WODIS: Water Obstacle Detection Network based on Image Segmentation for Autonomous Surface Vehicles in Maritime Environments

Xiang Chen, Yuanchang Liu, and Kamalasudhan Achuthan

***Abstract*—A reliable obstacle detection system is crucial for Autonomous Surface Vehicles (ASVs) to realise fully autonomous navigation with no need of human intervention. However, the current detection methods have particular drawbacks such as poor detection for small objects, low estimation accuracy caused by water surface reflection and a high rate of false-positive on water-sky interference. Therefore, we propose a new encoder-decoder structured deep semantic segmentation network, which is Water Obstacle Detection network based on Image Segmentation (WODIS), to solve above mentioned problems. The first design feature of WODIS utilises the use of an encoder network to extract high-level data based on different sampling rates. In order to improve obstacle detection at sea-sky-line areas, an Attention Refine Module (ARM) activated by both global average pooling and max pooling to capture high-level information has been designed and integrated into WODIS. In addition, a Feature Fusion Module (FFM) is introduced to help concatenate the multi-dimensional high-level features in the decoder network. The WODIS is tested and cross validated using four different types of maritime datasets with the results demonstrating that mIoU of WODIS can achieve superior segmentation effects for sea level obstacles to values as high as 91.3%.**

***Index Terms*—obstacle detection, image segmentation, deep neural networks, autonomous surface vehicles**

I. Introduction

AUTONOMOUS vehicles such as drones in the air and cars on the road will gradually be achieved in the near future, and such a trend towards realising the autonomy has been increasingly endorsed by shipping industries and maritime community [1]. Currently, with the help of advanced sensors such as vision systems, LiDARs and radars, a wide applications of marine autonomy have been witnessed in various domains such as hydrography, oceanography and off-shore technologies [2]. However, most of these applications are still using remote control or semi-autonomous navigation. To enable a fully autonomous capability for marine vehicles, critical functionalities such as highly intelligent environment perception is paramount. Currently, many companies and research institutes are working on autonomous marine vehicles, such as the Rolls Royce [3] and the Kongsberg autonomous shipping project [4], the Autonomous Waterborne Applications (AAWA) research project [5], and the Maritime Unmanned Navigation through Intelligence in Networks (MUNIN) [6]. These projects are targeted to realise full-size autonomous ships to replace current cargo vessels or passenger vessel capabilities. Due to the regulations and technique issues for these types of autonomous vehicles, it is difficult to achieve this in short time.

Apart from large-scale vessels described previously, there has been rapid development of smaller vessels with autonomous control, i.e. Autonomous Surface Vehicles (ASVs), in recent years [7]. Compared with large-scale autonomous ships, ASVs have many advantages when exploiting hazardous environments, conducting long duration missions and carrying out seabed surveys. In general, ASVs can significantly reduce human resource costs, improve navigation safety, and expand operational weather windows to be able to work in poor conditions.

Accurate perception of the navigational environment with reliable obstacle detection capability is critical for safe operation of ASV. Typical environment perception uses sensors such as camera, Light Detection and Range (LiDAR), gyro compasses and Global Positioning System (GPS) to acquire substantial information for sensor fusion, which is integrated into decision-making algorithms. In recent decades, cameras and LiDAR have been predominantly used on ASVs. In particular, compared to LiDAR, camera-based detection delivers higher capability in detecting various obstacles as cameras can provide more enriched texture information. However, cameras are also prone to performance failure brought on by poor environment conditions, e.g. strong sunlight reflection and impaired sea-sky-line area interference caused by hazy weather as shown in Fig.1 (a) and (b). Furthermore, small sized obstacles, as shown in Fig.1 (c), are difficult to detect using vision systems only, which will eventually result in unpredictable risks or harmful collisions. Currently, the state-of-art algorithms on small objects detection are based on pixel-segmentation. Compared with the object detection algorithms such as YOLO, mask RCNN [8], [9], there is an increasing need of computational resources for the image segmentation methods. Also, it is technically complicated to realise a near real-time segmentation which is the most important ability for ASVs when navigating at sea.

Currently, deep convolutional neural networks have a proved ability in obtaining richer deep features with remarkable obstacle detection results on practical platforms such as Autonomous Ground Vehicles (AGVs) [10]. Due to different

X. Chen and Kamalasudhan Achuthan are with the Department of Civil, Environmental and Geomatic Engineering, University College London, Chadwich Buidling, London WC1E 6BT, UK. E-mail: xiang.chen.17@ucl.ac.uk k.achuthan@ucl.ac.uk.

Yuanchang Liu is with the Department of Mechanical Engineering, University College London, Torrington Place, London WC1E 7JE, UK. E-mail:yuanchang.liu@ucl.ac.uk

Corresponding author: Yuanchang Liu and Kamalasudhan Achuthan

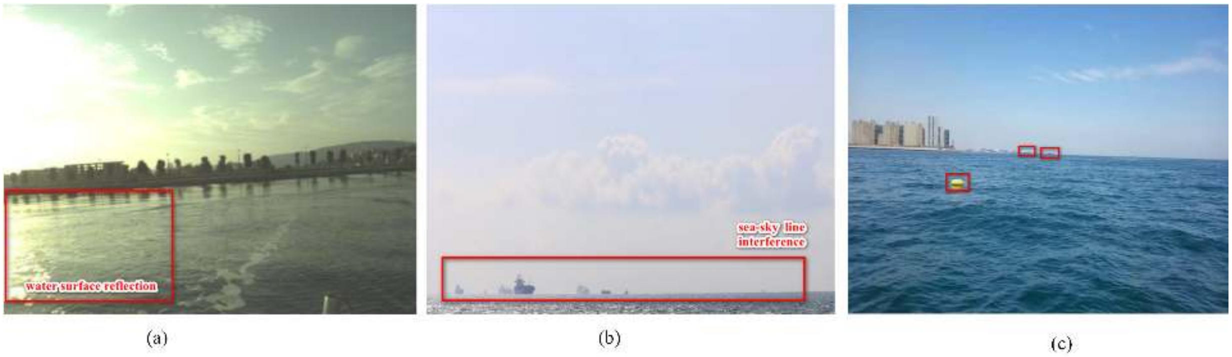

Fig. 1. Issues with camera-based detection: (a) water surface reflection intrigued by strong sunlight; (b) compromised sea-sky-line inference; (c) small obstacles detection.

applied semantic scenarios, these networks are unable to be directly implemented to an ASV's segmentation task. When compared with autonomous road vehicles, the most evident difference is that an ASV is prone to serve weather conditions (tide, poor illumination and reflection) at sea. In order to equip ASVs with strong obstacle detection systems, [11]–[14] have made attempts on improving the deep neural networks adopted on AGV or autonomous cars for a full application on ASVs. A common way for these works is to design a series of backbone networks to extract deep features from maritime scene and then transfer the extracted deep features to a designed upsampling network to output desired segmentation masks. However, most of these works have been designed with heavy structures making them infeasible for a real-time application.

To accommodate these challenges and innovate a new lightweight vision based obstacle detection system for ASVs, this paper proposes a novel Water Obstacle Detection network based on Image Segmentation (WODIS) using an encoder-decoder architecture. The encoder sub-network is used to extract deep features from the input image captured by a camera on an ASV, while the decoder sub-network is used to produce a desired segmentation mask. The contributions of our work are summarised as follows:

- A real-time and highly accurate segmentation network (named as WODIS) has been designed to accelerate inference speed and improve the segmentation accuracy for maritime obstacles detection;
- A new attention mechanism and a feature fusion module are introduced and revised based on ASV segmentation task requirements to address the issue inherent with sea-sky-line area detection and reduce false detection for obstacles at sea;
- Enriched cross validations based on four different maritime datasets are conducted with the results proving that the proposed WODIS network has a strong generalisation capability in understanding semantics of environment;
- Obstacle detection performances are validated using a series of practical maritime datasets with the results showing a high obstacle detection accuracy can be achieved when using the segmentation generated by the proposed WODIS network.

The organisation of this paper is as follow: Section II reviews related studies. Section III presents WODIS network structure, introduces the attention mechanism and feature fusion module, and discusses the design of loss functions and object detection algorithm. Experimental results are presented in Section IV with conclusions provided in Section V.

II. Related Work

Fast and accurate obstacle detection is crucial for ASVs' safe operation. With the advance of deep learning techniques, object detection based upon image segmentation using various deep neural networks has been extensively studied by the research community specialising in autonomous systems across various domains. Hence, the review of related work is centred on two aspects: 1) image segmentation methods for autonomous vehicles and 2) obstacle detection for ASVs in maritime environments.

A. Image Segmentation methods for autonomous vehicles

Image segmentation is a subset of semantic segmentation in computer vision with the aim being to generate high accuracy prediction masks under a set of conditions. With the development of the deep convolutional neural network in recent years, a series of image segmentation algorithms have been designed for practical purposes. For example, SegNet [15] uses a small encoder-decoder architecture and pooling indices strategy to reduce the network complexity. However, the disadvantage of SegNet is that feature extraction accuracy is highly related with the number of layers. When there is a large increase in the number of layers, the computational memories and processing time will increase with a significant degree making the network difficult to be trained for large dataset containing complex semantic segmentation scenes. DeepLab [11], [16], [17] technique is another important approach in image segmentation with the focus on increasing the dimension of receptive fields and not compromising the resolution of images. Atrous Spatial Pyramid Pooling (ASPP) has been widely used in the DeepLab methods to solve the problem associated with heavy backbones. Similarly, Xception network [18] has been designed with both depth-wise and point-wise convolutions to reduce the computational burden.

Note that due to the associated computational efficiency, Xception network has been implemented in the proposed WODIS network to speed up training and inference processes with the details explained in the next section.

Following the success of DeepLab, a large number of recent networks have been designed using the ASPP module. Among them, ESPNet [19] designs a new spatial pyramid module to improve the computational efficiency. BiSeNet [13] introduces a spatial path and a context path to reduce calculation complexity. Both ESPNet and BiSeNet rely only on one branch to produce features extraction with the rest of decoder network used for upsampling to recover image dimensions. Note that a problem exists that the feature extraction may not be sufficient in the encoder sub-network resulting a compromised output mask. Therefore, DFANet [14] has been proposed with an argument that with the layers of the encoder being deeper, the aggregation may be even worse than high-level feature maps. The solution to such an aggregation problem is to repeat the feature extraction procedures in the encoder sub-network several times. Based upon the discussion on these most recent networks, it should be emphasised that the trend of image segmentation algorithms has been diverted from focusing on the effect of segmentation to the acceleration of the segmentation speed. Such a trend is accompanied by designing the backbones of the feature extraction network from using heavy-weight networks such as VGG [20], ResNet (18, 34, 50, 101, 152) [21] to light-weight networks such as Xception [18], MobileNet [22].

At this juncture, it is important to summarise the variations of image segmentation networks. As detailed in Fig. 2, it can be observed that the network structure evolved from a multi-branch (Fig. 2 (a)) structure to a spatial pyramid pooling architecture (Fig. 2 (b)) and finally to a feature reusing principle (Fig. 2 (c)). In terms of building deep neural networks for image segmentation for ASVs, following aspects should be considered. When an ASV is navigating, the most significant aspects in its surrounding environments are sea and sky resulting a background that is simpler than the road scene. However, the difficulty is the sea-sky-line area detection, especially in unfavourable weather conditions, which can generate false positive obstacle detection around sea-sky-line areas as shown in Fig. 1(b).

B. Obstacle detection for ASV in maritime environments

The success of an accurate and efficient obstacle detection algorithm relies on three aspects: (1) a multi-sensor system to perceive an ASV's surrounding environments, (2) a fast and accurate algorithm to process the acquired data, (3) a dataset that is rich in scenarios for algorithms training.

For a safe navigation in maritime environments, a variety of types and modes of sensors are deployed to capture various information, including Automatic Identification System (AIS) [23], radar [24], camera, LiDAR, sonar [25] and speed log [26]. Of these sensors, the dominant equipment is the AIS, which is used in maritime navigation for broadcasting each ship's position, heading and speed to help inform correct situational awareness. Although AIS brings several benefits for the environment perception, it can be easily affected by severe weather conditions when being used for a short-range detection [27], [28]. Therefore, for small sized autonomous ships, AIS may not be the first option for collecting perceptional data. With the advance in sensors development for maritime environments, new sensors such as LiDAR, camera and etc. are being introduced and mounted onboard. [29] uses three dimensional point cloud produced by LiDAR to detect objects. Similarly, [30] uses LiDAR to generate a stable navigable region for ASVs.

Based on these developments of sensor technology, the associated efficient and robust obstacle detection algorithms are also being innovated rapidly. Before deep learning methods became the dominant methodology, one of the traditional paradigms was to first use feature extraction algorithms such as the Hough transformation method to detect sea-sky-line [31] and then, based on different grey scale values in the sea-sky-line area, the Otsu threshold method is used to detect obstacle edges [32]. These methods need substantial computational resources and the shape of an obstacle may not be easily identified due to the noises generated by sea wave. Consequently, disadvantages associated with such a paradigm include a high rate of false positive and a compromised capability when deployed in practical environments.

Most recent studies in marine obstacle detection are using machine learning algorithms, which can significantly improve the detection results. For example, [33] assumes that ASVs are always navigating in a diverse environment and in order to achieve a continuous obstacle detection using vision systems only, a robust generative graphical model based upon Markov random field framework has been proposed. The model is trained using the expectationmaximisation (EM) method, and by leveraging the inherent fast inference capability, a real-time obstacle detection under fast frame-per-second (fps) can be achieved. Following the same paradigm, [34], [35] further improve the work to have a higher detection accuracy rate by using stereo visions.

When an ASV is travelling at sea, its motion states are prone to adverse weather conditions such as high waves or strong winds, which can cause onboard cameras failing to track certain obstacles. Also, under high wave conditions, the sea-sky-line area detection is no longer accurate. To address these issues, information from Inertia Measurement Unit (IMU) has been integrated with the image in the work of [36] and [37]. The ablation study shows that it is effective to mitigate external influences. Also, [38] designs a background subtraction algorithm and directly detects objects from the subtracted background. However, the result shows that it has a higher false positive rate on obstacles. At the same time, the work indicates that it needs a pixel-based segmentation method to subtract the objects from background in order to maintain a high true positive rate.

Apart from methods and sensors, another important factor when aiming to produce accurate detection results is the availability of training datasets of maritime environments. Currently there are only a limited number of publicly available datasets. For example, the MaSTr1325 [39] is collected for ASVs' object detection. The Multi-modal Marine Obstacle

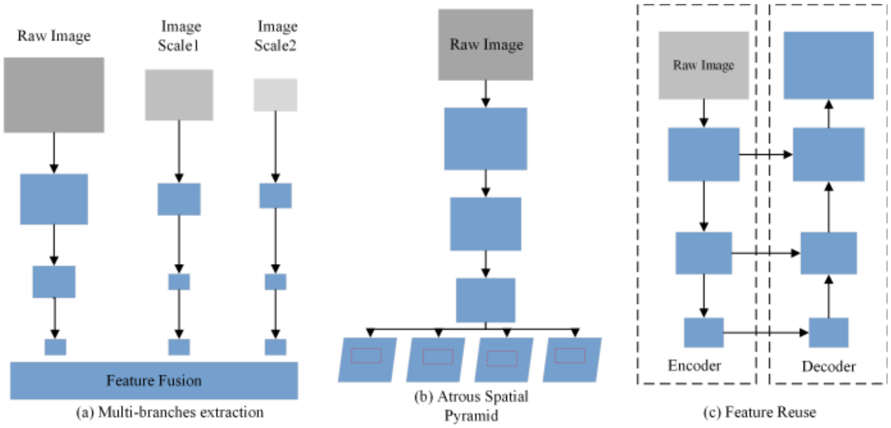

Fig. 2. Image segmentation network evolution, (a) Multi-branch downsampling by manual scaling; (b) Atrous Spatial pyramid pooling; (c) Feature reusing with encoder decoder architecture.

Detection Dataset 2 (MODD2) [36] has been intensively used for cross-validation due to the enriched scenes included. The Singapore Maritime Dataset (SMD) [40] is another maritime dataset collected from offshore Singapore and the background in most scenes does not contain any onshore buildings or obstructions. Finally, the Marine Image Dataset (MID) [41] contains a range of marine obstacles such as buoys and floating objects. In order to validate the generalisation of our proposed method, all these four datasets have been used in this paper.

III. WODIS NETWORK FOR A RELIABLE IMAGE SEGMENTATION

The overview of WODIS network is described in Section III-A, the Attention Refine Module (ARM) and Feature Fusion Module (FFM) are explained in Section III-B and the sea-sky-line enhancing mechanism is presented in Section III-C. Finally, the post processing algorithm will be discussed in Section III-D.

A. Network Architecture Overview

Based on recent studies on the semantic segmentation [13], [14], the encoder-decoder network has become the mainstream to replace other architectures such as the multi-scale information structure or the spatial pyramid network. The encoder network is commonly composed of backbones (using networks including VGG, ResNet, Xception and etc.), which are well pre-trained using large datasets (ImageNet, COCO, PASCAL VOC etc.). These backbones have a strong generalisation ability for feature extraction but are normally heavy in parameters making them difficult to be applied for real-time segmentation. Such a drawback is especially acute for ASVs' obstacle detection, which requires a lightweight backbone model to speed up inference while maintaining a high accuracy for obstacle segmentation.

To address these issues, the WODIS has been proposed to have a U-Net encoder-decoder structure with two sub-networks as shown in Fig. 3. The aim of the encoder sub-network is to extract deep features from input images, while the decoder is to fuse high-level features (global specifications of objects, such as positions and shapes) and low-level features (local specifications of objects, such as edges, corners and texture) from the encoder sub-network to generate an object segmentation mask.

Within the WODIS, the encoder sub-network consists of separable convolutions that are derived from Xception network [18]. The benefit of the Xception network is that it uses the depth-wise separable convolutions (short for 'Sep conv') replacing traditional convolutions to reduce the computing parameters, which makes it an ideal structure for a lightweight network design. By referring to the network design of the entry flow within the Xception, parameters of our WODIS are shown in Table I. Each 'Sep conv' is stacked by three separable convolutional layers. Kernel sizes for all 'Sep conv' are configured to be '3 x 3' with the stride number being 2 for all layers, while the only difference between different 'Sep conv' is the channel number. The repeating times in Table I represents how many each 'Sep conv' layer is repeated. For example, 'Sep conv2_x' with $x = 4$ means the separable convolutional layer should be repeated 4 times.

The backbone for our WODIS encoder sub-network follows the flow of $conv1 \rightarrow Sep\ conv2\_x \rightarrow Sep\ conv3\_x \rightarrow Sep\ conv4\_x$, which consists of a sub-stage in our network as shown by the red dashed box in Fig. 3. Because a single sub-stage is not able to fully fuse the high-level and low-level feature maps, it is necessary to repeat the backbone for several times and in the WODIS (as shown in Fig. 3), the backbone is repeated three times (the reason of such a design will be presented in Section IV-C). At the end of the backbone, we introduce an Attention Refine Module (ARM) to adjust the high-level feature weight. A detailed explanation of the ARM as well as its advantages will be provided in Section III-B. The decoder in our network is primarily consisting of several Feature Fusion Modules (FFM) with the aim being to fuse feature maps coming from multiple levels of the encoder. Therefore, apart from being connected to the Attention Refine Module within the third sub-stage, FFMs also process information from other layers from the encoder that can be possibly in different dimensions. The working mechanism of the FFM will be explained in Section III-B.

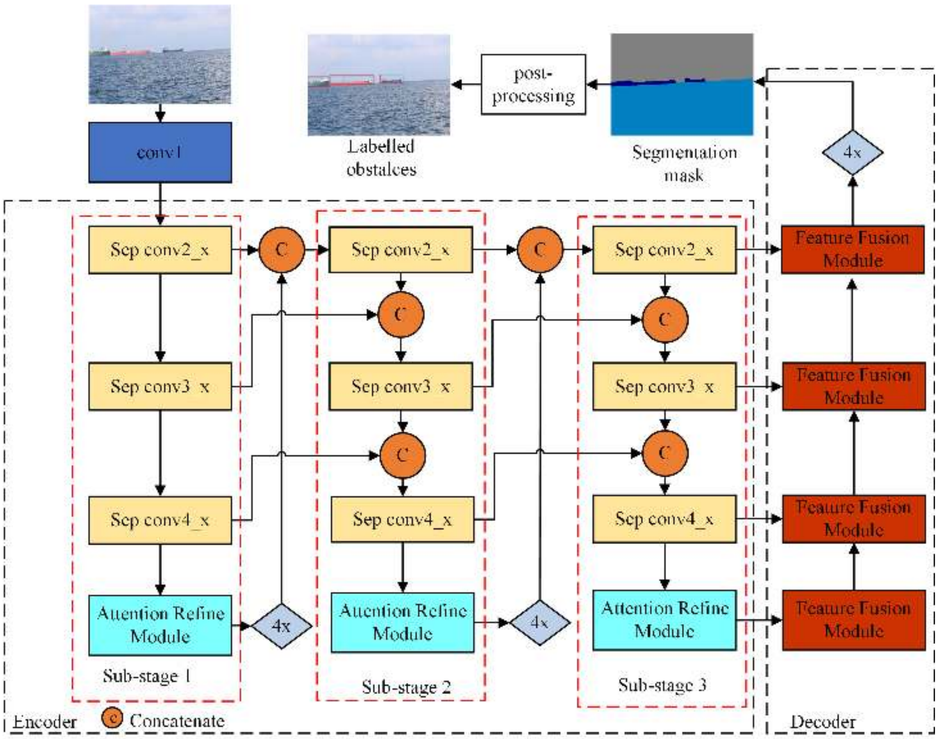

Fig. 3. The architecture of the Water Obstacle Detection network based on Image Segmentation (WODIS). The network consists of an encoder and a decoder network. Blocks in yellow represent networks stacked by a series of separable convolution layers. '4x' represents taking an upsampling operation for 4 times. The dark blue 'Conv1' block represents the convolutional layer before images are passed into the encoder sub-network. The output of the network is a segmentation mask which classifies sea, sky and obstacles by pixel. Based on the output mask, all obstacles are able to be labelled by rectangles.

TABLE I
FEATURE EXTRACTION BACKBONE PARAMETERS

Stage	Feature extraction backbone setting			
	kernel size	channel	stride	repeating times
Conv1	3 x 3	8	2	-
Sep conv2_x	3 x 3 3 x 3 3 x 3	12 12 48	2 2 2	x=4
Sep conv3_x	3 x 3 3 x 3 3 x 3	24 24 96	2 2 2	x=6
Sep conv4_x	3 x 3 3 x 3 3 x 3	48 48 192	2 2 2	x=4

B. Attention Refine Module and Feature Fusion Module

1) Attention Refine Module: It is well known that attentions play important roles in human perception [42]. The most important role of the human vision is that rather than attempting to process a whole scene at once, humans can exploit a sequence of partial glimpses and selectively focus on salient parts in order to have a better capture of visual structure [43].

The performance of an ASV's obstacle detection can be easily influenced by different situations on water surfaces such as reflections, obstacle mirroring or sky and sea misdiagnosis, as depicted in Fig.1. Such a compromised performance can be further worsened if a segmentation model fails to identify the sea and sky line area. To avoid a misunderstanding, segmentation networks need to have an attention ability to distinguish between general areas and focusing areas, which, for ASVs, are the sea-sky-line areas.

Based on the state-of-the-art [11], [13], the Attention Refine Module (ARM), an attention mechanism, has been successfully introduced to classify an image area into different weighed parts. ARM is an unique module to optimise each sub-stage feature characteristics, and uses a global average pooling to capture global context from feature maps and calculate an attention vector for feature learning. By an easy integration of global context information and local feature information without upsampling, ARM is not only able to enhance the image segmentation attention but also to accelerate the inference speed.

Motivated by ARM component developed in BiSeNet [13], we propose a revised Attention Refine Module (shown in Fig. 4) specifically for WODIS network with the most evident difference being the simultaneous use of the max pooling and global average pooling to process high-level feature maps. Based on the discussion in [44], the global average pooling is focused on the spatial information learning and the max pooling is suited for channel information combination. The proposed ARM firstly generates two 1-dimension feature maps by using the average pooling and max pooling operations denoted as $F_{avg} \in R^{1\times1\times C}$ and $F_{max} \in R^{1\times1\times C}$, respectively.

The input feature map of the ARM is from $F_{in} \in R^{H \times W \times C}$, where H, W and C denote the height, the width and channel numbers of the high-level feature map. The global average pooling in the c-th elements of the input feature map is expressed as:

$$\boldsymbol{F}^c_{avg} = \frac{1}{H \times W} \sum_{i=1}^{H} \sum_{j=1}^{W} \boldsymbol{f}^c(i,j) \tag{1}$$

and the max pooling in the c-th elements of the input feature map is expressed as:

$$\boldsymbol{F}^c_{max} = Max[\boldsymbol{f}^c(i,j)], \quad i \in [1, H], \quad j \in [1, W] \tag{2}$$

where $\boldsymbol{f}^c(i,j)$ represents the c-th elements of the input feature map F_{in} at position (i,j).

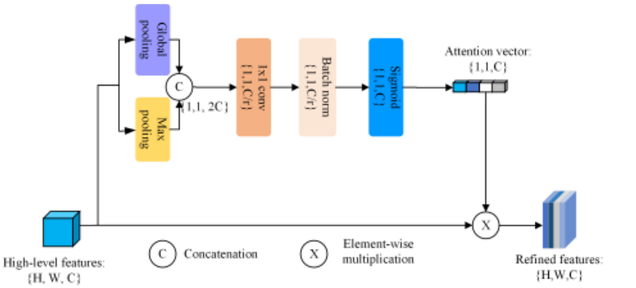
Global pooling
Max pooling
C
{1,1, 2C}
1x1 conv {1,1,C/r}
Batch norm {1,1,C/r}
Sigmoid {1,1,C}
Attention vector: {1,1,C}
X
High-level features: {H, W, C}
C Concatenation
X Element-wise multiplication
Refined features: {H,W,C}

Fig. 4. The architecture of the Attention Refine Module (ARM). H, W, C denotes the height, width and the number of channels of input feature map. r is the reduction ratio and is set to 16.

Then, $\boldsymbol{F}_{avg}$ and $\boldsymbol{F}_{max}$ are concatenated and the dimension of the feature becomes $(1, 1, 2C)$. By using a 1×1 convolution, the output dimension of the feature map can be converted to $(1, 1, C/r)$. The reduction ratio r is set to be 16 in this paper to reduce the channel numbers. A batch normalisation is introduced to reduce the gradient vanish or explosion and a Sigmoid function is applied to activate the convolution result to learn a 1-dimension attentive weight vector denoted as $(1, 1, C)$. The attention weight vector can be used as an improved feature map and after a multiplication with the input feature map, the final output, denoted as the refined features (H, W, C), can be obtained.

Different from the BiSeNet, WODIS not only uses the global average pooling in the ARM to capture the global context information of the input feature, but introduces the max pooling to extract the local discriminative information. More specifically, for ASVs image segmentation, the global average pooling can better learn obstacles or background positions; whereas the max pooling operation can improve the local feature distinguishing for details such as obstacle edges and shapes. Therefore, using and integrating such a revised ARM into the WODIS network, an improved detection accuracy can be obtained.

2) Feature Fusion Module: Feature maps from different encoder layers have different dimensional representation. For example, features extracted from the ARM in 'SuB-stage 3' mainly contain high-level information; whereas output features from the 'Sep conv2_x' or 'Sep conv3_x' involve low-level information. Therefore, in order to reduce the gap between the high-level and low-level feature, the Feature Fusion Module (FFM) is proposed to improve the fusion efficiency with its structure shown in Fig.5.

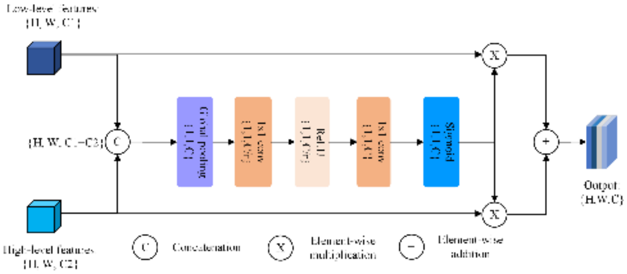

Fig. 5. The structure of Feature Fusion Module (FFM). H and W represent the height and width of input feature maps. C_1 and C_2 denote the different number of channels in different feature map. r represents reduction ratio and the number is 16 in this paper.

Given the low-level feature $\boldsymbol{X}_{low} \in \boldsymbol{R}^{H \times W \times C}$ and the high-level feature $\boldsymbol{X}_{high} \in \boldsymbol{R}^{H \times W \times C}$ inputs, where H, W and C denote the height, width and number of channels, the two features are first concatenated as $\boldsymbol{X}_{concat}$. Next, the global average pooling is conducted on the $\boldsymbol{X}_{concat}$ to generate an average feature $\boldsymbol{X}_{avg}$ shown in Eq.3.

$$\boldsymbol{X}^c_{avg} = \frac{1}{H \times W} \sum_{i=1}^{H} \sum_{j=1}^{W} \boldsymbol{X}^c_{concat}(i,j) \tag{3}$$

where, $\boldsymbol{X}^c_{concat}$ denotes the c-th elements of the feature map $\boldsymbol{X}_{concat}$ at position (i,j). Two 1×1 convolutions are added to reduce the dimension of the $\boldsymbol{X}_{avg}$. The reduction ratio r is 16 in the FFM. After reducing the dimensions, the output dimension $\boldsymbol{W}_{FFM}$ is $(1, 1, C)$ after the Sigmoid activation function. Finally, we use the element-wise addition to add all the features as:

$$\boldsymbol{X}_{FFM} = (\boldsymbol{X}_{low} \otimes \boldsymbol{W}_{FFM}) \oplus (\boldsymbol{X}_{high} \otimes \boldsymbol{W}_{FFM}) \tag{4}$$

where, $\otimes$ and $\oplus$ represent the channel-wise multiplication and the element-wise addition.

C. Network Implementation to Enhance Sea-sky-line Area Segmentation

For an image segmentation for ASVs' sailing environments, three types of objects, i.e. sky, sea and obstacles, need to be classified with the most prominent ones being the obstacles at the sea surfaces. Compared with the labelled sky and sea objects in training dataset, available labelled obstacles only occupy a small portion. Using such an unbalanced training dataset, the segmentation/detection results in the inference stage will be largely impaired. To solve such an issue, the most common way is to have well balanced and fine-tuned weights for loss functions when a network is being trained. Therefore, in the WODIS, a hybrid training loss function including both focal and cross entropy loss functions is implemented.

We denote the $t_i = \{obstacle, sea, sky\}$ as three segmentation types and each type i is expressed by the number from 1 to 3. For example, t_1 is obstacle, t_2 means sea and t_3 represents sky. The focal loss function [45] has been proved to be effective to solve the label unbalance problem, and for our WODIS it is defined as:

$$L_{focs}(p_{t_i}) = -(1-p_{t_i})^{\gamma}log(p_{t_i}) \tag{5}$$

where, p_{t_i} is the estimated probability for all three different types of objects t_i and γ is the hyperparameter (γ is set to be 2 during the training process). For example, if labels of sea (t_2) and sky (t_3) have a larger proportion than the obstacle (t_1) in the dataset, t_2 and t_3 become easier to be classified than t_1 and in order to have a more accurate segmentation for label t_1, a higher weight for this label should be assigned within the loss function making the parameter γ more biased towards obstacles.

Also, in our image segmentation task, there are three types of objects to be classified, which can be regarded as a multi-labels classification problem. We use the cross entropy loss function to classify different types of objects, as shown in Eq.6:

$$L_{loss}(y_{t_i}, \hat{y_{t_i}}) = -(y_{t_i}log(\hat{y_{t_i}}) + (1-y_{t_i})log(1-\hat{y_{t_i}})) \tag{6}$$

where, y_{t_i} is the label of three different types and the $\hat{y_{t_i}}$ is the estimated output through the network. The total loss function for WODIS is shown as:

$$L_{total} = (1-\lambda) * L_{focs} + \lambda * L_{loss} \tag{7}$$

where, $\lambda \in (0,1)$.

D. Post-processing after getting segmentation masks

The advantage of image segmentation over the object detection networks is that segmentation methods are based on pixel-by-pixel classification, which is more refined and accurate than object detection networks. The output of image segmentation through WODIS network is a segmentation mask containing spatial information on sea, sky and obstacles, which can be further used by post-processing algorithms for reliable obstacle detection. The post processing using segmentation mask is to use a minimal enclosing rectangle to highlight the water surface obstacles, as shown in Fig. 6. The pseudocode of the minimal enclosing rectangle algorithm used in this study is shown in Algorithm 1. Note that for ASVs, water surface obstacles are the most significant while sky-level objects can be neglected to reduce computational burden.

IV. Experiment results and discussions

In this section, the dataset and evaluation metrics are first described in Section IV-A. The data augmentation and training details are discussed in Section IV-B with the performance of feature extraction fusion in the encoder sub-network shown in Section IV-C. The comparison with other state-of-the-art networks in terms of the speed and accuracy is shown in Section IV-D and the ablation study for Attention Refine Module is shown in Section IV-E. The object detection results on the MaSTr1325, MODD2, MID, SMD dataset are finally presented in Section IV-F.

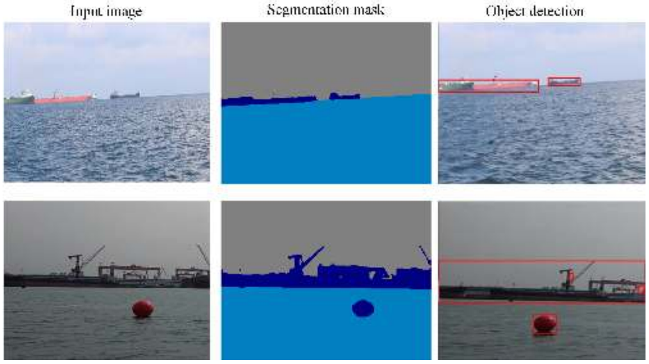

Fig. 6. Input image captured by ASV (left), segmentation mask obtained through our WODIS (middle) and post-processing for object detection (right). Sea, sky and obstacles are labelled with blue, grey and deep blue, respectively.

Algorithm 1 Minimal enclosing rectangle for obstacles

```
Input:
    The input images, $image$;
    The segmentation mask, $mask$;
Output:
    The labelled rectangle for water surface obstacles, $image\_output$;
Grayscale transformation for the input of $image$ and $mask$;
output:$image\_gray$ and $mask\_gray$;
Setting gray threshold range from $(230, 255)$ for $image\_gray$ and $mask\_gray$;
Based on the contour finding algorithm $cv2.findContour$ from OpenCV for finding the contours of $image\_gray$ and $mask\_gray$;
output:$contour\_image$ and $contour\_mask$;
Drawing the contour using the red color $(255, 0, 0)$ on the $image$;
return Output image: $raw\_image\_output$;
```

A. Dataset and Evaluation Metrics

The WODIS is trained on the MaSTr1325 dataset [39], which is a new large-scale marine semantic segmentation training dataset for the development of obstacle detection methods for small-sized coastal ASVs [37]. The dataset contains 1325 high resolution images taken in realistic conditions and all images are per-pixel labelled into three types: sea, sky and obstacles. The MaSTr1325 is used mainly for training of our model and in order to evaluate our model performance in various conditions, another three datasets (MODD2, the Singapore Maritime Dataset (SMD) and the Marine Image Dataset (MID)) are implemented for testing. The MODD2 contains mostly complex scenes such as sea water reflections, object mirroring, fog or haze and big waves, making it an ideal testing dataset to reflect extreme navigation environments. However, both the MaSTr1325 and MODD2 are captured around coastal areas and not able to be used for validating algorithms' performances in open sea areas. Therefore, we use additional two datasets, i.e. SMD and MID, which are taken in open sea areas, to validate the performance of detecting less silent obstacle features. Examples of the four dataset are shown in Fig.7.

To validate the segmentation performance of our proposed network, evaluation metrics including the precision, recall, F1 score and mean Intersection over Union (mIoU) [46] are used, which are defined, respectively, as follows:

$$precision = \frac{TP}{TP + FP} \tag{8}$$

$$recall = \frac{TP}{TP + FN} \tag{9}$$

$$F_1 = \frac{2 * precision * recall}{precision + recall} \tag{10}$$

$$mIoU = \frac{1}{k+1} \frac{Region_x \cap Region_{GT}}{Region_x \cup Reion_{GT}} \tag{11}$$

where TP represents the number of positive samples that are predicted correctly, FP denotes the number of negative samples that are predicted as positive, and FN indicates the number of positive samples that are predicted as negative samples. k in Eq. 11 is the total classification types, and $k = 3$ in this paper as there are three types, i.e. sky, sea and obstacles. $Region_x$ represents the predicted area and $Region_{GT}$ means the ground truth. In addition, the Floating Point Operations (FLOPs) and network parameters (Params) are used to evaluate the network complexity. The Frame per Second (fps) is used for indicating the inference speed.

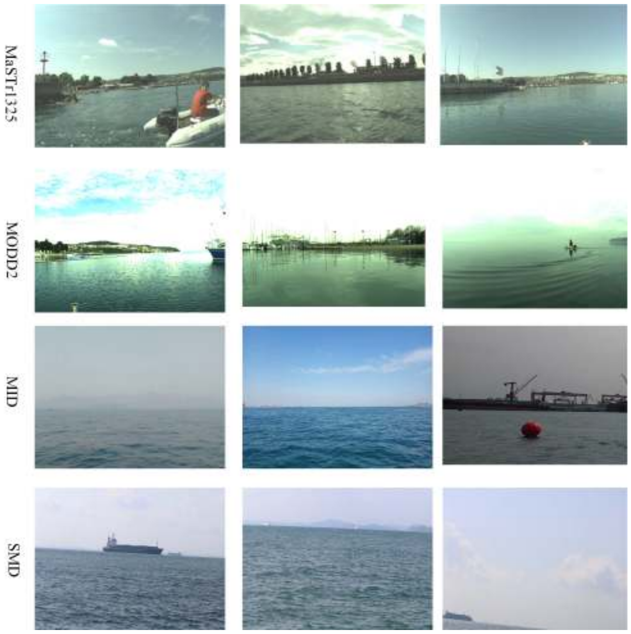

Fig. 7. Example images of the four dataset, MaSTr1325 (top), MODD2 (second), Marine Image Dataset (MID) (third) and Singapore Maritime Dataset (SMD) (bottom). The four dataset have different features. MODD2 has various water reflection, glittering and sunrise and sunset. The background of SMD mostly belongs to the maritime environment, barely including shore or land scenes. The MID has plentiful marine objects, such as buoys, floating objects and ships. WODIS is only trained in MaSTr1325 and the other three dataset are regarded as testing dataset.

B. Data Augmentation and Training Setup

Data augmentation is one of the important steps to increase the generalisation capability of models. Before training, the MaSTr1325 dataset has a total 1325 images and each image has a resolution of 512 x 384 pixels. To improve the diversity of the dataset, transformation methods such as mirroring, rotation and shadow are applied in the MaSTr1325 dataset, and the overall number of images after data augmentation is 38240.

TABLE II
PARAMETERS OF HARDWARE AND SOFTWARE

Drives	Parameters
CPU-inference	i7-7600 2.8GHz
CPU-training	Xeon 6240 2.6GHz
GPU	Nvidia Tesla V100
Deep Learning Network API	Pytorch 1.6
Language	Python, C++
Image size	512 x 384
Training epochs	100
Optimiser	Adam
Learning rate	0.0001, 0.003
Batch size	2, 4
Training images	38240

The network is trained using an Adam optimiser with the initial learning rate being 10^{-4}. The backbone of the network is derived from Xception which is trained on the ImageNet dataset and the network only fetches the Entry Flow weights from the Xception. The network is trained around 100 epochs in the training dataset. After 50 epochs of training, the learning rate is set to be 0.003 for the rest of epochs and the batch size is 2 for each epoch training. WODIS is implemented in Pytorch. All experiments are undertaken using the High Performance Computing at UCL with two nodes of NVidia Tesla V100 and the inference tests are conducted on the Intel Core i7-7600 2.8 GHz CPU with 16 GB RAM. All parameters of hardware and software are shown in Table II.

C. Encoder Sub-network Backbone Initialisation

The difference of the WODIS with other networks is that multiple sub-stages are used in the encoder subnetwork to enhance the feature extraction ability. Although adding more sub-stages in the encoder sub-network will inevitably help boosting feature extraction ability, it is also associated with negative influences for the segmentation. Therefore, in order to determine an optimal number of sub-stages in the WODIS, a series of initial setups have been carried out using the MaSTr1325 training dataset. The experiment has been undertaken in a way that after each backbone setup is trained for 20 epochs, a testing is carried out using the testing dataset.

The results of backbone initialisation experiments are shown in Table III. It can be observed that when only one backbone is used, the mIoU is 79.5% and when the backbone is repeated for three times, the mIoU is improved from 79.5% to the peak value of 91.3%, where a highly accurate segmentation

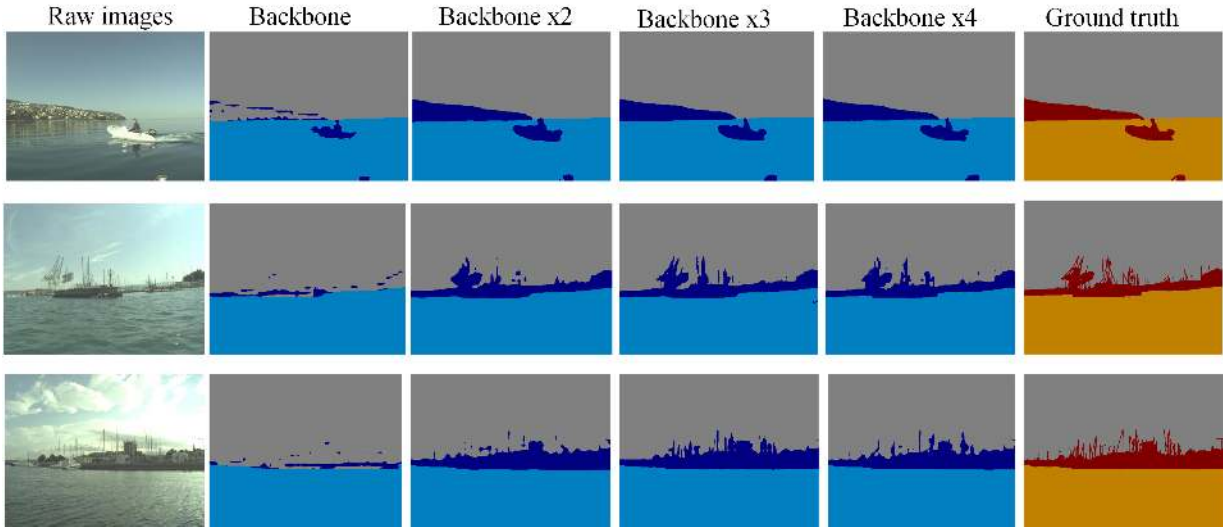

Fig. 8. Results of the WODIS backbone on MaSTr1325 test dataset. The first column is raw images, and column 2-5 show the output of each backbone in WODIS. The final column is the ground truth of the raw images. Compared with ground truth, repeating backbone three times in the fourth column gets more details like the texture and outline of the objects, especially using the second and third input images.

result can be generated. If we continue to repeat backbone for four times, the accuracy is compromised with the mIoU been decreased from 91.3% to 85.7%. Based on the backbone initialisation, we suppose that repeating the backbone three times is an optimal option for the MaSTr1325 dataset. Also, we argue that the receptive field of repeating backbone 4 times is larger than the image size (512 x 384). Therefore, if we continue to increase the number of repetitions of the backbone, a large number of noises will be generated and the feature map will become too small to be used as a valid extraction. Therefore, a proper backbone that fits the input image size becomes significant. Based on above initialisation setups, repeating the backbone three times is determined to be optimal for this training dataset.

TABLE III
PERFORMANCE COMPARISON OF OUR WODIS SUB-STAGES STRATEGY.

Model	FLOPs (MB)	Params (MB)	mIoU(%)
Backbone	60.8	42.1	79.5
Backbone x2[1]	89.2	60.8	80.3
Backbone x3	105.7	89.5	**91.3**
Backbone x4	135.3	96.4	85.7

[1] 'x N' means the replication number of backbone.

Fig. 8 displays the segmentation results using backbones with four different numbers of sub-stages, respectively. It can be observed that in the second column, where only one backbone has been implemented, the extraction effectiveness compared with the ground truth mask is not satisfactory with a large number of key features missing. Repeating the backbone for two and three iterations as illustrated in the third and fourth column of Fig. 8 shows that the outputs become smoother and more details such as texture and outline of the objects from input images can be segmented. Comparing the fifth column with the fourth column, the ability of details extraction by repeating the backbone three iterations (shown in column 4) is stronger than for four iterations (shown in column 5), which supports our argument that a continued iterations of the repetition of backbones does not guarantee an improvement of feature extraction performance.

TABLE IV
SPEED COMPARISON ON THE MASTR1325 DATASET

Model	FLOPs(MB)	Time(ms)	Frame(fps)	mIoU(%)
BiSeNet	68.5	21	**45.7**	67.9
SegNet	103.6	1286	0.85	81.8
DeepLabV3+	145.2	3287	0.56	85.4
WODIS	105.7	28	43.2	**91.3**

TABLE V
RESULTS ON SMD, MODD2 AND MID.

Dataset	Model	Time(ms)	Frame(fps)	mIoU(%)
SMD	BiSeNet	19.8	**43.2**	65.6
	SegNet	205.1	0.65	51.3
	DeepLabV3+	112.1	0.43	79.8
	WODIS	18.01	45.6	**94.2**
MODD2	BiSeNet	20.5	**41.2**	48.5
	SegNet	219.5	0.72	60.5
	DeepLabV3+	117.3	0.67	78.3
	WODIS	18.95	41.9	**89.5**
MID	BiSeNet	27.7	**39.8**	32.1
	SegNet	302.5	0.83	70.2
	DeepLabV3+	155.2	0.78	79.1
	WODIS	20.5	46.8	**91.2**

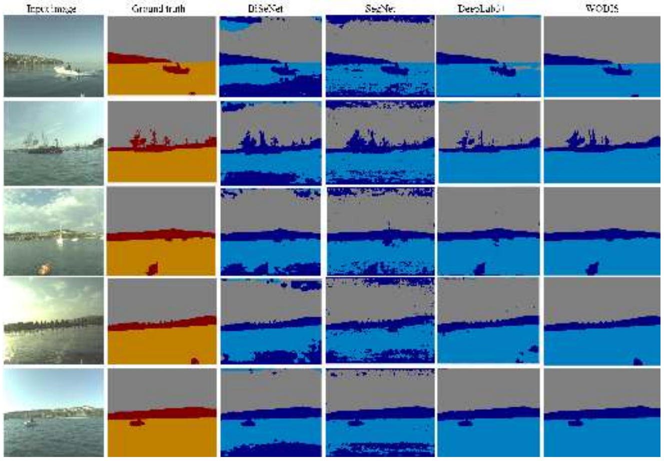

Fig. 9. Qualitative comparison on the MaSTr1325 testing Dataset. The sea, sky and obstacle are denoted by blue, grey and deep blue colours, respectively. Although the BiSeNet has a higher inference speed, the DeepLba3+ and WODIS have more accurate segmentation results.

D. Comparison with other networks

In this section, the proposed WODIS network has been compared to three other benchmark networks including BiSeNet, SegNet and DeepLabv3+, The BiSeNet [13] is a lightweight networks and its inference speed is faster than the other networks. The SegNet [15] and DeepLabV3+ [16] networks have strong ability in feature extraction.

All comparison experiments are conducted on Intel Core i7-7600 2.8 GHz CPU with 16 GB RAM. All input images are resized into 512 x 384 pixels and data augmentation methods in the MaSTr1325 test dataset are not applied. The total number of test images is 40.

Inference speed is a significant factor for ASVs to detect obstacles and a speed comparison among different networks is shown in Table IV. From the metric of FLOPS, the BiSeNet has a smaller amount of parameters than the other three models and the DeepLabV3+ is the heaviest model in the MaSTr1325 dataset (the higher the FLOPS, the heavier the network). However, although the BiSeNet has the smallest parameters, the mIOU only reaches to 67.9%, which indicates that the segmentation effectiveness is worse than the other models. Also, from Table IV, it can be seen that both BiSeNet and WODIS have very fast processing time whereas SegNet and DeepLabV3+ are significantly slow. However, although the DeepLabV3+ and SegNet have relatively slow inference speeds, their mIoU values are higher than BiSeNet's but lower than WODIS's. The WODIS achieves the highest mIoU around 91.3% compared with the other three networks. Therefore, for a practical application, if the model is prone to inference speed, the BiSeNet is the best option but with a significant compromise in accuracy. If the application considers the trad-off between speed and inference accuracy, the WODIS is then the best model.

A further comparison can also indicate the real-time performance of our proposed WODIS. For example, compared with BiSeNet, the WODIS can almost provide an equivalent fast processing speed (43 fps of WODIS as opposed to 45 fps of BiSeNet) but with a much higher mIoU value (the mIoU value of WODIS can reach as high as 91.3% whereas BiSeNet can only reach a 67.9% of mIoU). This can well demonstrate that our proposed WODIS has a highly accurate and efficient segmentation capability, that is well suited for real-time requirement. Such a capacity can further be proved by some examples of output masks shown in Fig. 9. It can be seen that although BiSeNet has a higher inference speed, the output mask is not as good as the DeepLabv3+ and WODIS. Also, the output mask of SegNet shows that the sea and sky can hardly be distinguished.

We also evaluate WODIS on other datasets (MID, MODD2 and SMD). Due to the various sizes in these datasets, the test images are resized into 512 x 384 pixels and we randomly select 40 images from these dataset for testing. Other settings are retained the same as settings within the MaSTr1325 testing. The results are presented in Table V. Similarly, by retaining a relatively fast inference speed, the highest mIoU is also generated by the WODIS in these dataset, which proves that WODIS makes a good balance between inference speed and accuracy. In order to have an intuitive qualitative comparison using in the three dataset, all comparison images and masks are presented in Fig. 10, 11 and 12, where the best performances are all provided by our proposed WODIS.

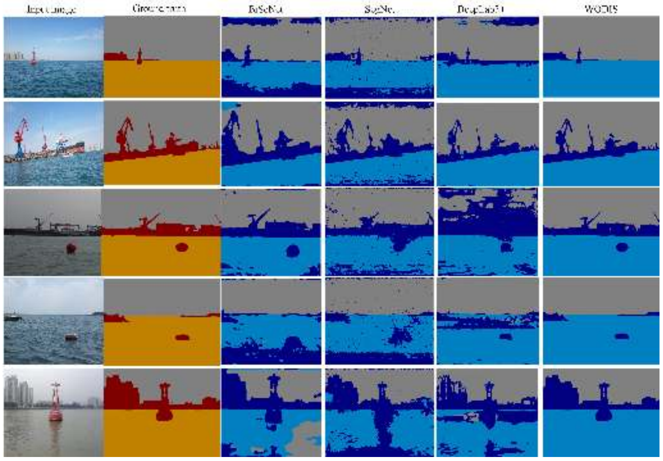

Fig. 10. Visual comparison on the Marine Image Dataset (MID). The MID has plentiful marine obstacles such as buoys, floating objects and ships with different sizes. If the segmentation scene becomes more complex, it becomes difficult for all networks except the WODIS to distinguish the sea, sky and objects.

E. Ablation Study of Attention Refine Module

Based on the Section III-B, the attention mechanism has been integrated into the WODIS. The difference between attention module used in the WODIS and the BiSeNet is that WODIS uses the global average pooling and max pooling simultaneously. The ablation study for the attention module is conducted via a series of experiments to test the effective of the global average pooling and the max pooling in our network with the results presented in Table VI. We set four different comparison experiments: (1) WODIS without ARM; (2) WODIS without global average pooling; (3) WODIS

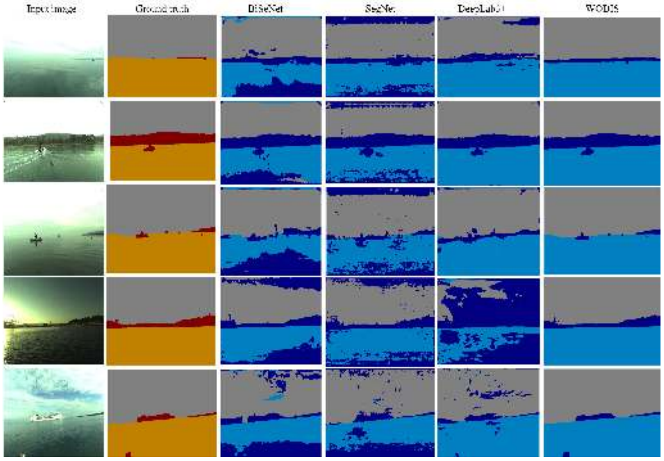

Fig. 11. Visual comparison on the MODD2 Dataset. The MODD2 has rich semantic scenes including water reflection and sunset or sunrise changing scenes. Three typical scenes, i.e. areas with foggy situation, areas with strong sun reflection and areas with water mirroring are selected. BiSeNet and SegNet have a high false positive rate for obstacles detection. WODIS is almost unaffected by external environment changing.

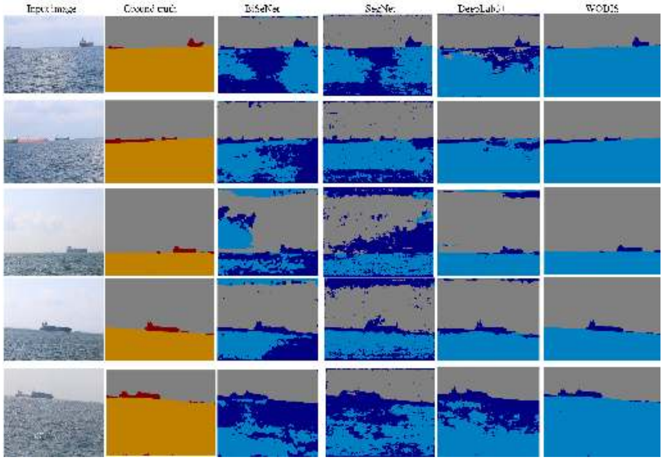

Fig. 12. Visual comparison on the Singapore Maritime Dataset. The SMD has a very pure maritime environment with the large ships being the main obstacles to be detected.

without max pooling; (4) WODIS with ARM. We evaluate the four different comparisons using the MaSTr1325, MODD2, MID and SMD against the mIoU. The result shows that WODIS with ARM (4) has the highest mIoU for image segmentation across four different datasets. Without ARM, the mIoU returns the lowest accuracy of the four experiments. The reason for this is that the ARM can improve the high-level feature map diversity. Comparing with the experiments $WODIS_{NOAVG}(2)$ and $WODIS_{NOMAX}(3)$, it proves that adding global average pooling and max pooling into WODIS in the ARM ($WODIS(4)$) leads to a significant improvement in the segmentation accuracy.

F. Object Detection Results

After getting the segmentation masks from WODIS, the final output for ASVs is to determine the position of obstacles. The bounding box is used to indicate the position of obstacles.

TABLE VI
ABLATION STUDY RESULTS ON MASTR1325 TESTING DATASET.

Architecture	mIoU(%)			
	MaSTr1325	MODD2	MID	SMD
$WODIS_{NOARM}(1)$	69.2	65.3	66.2	67.3
$WODIS_{NOAVG}(2)$	81.9	82.4	83.6	86.5
$WODIS_{NOMAX}(3)$	81.7	77.5	82.1	82.3
$WODIS(4)$	91.3	88.2	88.1	93.7

The processes for getting the bounding box are explained in Algorithm 1 and the results of final bounding boxes are shown in Fig.13, 14, 15, 16, based on four different dataset.

In general, all the bounding boxes can enclose the obstacles based on the binary masks. Apart from certain complex obstacle shapes, as shown in the second figure line in Fig. 13 and Fig.15, respectively, good bounding results can always be obtained for all small object detection after the network provides a pixel-by-pixel segmentation. For more salient objects, reliable detection can be ensured. For example, in Fig.16, ships can be detected accurately and the bounding boxes provide precise enclosure when compared to other three Figures.

Quantitative analysis is also provided here, and in Table VII, precision, recall and F1 values are shown. Based on the testing metrics from the [47], we randomly select 60 images from three dataset to verify the algorithm's detection effectiveness. Since the labelling information is incomplete, we manually get the object bounding box information for the selected 60 images with the tool named LabelMe [48]. Table VII shows that the highest accuracy for the object detection is obtained using the SMD dataset with the F1 score value being around 92.9%. The reason for such a performance is that the sea surrounding in SMD dataset is relative uncluttered thereby less inference is required for segmentation compared with other two dataset. It also should be noted that since the MaSTr1325 is mainly designed for segmentation training for ASVs, limited statistics are available and hence we only present the obstacle detection results and do not make any comparison of MaSTr1325 with the other three datasets. Based on the results in Table VII, this demonstrates that WODIS can have a good object detection result on different datasets.

TABLE VII
RESULTS ON SMD, MODD2 AND MID DATASETS FOR OBJECT DETECTION

Dataset	TP	FP	FN	P[1](%)	R[2](%)	F1 [3](%)
SMD	238	12	24	95.2	90.8	**92.9**
MODD2	300	50	30	85.7	91.3	88.4
MID	60	12	6	83.1	91.2	86.9

[1] 'P' is the abbreviation of precision.
[2] 'R' means the recall value.
[3] 'F1' indicates the F1 score.

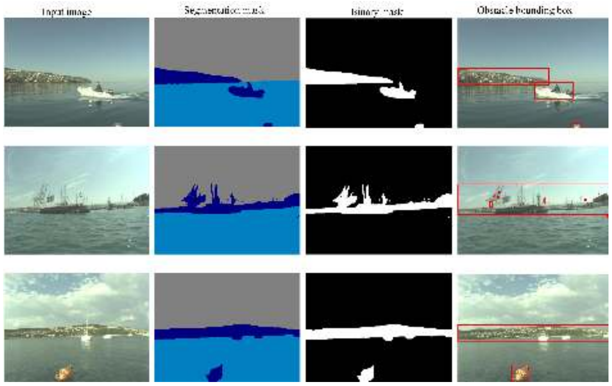

Fig. 13. Obstacle detection result in MaSTr1325 with segmentation mask, binary mask and bounding box.

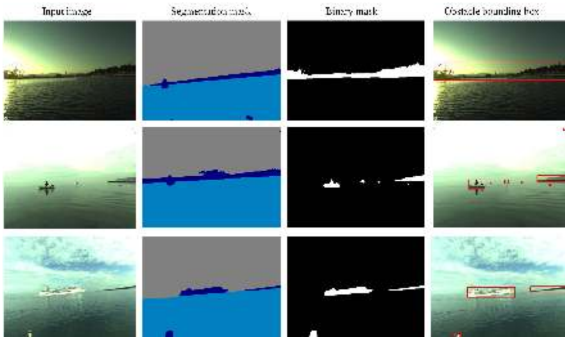

Fig. 14. Obstacle detection result in MODD2 with segmentation mask, binary mask and bounding box.

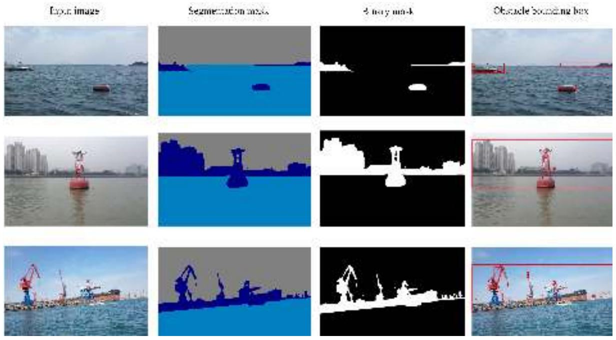

Fig. 15. Obstacle detection result in MID with segmentation mask, binary mask and bounding box.

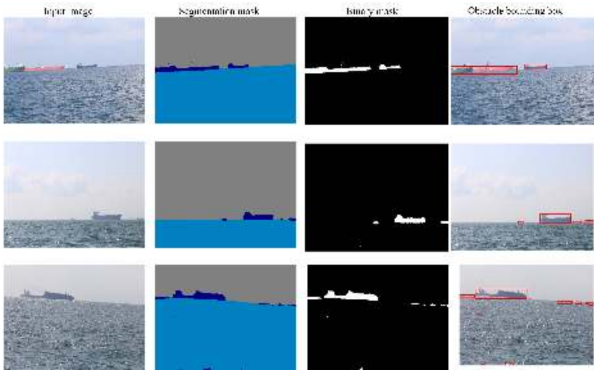

Fig. 16. Obstacle detection result in SMD with segmentation mask, binary mask and bounding box.

V. Conclusion

In this paper, we propose a new water obstacle detection network based on image segmentation (WODIS) to achieve real-time maritime environments semantic segmentation and object detection for ASVs. Compared with other networks, the enhancement to the standard encoder-decoder architecture brought by the WODIS includes to use a lightweight backbone (Xception) to accelerate inference speed and to reuse the backbone in the encoder network to extract the features from different resolutions. The WODIS can make a full use of both high-level and low-level features. In addition, the decoder network of the WODIS combines the feature fusion module to better recover the segmentation masks. Extensive experiments based upon publicly available maritime datasets (MaSTr1325, MODD2, SMD and MID) have been conducted. Detailed analysis and quantitative results demonstrate the effectiveness of the proposed network in various conditions.

For future work, we recommend that further investigations should be carried out from the following aspects:

- Although the WODIS has offered good segmentation results on four datasets, testing on a practical ASV platform in a water environment is necessary. In particular, the deployment of multi-ASV platforms is vital for future ocean missions. Coordinated object detection supported by the coordination of multiple ASVs should be investigated to achieve a more comprehensive environment perception capability.
- Additional experiments under extreme weather conditions, e.g. high waves, heavy rain and fog, are required to ensure that ASVs are able to continue autonomous operation in poor working environments. This could be potentially resolved via the improvement in both hardware (more sensors) and software (adding environment prediction model into networks).
- The investigation of object detection at night can be carried out, by incorporating additional sensors such as infrared cameras to capture images. Proper data fusion algorithms will be adopted for multi-sensor fusion purposes.
- Based on the results presented, it was demonstrated that WODIS can get the obstacle's outline or texture from the background. However, WODIS needs more precise contour for complex obstacle shapes for post processing. Therefore, while keeping the segmentation speed, exploring new ways to retain more precise features of obstacles' contours is a worthy research direction.

Appendix A
Real-time segmentation video testing in Singapore Marine Dataset (SMD)

In order to test the real-time segmentation performance of our model, we have made a test on the Singapore Maritime Dataset (SMD). The length of the video is 10 seconds. All the data and code for this paper are made available: http://github.com/rechardchen123/ASV_Image_Segmentation.

Acknowledgment

The authors acknowledge the use of the UCL Myriad High Performance Computing Facility (Myriad@UCL), and associated support services, in the completion of this work. The authors are also indebted to Konrad Yearwood for his valuable critique and in-depth revision of this paper.

References

[1] A. Felski and K. Zwolak, "The ocean-going autonomous shipchallenges and threats," *Journal of Marine Science and Engineering*, vol. 8, no. 1, p. 41, 2020.

[2] M. Lin and C. Yang, "Ocean observation technologies: A review," *Chinese Journal of Mechanical Engineering*, vol. 33, pp. 1–18, 2020.

[3] O. Levander, "Autonomous ships on the high seas," *IEEE spectrum*, vol. 54, no. 2, pp. 26–31, 2017.

[4] S. C. Mallam, S. Nazir, and A. Sharma, "The human element in future maritime operations–perceived impact of autonomous shipping," *Ergonomics*, vol. 63, no. 3, pp. 334–345, 2020.

[5] R.-R. S. Intelligence, "Autonomous ships. the next step," *Rolls-Royce Marine*, 2016.

[6] C. Fraunhofer *et al.*, "Maritime unmanned navigation through intelligence in networks," *Fraunhofer CML: Hamburg, Germany*, 2016.

[7] Z. Liu, Y. Zhang, X. Yu, and C. Yuan, "Unmanned surface vehicles: An overview of developments and challenges," *Annual Reviews in Control*, vol. 41, pp. 71–93, 2016.

[8] J. Redmon and A. Farhadi, "Yolov3: An incremental improvement," *arXiv preprint arXiv:1804.02767*, 2018.

[9] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask r-cnn," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2961–2969.

[10] J. Wang, J. Steiber, and B. Surampudi, "Autonomous ground vehicle control system for high-speed and safe operation," *International Journal of Vehicle Autonomous Systems*, vol. 7, no. 1-2, pp. 18–35, 2009.

[11] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs," *IEEE transactions on pattern analysis and machine intelligence*, vol. 40, no. 4, pp. 834–848, 2017.

[12] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, "Pyramid scene parsing network," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 2881–2890.

[13] C. Yu, J. Wang, C. Peng, C. Gao, G. Yu, and N. Sang, "Bisenet: Bilateral segmentation network for real-time semantic segmentation," in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 325–341.

[14] H. Li, P. Xiong, H. Fan, and J. Sun, "Dfanet: Deep feature aggregation for real-time semantic segmentation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 9522–9531.

[15] V. Badrinarayanan, A. Kendall, and R. Cipolla, "Segnet: A deep convolutional encoder-decoder architecture for image segmentation," *IEEE transactions on pattern analysis and machine intelligence*, vol. 39, no. 12, pp. 2481–2495, 2017.

[16] L.-C. Chen, G. Papandreou, F. Schroff, and H. Adam, "Rethinking atrous convolution for semantic image segmentation," *arXiv preprint arXiv:1706.05587*, 2017.

[17] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam, "Encoder-decoder with atrous separable convolution for semantic image segmentation," in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 801–818.

[18] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 1251–1258.

[19] S. Mehta, M. Rastegari, A. Caspi, L. Shapiro, and H. Hajishirzi, "Espnet: Efficient spatial pyramid of dilated convolutions for semantic segmentation," in *Proceedings of the european conference on computer vision (ECCV)*, 2018, pp. 552–568.

[20] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[21] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[22] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, "Mobilenets: Efficient convolutional neural networks for mobile vision applications," *arXiv preprint arXiv:1704.04861*, 2017.

[23] P. Chen, Y. Huang, J. Mou, and P. van Gelder, "Ship collision candidate detection method: A velocity obstacle approach," *Ocean Engineering*, vol. 170, pp. 186–198, 2018.

[24] T. Kato, Y. Ninomiya, and I. Masaki, "An obstacle detection method by fusion of radar and motion stereo," *IEEE transactions on intelligent transportation systems*, vol. 3, no. 3, pp. 182–188, 2002.

[25] H. K. Heidarsson and G. S. Sukhatme, "Obstacle detection and avoidance for an autonomous surface vehicle using a profiling sonar," in *2011 IEEE International Conference on Robotics and Automation*. IEEE, 2011, pp. 731–736.

[26] D. Hermann, R. Galeazzi, J. C. Andersen, and M. Blanke, "Smart sensor based obstacle detection for high-speed unmanned surface vehicle," *IFAC-PapersOnLine*, vol. 48, no. 16, pp. 190–197, 2015.

[27] G. Pallotta, M. Vespe, and K. Bryan, "Traffic knowledge discovery from ais data," in *Proceedings of the 16th International Conference on Information Fusion*. IEEE, 2013, pp. 1996–2003.

[28] S. Hexeberg, A. L. Flåten, E. F. Brekke *et al.*, "Ais-based vessel trajectory prediction," in *2017 20th International Conference on Information Fusion (Fusion)*. IEEE, 2017, pp. 1–8.

[29] J. Muhovič, B. Bovcon, M. Kristan, J. Perš *et al.*, "Obstacle tracking for unmanned surface vessels using 3-d point cloud," *IEEE Journal of Oceanic Engineering*, vol. 45, no. 3, pp. 786–798, 2019.

[30] Y. Shan, X. Yao, H. Lin, X. Zou, and K. Huang, "Lidar-based stable navigable region detection for unmanned surface vehicles," *IEEE Transactions on Instrumentation and Measurement*, vol. 70, pp. 1–13, 2021.

[31] D. Yuxing, L. Weining, and W. Shuang, "Study of sea? sky-line detection algorithm based on canny theory," *Computer Measurement & Control*, vol. 18, no. 3, pp. 697–698, 2010.

[32] D. Yongshou, L. Bowen, L. Ligang, J. Jiucai, S. Weifeng, and S. Feng, "Sea-sky-line detection based on local otsu segmentation and hough transform," *Opto-Electronic Engineering*, vol. 45, no. 07, p. 180039, 2018.

[33] M. Kristan, V. S. Kenk, S. Kovačič, and J. Perš, "Fast image-based obstacle detection from unmanned surface vehicles," *IEEE transactions on cybernetics*, vol. 46, no. 3, pp. 641–654, 2015.

[34] B. Bovcon and M. Kristan, "Obstacle detection for usvs by joint stereo-view semantic segmentation," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2018, pp. 5807–5812.

[35] T. Huntsberger, H. Aghazarian, A. Howard, and D. C. Trotz, "Stereo vision–based navigation for autonomous surface vessels," *Journal of Field Robotics*, vol. 28, no. 1, pp. 3–18, 2011.

[36] B. Bovcon, J. Perš, M. Kristan *et al.*, "Stereo obstacle detection for unmanned surface vehicles by imu-assisted semantic segmentation," *Robotics and Autonomous Systems*, vol. 104, pp. 1–13, 2018.

[37] B. Bovcon and M. Kristan, "A water-obstacle separation and refinement network for unmanned surface vehicles," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 9470–9476.

[38] D. K. Prasad, C. K. Prasath, D. Rajan, L. Rachmawati, E. Rajabally, and C. Quek, "Object detection in a maritime environment: Performance evaluation of background subtraction methods," *IEEE Transactions on Intelligent Transportation Systems*, vol. 20, no. 5, pp. 1787–1802, 2018.

[39] B. Bovcon, J. Muhovič, J. Perš, and M. Kristan, "The mastr1325 dataset for training deep usv obstacle detection models," in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2019, pp. 3431–3438.

[40] D. K. Prasad, D. Rajan, L. Rachmawati, E. Rajabally, and C. Quek, "Video processing from electro-optical sensors for object detection and tracking in a maritime environment: a survey," *IEEE Transactions on Intelligent Transportation Systems*, vol. 18, no. 8, pp. 1993–2016, 2017.

[41] J. Liu, H. Li, J. Luo, S. Xie, and Y. Sun, "Efficient obstacle detection based on prior estimation network and spatially constrained mixture model for unmanned surface vehicles," *Journal of Field Robotics*, vol. 38, no. 2, pp. 212–228, 2021.

[42] L. Itti, C. Koch, and E. Niebur, "A model of saliency-based visual attention for rapid scene analysis," *IEEE Transactions on pattern analysis and machine intelligence*, vol. 20, no. 11, pp. 1254–1259, 1998.

[43] H. Larochelle and G. E. Hinton, "Learning to combine foveal glimpses with a third-order boltzmann machine," *Advances in neural information processing systems*, vol. 23, pp. 1243–1251, 2010.

[44] S. Woo, J. Park, J.-Y. Lee, and I. S. Kweon, “Cbam: Convolutional block attention module,” in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 3–19.

[45] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, “Focal loss for dense object detection,” in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2980–2988.

[46] M. Thoma, “A survey of semantic segmentation,” *arXiv preprint arXiv:1602.06541*, 2016.

[47] R. Padilla, W. L. Passos, T. L. Dias, S. L. Netto, and E. A. da Silva, “A comparative analysis of object detection metrics with a companion open-source toolkit,” *Electronics*, vol. 10, no. 3, p. 279, 2021.

[48] K. Wada, “labelme: Image Polygonal Annotation with Python,” https://github.com/wkentaro/labelme, 2016.