✗	✗	✓	95.52	97.16
✗	✓	✓	95.78	97.56
✓	✗	✓	96.09	97.80
✓	✓	✓	96.16	97.92

#### 4.7. Discussion

Our experimental evaluations on MaSTr1325, LaRS, Water Segmentation in the USVInland, and Cityscapes have demonstrated the significant advantages of LDMNet in terms of its sensitivity to object edges and its robustness in utilizing global information, highlighting the practicality of LDMNet in real-world scenarios. Furthermore, we have compared our proposed Reorganized Attention Module (RAM) and Hilbert Curve Scanning (HCS) with advanced similar modules, as shown in Tables 7 and 8, illustrating their superiority. However, despite the encouraging results, our method is not without limitations. For instance, the computational cost associated with HCS is still higher than that of traditional methods. Additionally, although LDMNet has been proven to perform well on various datasets, its performance may vary with different types of data, which warrants further investigation.

#### 4.8. Impact on Future Intelligent Marine Traffic Systems

In the field of intelligent marine traffic systems, the research and application of efficient networks for semantic segmentation tasks of Complex Waterway Scenes have provided effective and practical solutions. Our proposed LDMNet, with its meticulously designed network structure, achieves rapid and accurate identification of various objects and obstacles within Complex Waterway Scenes. This provides crucial environmental information for autonomous navigation and collision decision-making. As artificial intelligence algorithms are applied in the field of Unmanned Surface Vehicles, they can significantly accelerate the research, development, and application of USVs, enhancing the overall operational capabilities of the system. With the maturation and advancement of technology, there will be a drive to update relevant regulations and standards, ensuring navigation safety and promoting technological progress and standardization in the entire waterborne transportation industry.

## 5. Conclusions

This paper delves into the issue of inaccurate obstacle edge recognition in Complex Waterway Scenes for Unmanned Surface Vehicles (USVs), which is caused by factors such as lighting variations, surface fluctuations, and reflections. We propose a novel Lightweight Dual-branch Mamba Network named LDMNet, which is the first method to combine a Deep Dual-branch Network with Mamba-2 for semantic segmentation tasks in Complex Waterway Scenes. Through a series of rigorous experimental evaluations, we demonstrate the effectiveness of our approach in understanding Complex Waterway Scenes and achieve significant performance improvements compared to previous works. In particular, to address the spatial information loss and directional sensitivity issues that arise during image serialization, we introduce the Hilbert curve scanning mechanism to achieve multi-scale feature serialization. By stacking serialized feature maps, we alleviate the local bias of Mamba-2 for non-causal data. This research offers a new perspective on solutions for obstacle recognition in Complex Waterway Scenes. We believe that with the continuous

advancement of technology and further research, LDMNet will play an even greater role in the field of USV visual perception, driving development and innovation in related areas.

**Author Contributions:** Investigation, C.L.; methodology, T.D.; software, T.D. and S.H. (Shishuo Han); supervision, H.X.; validation, C.L. and S.H. (Shishuo Han); writing—original draft, T.D. and S.H. (Song Huang); writing—review and editing, G.H. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research received no external funding.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** Data are contained within the article.

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

## References

1. Steccanella, L.; Bloisi, D.; Castellini, A.; Farinelli, A. Waterline and obstacle detection in images from low-cost autonomous boats for environmental monitoring. *Robot. Auton. Syst.* **2020**, *124*, 921–8890. [\[CrossRef\]](#)
2. Sravanthi, R.; Sarma, A. Efficient image-based object detection for floating weed collection with low cost unmanned floating vehicles. *Soft Comput.* **2021**, *25*, 13093–13101. [\[CrossRef\]](#)
3. Bovcon, B.; Kristan, M. WaSR-A Water Segmentation and Refinement Maritime Obstacle Detection Network. *IEEE Trans. Cybern.* **2022**, *52*, 12661–12674. [\[CrossRef\]](#)
4. Teršek, M.; Žust, L.; Kristan, M. ewasr-an embedded-compute-ready maritime obstacle detection network. *Sensors* **2023**, *23*, 5386. [\[CrossRef\]](#) [\[PubMed\]](#)
5. Yao, L.; Kanoulas, D.; Ji, Z.; Liu, Y. ShorelineNet: An Efficient Deep Learning Approach for Shoreline Semantic Segmentation for Unmanned Surface Vehicles. In Proceedings of the 2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Prague, Czech Republic, 27 September–1 October 2021; pp. 5403–5409.
6. Cai, Q.; Wang, Q.; Zhang, Y.; He, Z.; Zhang, Y. LWDNet-A lightweight water-obstacles detection network for unmanned surface vehicles. *Robot. Auton. Syst.* **2023**, *166*, 921–8890. [\[CrossRef\]](#)
7. Xu, J.; Xiong, Z.; Bhattacharyya, S. PIDNet: A Real-time Semantic Segmentation Network Inspired by PID Controllers. In Proceedings of the 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Vancouver, BC, Canada, 18–22 June 2023; pp. 19529–19539.
8. Pan, H.; Hong, Y.; Sun, W.; Jia, Y. Deep Dual-Resolution Networks for Real-Time and Accurate Semantic Segmentation of Traffic Scenes. *IEEE Trans. Intell. Transp. Syst.* **2023**, *24*, 3448–3460. [\[CrossRef\]](#)
9. Peng, J.; Liu, Y.; Tang, S.; Hao, Y.; Chu, L.; Chen, G.; Wu, Z.; Chen, Z.; Yu, Z.; Du, Y.; et al. PP-LiteSeg: A Superior Real-Time Semantic Segmentation Model. *arXiv* **2022**, arXiv:2204.02681.
10. Carrillo-Perez, B.; Rodriguez, A.; Barnes, S.; Stephan, M. Improving YOLOv8 with Scattering Transform and Attention for Maritime Awareness. In Proceedings of the 2023 International Symposium on Image and Signal Processing and Analysis (ISPA), Rome, Italy, 18–19 September 2023; pp. 1–6.
11. Tri, D.; Gu, A. Transformers Are SSMs: Generalized Models and Efficient Algorithms through Structured State Space Duality. *arXiv* **2024**, arXiv:2405.21060.
12. Gu, A.; Dao, T. Mamba: Linear-Time Sequence Modeling with Selective State Spaces. *arXiv* **2023**, arXiv:2312.00752.
13. Liu, Y.; Tian, Y.; Zhao, Y.; Yu, H.; Xie, L.; Wang, Y.; Ye, Q.; Liu, Y. Vmamba: Visual state space model. *arXiv* **2024**, arXiv:2401.10166v1.
14. Ruan, J.; Xiang, S. Vm-unet: Vision mamba unet for medical image segmentation. *arXiv* **2024**, arXiv:2402.02491.
15. Zhang, M.; Yu, Y.; Gu, L.; Lin, T.; Tao, X. VM-UNET-V2 Rethinking Vision Mamba UNet for Medical Image Segmentation. *arXiv* **2024**, arXiv:2403.09157.
16. Liu, M.; Jun, D.; 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.
17. Li, W.; Wu, J.; Chen, H.; Wang, Y.; Jia, Y.; Gui, G. UNet Combined With Attention Mechanism Method for Extracting Flood Submerged Range. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2022**, *15*, 6588–6597. [\[CrossRef\]](#)
18. Wang, L.; Li, W.; Wang, X.; Xu, J. Remote sensing image analysis and prediction based on improved Pix2Pix model for water environment protection of smart cities. *PeerJ Comput. Sci.* **2023**, *9*, e1292. [\[CrossRef\]](#) [\[PubMed\]](#)
19. Han, D.; Wang, Z.; Xia, Z.; Han, Y.; Pu, Y.; Ge, C.; Song, J.; Song, S.; Zheng, B.; Huang, J. Demystify Mamba in Vision: A Linear Attention Perspective. *arXiv* **2024**, arXiv:2405.16605.
20. Wang, L.; Li, R.; Zhang, C.; Fang, S.; Duan, C.; Meng, X. UNetFormer: A UNet-like transformer for efficient semantic segmentation of remote sensing urban scene imagery. *ISPRS J. Photogramm. Remote Sens.* **2022**, *190*, 196–214. [\[CrossRef\]](#)
21. Gui, L.; Suo, F.; Lin, Z.; Li, Y.; Xiang, J. Real-Time Water Area Segmentation for USV Using Enhanced U-Net. In Proceedings of the 2020 Chinese Automation Congress (CAC), Shanghai, China, 6–8 November 2020; pp. 2533–2538.

22. Chen, L.C.; Zhu, Y.; Papandreou, G.; Schroff, F.; Adam, H. Encoder-decoder with atrous separable convolution for semantic image segmentation. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 801–818.
23. Badrinarayanan, V.; Kendall, A.; Cipolla, R. SegNet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.* **2016**, *39*, 2481–2495. [[CrossRef](#)]
24. 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, BC, Canada, 10–17 October 2021; pp. 7242–7252.
25. Poudel, R.; Liwicki, S.; Cipolla, R. Fast-SCNN: Fast Semantic Segmentation Network. *BMVC* **2019**, *2019*, 187.1–187.12.
26. Howard, A.; Sandler, M.; Chu, G.; Chen, L.C.; Chen, B.; Tan, M.; Wang, W.; Zhu, Y.; Pang, R.; Vasudevan, V.; et al. Searching for MobileNetV3. In Proceedings of the 2019 IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, Republic of Korea, 27 October–2 November 2019; pp. 1314–1324.
27. Oršić, M.; Krešo, I.; Bevandic, P.; Šegvic, S. In Defense of Pre-Trained ImageNet Architectures for Real-Time Semantic Segmentation of Road-Driving Images. In Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–20 June 2019; pp. 12599–12608.
28. Vasu, P.; Gabriel, J.; Zhu, J. MobileOne: An Improved One millisecond Mobile Backbone. In Proceedings of the 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Vancouver, BC, Canada, 18–22 June 2023; pp. 7907–7917.
29. Xie, E.; Wang, W.; Yu, Z.; Anandkumar, A.; Alvarez, J.M.; Luo, P. Segformer: Simple and efficient design for semantic segmentation with transformers. *Adv. Neural Inf. Process. Syst.* **2021**, *34*, 12077–12090.
30. Nirkin, Y.; Wolf, L.; Hassner, T. HyperSeg: Patch-wise hypernetwork for real-time semantic segmentation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Nashville, TN, USA, 19–25 June 2021; pp. 4061–4070.
31. Wang, J.; Sun, K.; Cheng, T.; Jiang, B.; Deng, C.; Zhao, Y.; Liu, D.; Mu, Y.; Tan, M.; Wang, X. Deep High-Resolution Representation Learning for Visual Recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **2021**, *43*, 3349–3364. [[CrossRef](#)] [[PubMed](#)]
32. Lin, T.Y.; Dollár, P.; Girshick, R.; He, K.; Hariharan, B.; Belongie, S. Feature Pyramid Networks for Object Detection. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 936–944.
33. Huang, S.; Lu, Z.; Cheng, R.; He, C. FaPN: Feature-aligned Pyramid Network for Dense Image Prediction. In Proceedings of the 2021 IEEE/CVF International Conference on Computer Vision (ICCV), Montreal, BC, Canada, 11–17 October 2021; pp. 844–853.
34. Fu, J.; Liu, J.; Tian, H.; Li, Y.; Bao, Y.; Fang, Z.; Lu, H. Dual Attention Network for Scene Segmentation. In Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–20 June 2019; pp. 3141–3149.
35. Pan, X.; Ge, C.; Lu, R.; Song, S.; Chen, G.; Huang, Z.; Huang, G. On the Integration of Self-Attention and Convolution. In Proceedings of the 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), New Orleans, LA, USA, 18–24 June 2022; pp. 805–815.
36. Bello, I.; Zoph, B.; Le, Q.; Vaswani, A.; Shlens, J. Attention Augmented Convolutional Networks. In Proceedings of the 2019 IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, Republic of Korea, 27 October–2 November 2019; pp. 3285–3294.
37. Hu, J.; Shen, L.; Sun, G. Squeeze-and-Excitation Networks. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 7132–7141.
38. Misra, D.; Nalamada, T.; Arasanipalai, A.; Hou, Q. Rotate to Attend: Convolutional Triplet Attention Module. In Proceedings of the 2021 IEEE Winter Conference on Applications of Computer Vision (WACV), Waikoloa, HI, USA, 5–9 January 2021; pp. 3138–3147.
39. Hou, Q.; Zhou, D.; Feng, J. Coordinate Attention for Efficient Mobile Network Design. In Proceedings of the 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Nashville, TN, USA, 19–25 June 2021; pp. 13708–13717.
40. Song, Q.; Mei, K.; Huang, R. AttaNet: Attention-Augmented Network for Fast and Accurate Scene Parsing. In Proceedings of the AAAI Conference on Artificial Intelligence, Virtually, 2–9 February 2021; pp. 2567–2575.
41. Dai, Y.; Gieseke, F.; Oehmcke, S.; Wu, Y.; Barnard, K. Attentional Feature Fusion. In Proceedings of the 2021 IEEE Winter Conference on Applications of Computer Vision (WACV), Waikoloa, HI, USA, 5–9 January 2021; pp. 3559–3568.
42. Baron, E.; Zimmerman, I.; Wolf, L. 2-D SSM: A General Spatial Layer for Visual Transformers. *arXiv* **2023**, arXiv:2306.06635.
43. Jimmy, T.; Shalini, D.; Jan, K.; Scott, L.; Wonmin, B. Convolutional State Space Models for Long-Range Spatiotemporal Modeling. In Proceedings of the Thirty-Seventh Conference on Neural Information Processing Systems, New Orleans, LA, USA, 10–16 December 2023.
44. Schiff, Y.; Kao, C.; Gokaslan, A.; Dao, T.; Gu, A.; Kuleshov, V. Caduceus: Bi-Directional Equivariant Long-Range DNA Sequence Modeling. In Proceedings of the International Conference on Machine Learning, Vienna, Austria, 21–27 July 2024.
45. Zhu, L.; Liao, B.; Zhang, Q.; Wang, X.; Liu, W.; Wang, X. Vision Mamba: Efficient Visual Representation Learning with Bidirectional State Space Model. *arXiv* **2024**, arXiv:2401.09417.
46. Bovcon, B.; Muhovič, J.; Perš, J.; Kristan, M. The MaSTr1325 dataset for training deep USV obstacle detection models. In Proceedings of the 2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Macau, China, 3–8 November 2019; pp. 3431–3438.

47. Žust, L.; Perš, J.; Kristan, M. Lars: A diverse panoptic maritime obstacle detection dataset and benchmark. In Proceedings of the International Conference on Computer Vision (ICCV), Paris, France, 1–6 October 2023; pp. 20247–20257.
48. Cheng, Y.; Jiang, M.; Zhu, J.; Liu, Y. Are We Ready for Unmanned Surface Vehicles in Inland Waterways? The USVInland Multisensor Dataset and Benchmark. *IEEE Robot. Autom. Lett.* **2021**, *6*, 3964–3970. [\[CrossRef\]](#)
49. Cordts, M.; Omran, M.; Ramos, S.; Rehfeld, T.; Enzweiler, M.; Benenson, R.; Franke, U.; Roth, S.; Schiele, B. The cityscapes dataset for semantic urban scene understanding. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 3213–3223.
50. MMSegmentation Contributors. MMSegmentation: OpenMMLab Semantic Segmentation Toolbox and Benchmark. 2020. Available online: <https://github.com/open-mmlab/mms Segmentation> (accessed on 13 January 2024).
51. Shrivastava, A.; Gupta, A.; Girshick, R. Training region-based object detectors with online hard example mining. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 761–769.
52. Chen, X.; Liu, Y.; Achuthan, K. WODIS: Water Obstacle Detection Network Based on Image Segmentation for Autonomous Surface Vehicles in Maritime Environments. *IEEE Trans. Instrum. Meas.* **2021**, *70*, 1–13. [\[CrossRef\]](#)
53. Shi, D. TransNeXt: Robust Foveal Visual Perception for Vision Transformers. In Proceedings of the 2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 17–21 June 2024; pp. 17773–17783.
54. Zhao, H.; Qi, X.; Shen, X.; Shi, J.; Jia, J. ICNet for real-time semantic segmentation on high-resolution images. *Comput. Vis. ECCV* **2018**, *2018*, 405–420.
55. Fan, M.; Lai, S.; Huang, J.; Wei, X.; Chai, Z.; Luo, J.; Wei, X. Rethinking BiSeNet for real-time semantic segmentation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Nashville, TN, USA, 19–25 June 2021; pp. 9716–9725.
56. Lee, J.; Kim, D.; Ponce, J.; Ham, B. SFNet: Learning Object-Aware Semantic Correspondence. In Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–20 June 2019; pp. 2273–2282.
57. Chen, K.; Chen, B.; Liu, C.; Li, W.; Zou, Z.; Shi, Z. RSMamba: Remote Sensing Image Classification with State Space Model. *arXiv* **2024**, arXiv:2403.19654. [\[CrossRef\]](#)
58. Yu, C.; Gao, C.; Wang, J.; Yu, G.; Shen, C.; Sang, N. Bisenetv2: Bilateral network with guided aggregate-on for real-time semantic segmentation. *Int. J. Comput. Vis.* **2021**, *129*, 3051–3068. [\[CrossRef\]](#)

**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.