


Article

Real-Time Underwater Maritime Object Detection in Side-Scan Sonar Images Based on Transformer-YOLOv5

Yongcan Yu ^{1,2} , Jianhu Zhao ^{1,2,*}, Quanhua Gong ³, Chao Huang ^{1,2}, Gen Zheng ^{1,2} and JinYE Ma ^{1,2}

- ¹ School of Geodesy and Geomatics, Wuhan University, Wuhan 430079, China; 2019202140056@whu.edu.cn (Y.Y.); 2014301610145@whu.edu.cn (C.H.); 2015202140040@whu.edu.cn (G.Z.); 2017301610331@whu.edu.cn (J.M.)
- ² Institute of Marine Science and Technology, Wuhan University, Wuhan 430079, China
- ³ New Energy Engineering Limited Company of China Communications Construction Company Third Harbor Engineering Limited Company, Shanghai 200137, China; 13917099537@ccccltd.cn
- * Correspondence: jhzhao@whu.edu.cn

Abstract: To overcome the shortcomings of the traditional manual detection of underwater targets in side-scan sonar (SSS) images, a real-time automatic target recognition (ATR) method is proposed in this paper. This method consists of image preprocessing, sampling, ATR by integration of the transformer module and YOLOv5s (that is, TR-YOLOv5s), and target localization. By considering the target-sparse and feature-barren characteristics of SSS images, a novel TR-YOLOv5s network and a down-sampling principle are put forward, and the attention mechanism is introduced in the method to meet the requirements of accuracy and efficiency for underwater target recognition. Experiments verified the proposed method achieved 85.6% mean average precision (mAP) and 87.8% macro-F₂ score, and brought 12.5% and 10.6% gains compared with the YOLOv5s network trained from scratch, and had the real-time recognition speed of about 0.068 s per image.



Citation: Yu, Y.; Zhao, J.; Gong, Q.; Huang, C.; Zheng, G.; Ma, J. Real-Time Underwater Maritime Object Detection in Side-Scan Sonar Images Based on Transformer-YOLOv5. *Remote Sens.* **2021**, *13*, 3555. <https://doi.org/10.3390/rs13183555>

Academic Editor: Alexandre Baussard

Received: 29 June 2021

Accepted: 3 September 2021

Published: 7 September 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2021 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/>).

Keywords: sonar automatic target recognition (ATR); real time; underwater maritime object; deep learning; side-scan sonar images

1. Introduction

The rapid development of the marine economy and shipping business has put forward higher requirements for maritime safety. Underwater maritime targets, such as shipwrecks, submerged containers, etc., will bring a threat to navigation safety, and affect the efficiency of water transportation. Furthermore, oil and sewage from the shipwreck, as well as chemicals in the container, can cause heavy marine pollution once leaked. Therefore, it is essential to detect the underwater maritime targets fast and accurately.

Side-scan sonar (SSS) can provide high-resolution images [1–3], which is extensively used in underwater object detection [4] and maritime search and rescue (SAR) [5]. Until recently, object detection from SSS images has mainly relied on manual visual interpretation [6], and the detection result is thereby influenced by personal quality and experience. Many scholars have studied the automatic target recognition (ATR) from SSS images, such as the machine learning (ML) method and deep learning (DL) method. The ML-based methods focus on unsupervised object detection based on artificial features [7] such as texture [8,9], corners and edges [10] with Independent Component Analysis (ICA) [11]. The traditional ML-based methods usually aim at a specific category of targets. Therefore, recognizing each type of target requires finding the optimal features manually, which will lead to feature engineering problems [12]. In addition, the found optimum feature extractor and the classifier may not be compatible because the ML method is a process of separate optimization, which leads to error accumulation.

Deep learning (DL) is an end-to-end learning algorithm that extracts and selects features automatically. When the DL method is used for ATR from SSS images, two

problems have to be faced, insufficient SSS target images for training the recognition network and the incomplete applicability of the existing detection networks designed for optical images. To get sufficient SSS target images, Nguyen et al. increased the SSS target images by scattering, polarization, and geometric transformation and achieved 91.6% accuracy in submerged human body detection [13]. Additionally, the Generative Adversarial Network (GAN) was also introduced to synthesize SSS target images [14,15] and improved the style transferring [16]. On the other hand, to adapt the detection networks into SSS images, Kim et al. studied the applicability of convolutional neural networks (CNN) for synthetic underwater object detection [17]. Einsidler et al. introduced the You Only Look Once (YOLO) algorithm for ATR from Autonomous Underwater Vehicles (AUVs) [18]. Some scholars have also worked to improve the recognition accuracy by combining the artificial features [19] and transfer learning [20,21].

In the real-time underwater target recognition, since the detected SSS images or waterfall images are original and not processed, the implementation strategy and detection speed need to be taken into consideration besides recognition precision. Currently, the studies on real-time ATR mainly focus on SSS image segmentation. Song et al. provided a real-time SSS image segmentation method based on self-cascaded CNN [22]. Wu et al. proposed an encoder–decoder real-time SSS image semantic segmentation model based on the depth-wise separable convolution and achieved 66.2% mean Intersection over Union (mIOU) [23]. Burguera et al. proposed an on-line SSS image segmentation method based on the fully convolutional neural network and a temporal window method for real-time sampling and achieved an 87.8% F_1 score on AUV [24]. However, the segmentation merely provides the coarse-grained classes, such as the highlight and shadow of objects and the seafloor instead of the certain target class.

The underwater targets, such as shipwrecks and submarine containers, are usually detected based on the target information such as shapes, sizes and textures. However, the SSS images are always target-sparse and feature-barren with complex noise, and the distortion of a target is caused by the difference of resolution in different directions, especially for real-time detection. These problems will affect the real-time detection accuracy. To implement the real-time target recognition in SSS images efficiently, the following works have been done in the paper.

1. A real-time SSS ATR method is proposed, including preprocessing, sampling, target recognition by TR–YOLOv5s and target localization;
2. To deal with the target-sparse and feature-barren characteristics of SSS image, the attention mechanism is introduced by improving the state-of-the-art (SOTA) object detection algorithm YOLOv5s with transformer module;
3. A down-sampling principle is proposed for the echoes of cross-track direction to maintain the actual aspect ratio of target roughly and reduce the amount of calculation for real-time recognition.

The paper is structured as follows: Section 2 details the proposed real-time SSS ATR method, including the SSS image preprocessing, the sampling, the recognition model based on the proposed TR–YOLOv5s and the target localization. Section 3 verifies the proposed method by experiments. Section 4 discusses the superiorities and limitations of the proposed method in the real-time ATR. Finally, some conclusions are drawn out.

2. Materials and Methods

The flow chart of the proposed real-time SSS ATR method is shown in Figure 1, which mainly consists of sonar image preprocessing, sampling, automatic target recognition by TR–YOLOv5s and target localization.

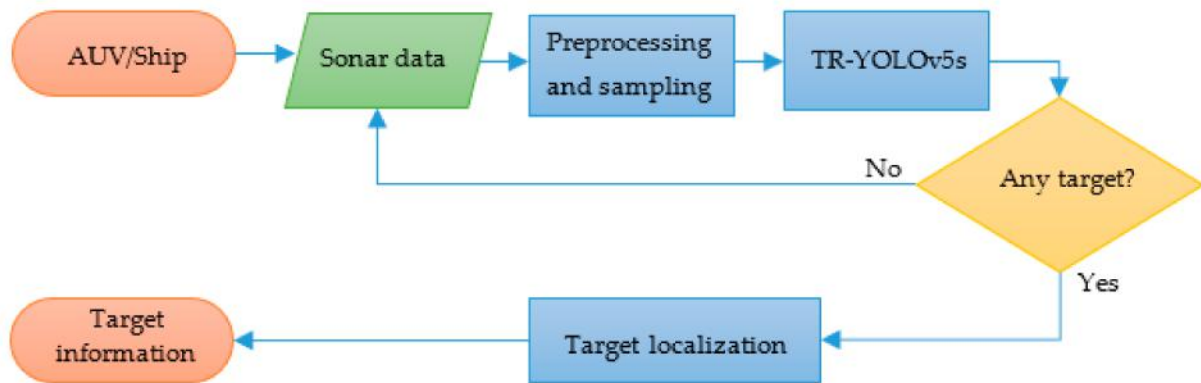


Figure 1. The flow chart of the proposed real-time SSS ATR method.

2.1. SSS Image Preprocessing and Sampling

2.1.1. Image Preprocessing

The real-time SSS image or waterfall image is updated continuously with newly received acoustic signals. These signals can be quantified into the 11-bit or 64-bit range. To reduce the cost of subsequent computer processing and storage, and improve the image contrast, the data needs to be re-quantized to 0–255 (8-bit) by:

$$gl = \ln \left(1 + \frac{2^n \times rl}{2^m} \right), \quad (1)$$

where gl is the quantified value corresponding to the bit number n as 8, and rl is the original recorded value corresponding to the bit number m .

2.1.2. SSS Image Sampling

Different from the real-time segmentation in literature [22], object detection requires the dimension, geometric shape and spatial information of underwater targets from pings for localization and classification. Therefore, it is necessary to set a common coverage area between two adjacent samples to prevent the underwater target from being divided into two images. To achieve this, the real-time underwater target detection requires intensive sampling in the direction along the track, which means that detection needs to be performed for each ping update.

Firstly, a dense sampling in the along-track direction is performed as shown in Figure 2. The SSS image is constantly updated as the survey ship sails, and the various colors of boxes in Figure 2 denote the sampling area. The sampling width is the same as the scanning width of SSS, and the sampling height d should be longer than the target length in the along-track direction, which is set to the same as the standard input to meet the detection algorithm requirements. The sampling interval in the along-track direction is equal to the ping sampling interval of SSS.

In SSS measurement, the difference between the cross-track resolution M_{ct} , the actual distance between two adjacent pixels in the cross-track direction, and the along-track resolution M_{at} , the actual distance between two adjacent pixels in the along-track direction, will result in the shape distortion of the underwater target as shown in Figure 3.

Therefore, the height of the grids in Figure 3b is longer than the width, which means the target is stretched and that will influence the detection precision due to the loss of target information.

To maintain the actual aspect ratio of the target roughly and reduce the amount of calculation for real-time recognition, a down sampling method is proposed and applied in the cross-track direction, and the sampling multiple n is given as follows:

$$n = \frac{M_{at}}{M_{ct}} = \frac{v \times T}{L/N}, \quad (2)$$

where v and T refer to the ship speed and the ping sampling interval respectively, and the L and N denote the scanning width of SSS and sampling numbers in the cross-track direction respectively.

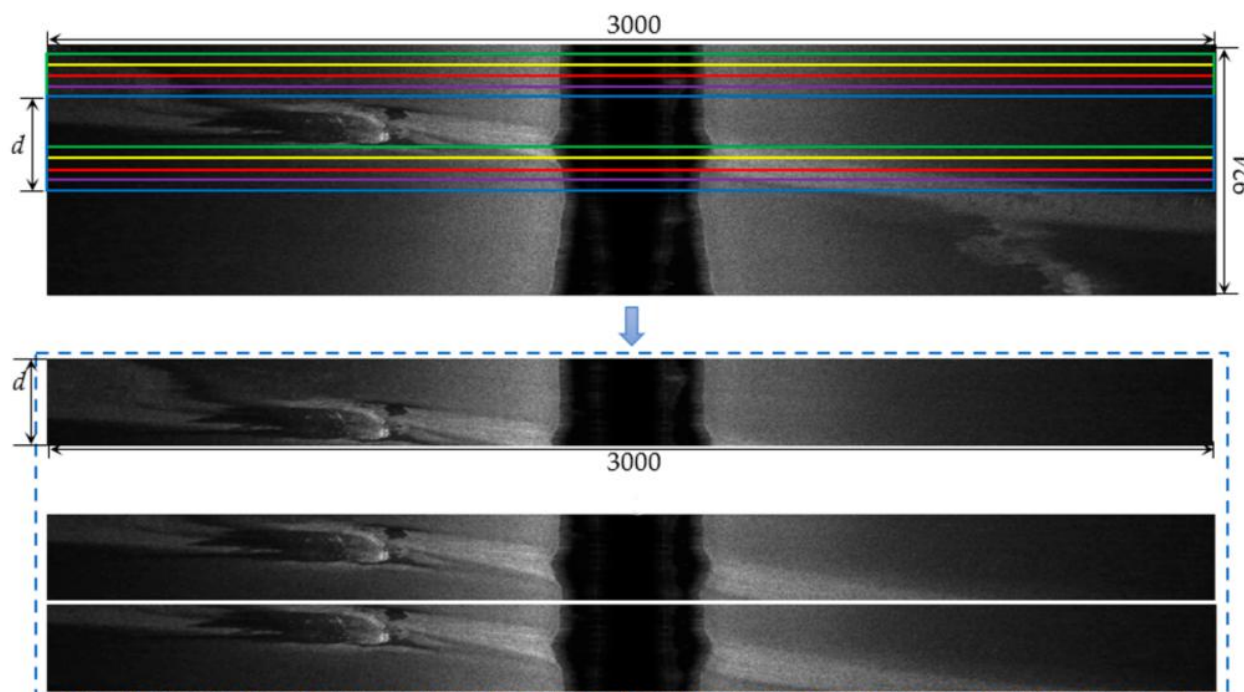


Figure 2. Dense sampling in the along-track direction. The various colors of boxes in the above image denote the sampling operation of which the width is equal to the scanning width of SSS divided by the cross-track resolution and the height is d pixels. The images in the blue dashed line are the sampling results.

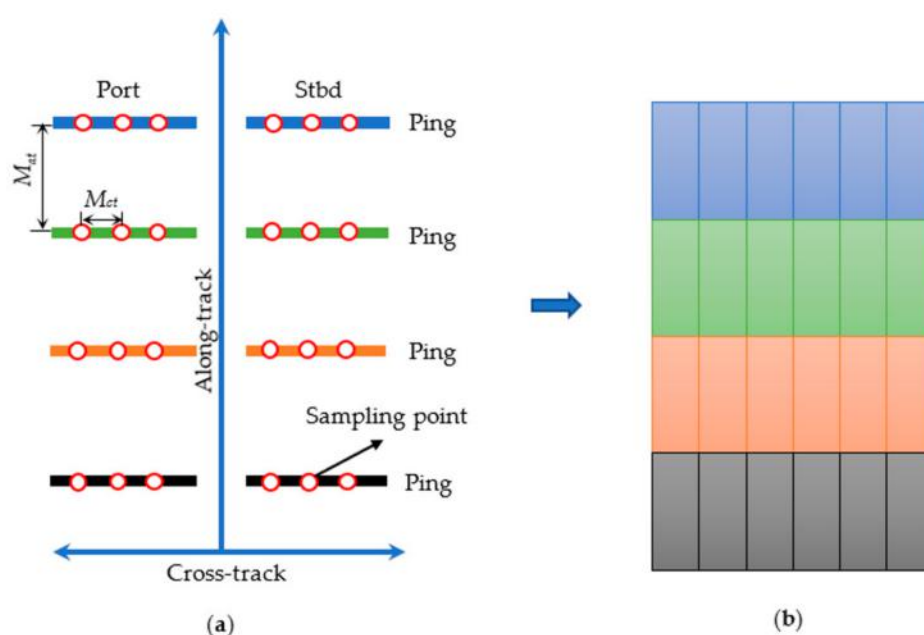


Figure 3. The distortion caused by the difference of mapping resolution in the along-track direction and cross-track direction: (a) is the schematic diagram of SSS measurement; (b) is the image generated from (a). The thick lines of four colors in (a) denote the scanning lines of SSS. M_{at} and M_{ct} denote the mapping resolutions in the along-track direction and cross-track direction, which are equal to the height and width of the grids in (b).

According to Equation (2), the raw SSS image pixels in the cross-track direction are reduced by a factor of 5 and the along-track remains the same, and the result is shown in Figure 4b. The shipwreck in the SSS image shows a more realistic shape because the resolutions in the cross-track direction and along-track direction are more similar.

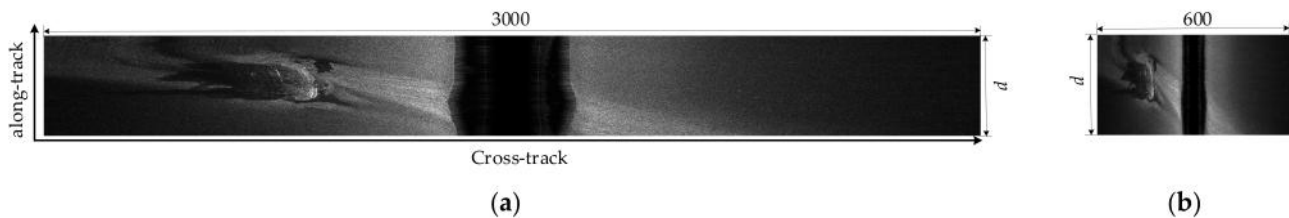


Figure 4. Raw image and the resampled image. (a) Raw SSS image of which the mapping resolution is 0.5 m in the along-track direction and 0.1 m in the cross-track direction. It means that the actual distance between two adjacent pixels in the along-track is 0.5 m and 0.1 m in the cross-track direction. (b) The sampled SSS image of which the mapping resolution is 0.1 m both in the along-track and cross-track direction.

Finally, for one sampling, the patches with the size of $d \times d$ pixels are generated by a sliding window of $d \times d$ pixels from left to right along the cross-track direction. To obtain the accurate profile and position by image segmentation or other post-processing, every two adjacent patches, for example, P1 and P2 in Figure 5 have a common coverage area of size 40% of patch size to prevent the target from being divided into two parts (Figure 5).

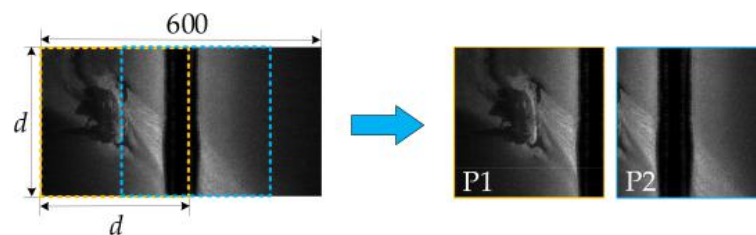


Figure 5. The schematic diagram of patches generation by sliding operation in a sampling. The orange and blue dashed boxes denote two adjacent patches, P1 and P2, with the size of $d \times d$ pixels.

2.2. TR-YOLOv5s

2.2.1. Architecture of TR-YOLOv5s

The key to the widespread use of real-time underwater target detection is to find a suitable detection algorithm that balances speed and accuracy, and which should also have the lowest possible computational requirements due to the slow replacement of deck units and the limited size of the AUV on board. YOLOv5 [25] is the SOTA object detection algorithm with fast detection speed and exact precision, which gets 72% AP@0.5 for COCO val2017 dataset. Besides, the minimum model size of YOLOv5s is only 14 megabytes, which is convenient for deployment [26]. However, YOLOv5s is constructed with an optical sample set which is not fully applicable to SSS images. In addition, the number of underwater target samples is much smaller than the optical samples. Using YOLOv5s for real-time detection alone does not satisfy the demand, and we need to improve YOLO by combining the specific differences between side-scan sonar images and optical images. Compared with optical images, sonar images have sparse features due to the resolution limitation, and sonar images often have sparse features around the target because the sea is too vast. Therefore, this paper specifically adds a transformer module to YOLOv5s to focus on the target itself features and ignore the surrounding features of the target, i.e., the attention mechanism is introduced into YOLOv5s, and an improved algorithm TR-YOLOv5s is proposed for real-time object detection as shown in Figure 6.

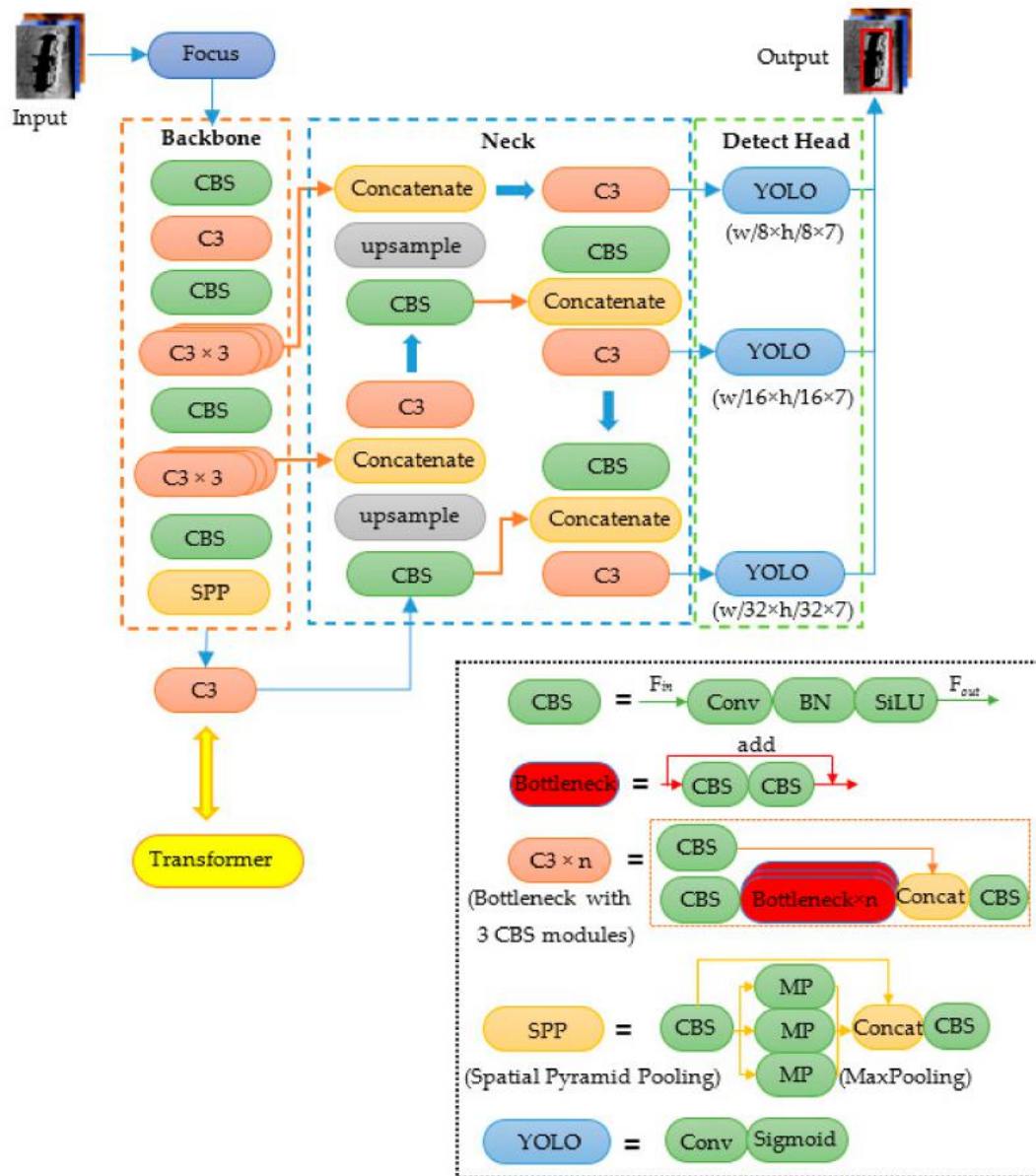


Figure 6. The architecture of TR-YOLOv5s including the raw YOLOv5s and the transformer module marked in yellow.

In Figure 6, the CBS module is the basic module in TR-YOLOv5s, which consists of convolution (Conv) layer, batch normalization (BN) [27] layer and activation layer by using a Sigmoid Weighted Liner Unit (SiLU) [28].

The Conv layer is calculated as follows:

The input feature map $F_{in} \in \mathbb{R}^{H \times W \times C_{in}}$ is a tensor of size $H \times W \times C_{in}$, where the H, W and C_{in} denote the height, width and channels of input feature map, respectively, and define a cohort of convolution kernels $K \in \mathbb{R}^{k \times k \times C_{in} \times C_{out}}$, tensors with a fixed size of $k \times k \times C_{in}$, of which the number of convolution kernels C_{out} decides the channels of output feature map $F_{out} \in \mathbb{R}^{H \times W \times C_{out}}$. Based on the above definition, the equation of convolution operation is given as

$$F_{out}^{x,y,z} = \sum_{c=1}^{C_{in}} \sum_{i=\frac{1-k}{2}}^{\frac{k-1}{2}} \sum_{j=\frac{1-k}{2}}^{\frac{k-1}{2}} K^{i+\frac{k-1}{2}, j+\frac{k-1}{2}, c, z} \times F_{in}^{i+x, j+y, c}, \quad (3)$$

where the x and y denote the pixel in the output feature map located in (x, y) in channel z .

The BN layer is set for regularization and improving the generalization ability of the neural network. By batch normalization operation, the distribution of input data in each channel will be compacted between -1 and 1 , obeying a Gaussian distribution with a mean of 0 and a variance of 1 . That will put most of the activation values into the linear region of the activation function, and its corresponding derivatives away from the saturated region to accelerate the training convergence process. The equation of batch normalization is given as

$$F_{out}^{x,y,c} = \frac{F_{in}^{x,y,c} - E(F_{in}^c)}{\sqrt{\text{Var}(F_{in}^c) + \varepsilon}} \times \gamma + \beta, \quad (4)$$

where $E(F_{in}^c)$ refers to the mean value of pixels of input feature maps in channel c in the same batch, and the $\text{Var}(F_{in}^c)$ denotes the variance likewise.

$$E(F_{in}^c) = \frac{1}{n} \times \sum_{n=1}^N \sum_{i=1}^W \sum_{j=1}^H F_n^{i,j,c}$$

$$\text{Var}(F_{in}^c) = \frac{1}{n-1} \times \sum_{n=1}^N \sum_{i=1}^W \sum_{j=1}^H [F_n^{i,j,c} - E(F_{in}^c)]^2, \quad (5)$$

where the $F_n^{i,j,c}$ refers to the pixel of the n th feature map in the input batch, which is located in (x, y) in channel z .

The parameter ε is the stability factor, and the parameter γ and β are the scale factor and the shift factor respectively which can be learned and updated in the training process.

The activation layer uses SiLU, an activation function that can operate a nonlinear transformation on the tensor from batch normalization to enhance useful information and suppress useless information.

The calculation principle of SiLU is given as:

$$\text{SiLu}(x) = x \times \frac{1}{1 + e^{-x}}, \quad (6)$$

where x refers to the input feature map pixels.

As shown in Figure 6, the proposed TR-YOLOv5s includes Backbone, Neck and Detect Head. A slice operation is performed for the input image firstly by Focus Module to transfer the spatial information to the channel dimension without information loss (Figure 7). Then, a 3×3 convolution is performed to change the number of channels of the feature map.

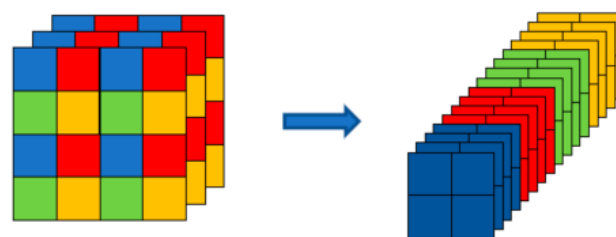


Figure 7. Focus Module. A 4×4 input image is divided into four 2×2 slices and the pixels in the same slice have a similar relative position in the input image.

- Backbone

The raw Backbone of YOLOv5s is CSPDarknet53, surrounded by the orange dashed line in Figure 6, and it is used for feature extraction and selection by Cross Stage Partial (CSP) [29] structure which consists of a Bottleneck structure and three convolutions (C3). When the feature map F_{in} from previous layer inputs, it will be calculated into two parallel branches as shown in Figure 8, and the number of channels of F_{in} will be reduced by half

to generate two new feature maps. Then, the two feature maps are concatenated as the output F_{out} of the C3 module.

- Neck

The Neck part, surrounded by the blue dashed line in Figure 6, is a combination of a Feature Pyramid Network (FPN) [30] and a Path Aggregation Network (PAN) [31]. By enhancing the shallow features of a smaller receptive field, the combination of FPN and PAN will improve the detection precision.

- The Detect Head

The Detect Head is used to generate the position and classification information of the predicted targets in SSS imagery. In this part, the multi-scale detection head of YOLOv3 [32] is used to output detection results in three aspect ratios for large, medium and small targets respectively. Every YOLO module in Figure 6 consists of a convolution layer (Conv) and an activation layer by using the sigmoid function.

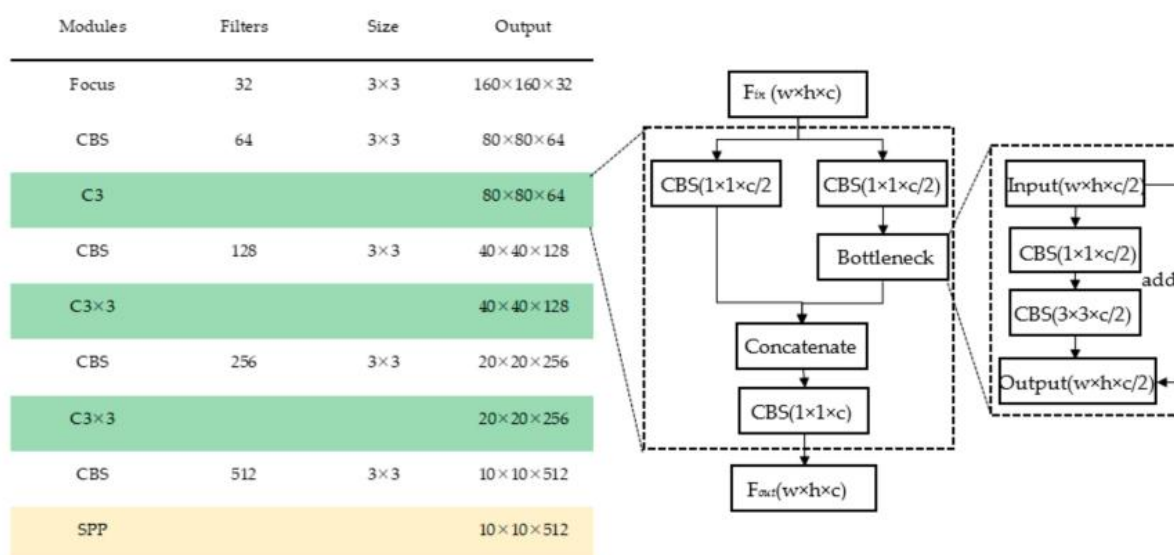


Figure 8. The detailed structure of Backbone, including the number of filters, the filters size and the feature map shape of each layer.

2.2.2. Transformer Module

Usually, the SSS images are target-sparse and feature-barren. Compared with the open-source optical dataset, the SSS images have obvious differences in the small relative size of the underwater target to the SSS image and the less occupancy rate of the target in the whole detected image. To weaken the effects on the recognition, a transformer module [33] is introduced to calculate the self-attention of all pixels before the target detection so as to pay more attention to the targets instead of the background or blank areas.

The Transformer is a full-attention model and is made up of a multi-head self-attention (MHSA) module as shown in Figure 9. When used for SSS imagery, it can calculate the relationship among pixels in different positions to enhance the key information especially for the target from multi subspaces. In fact, each head of self-attention can be viewed as a subspace of feature and information.

To satisfy the calculation requirement of the transformer module, the input feature map from Backbone will be reshaped to vector form by flattening the spatial dimension as shown in Figure 10.

To reduce the loss of position information induced by the reshape operation, a positional encoding operation by using a linear layer is added. Then, for the self-attention, the query vector Q , the key vector K and the value vector V all come from the same input vector

I by different linear transformations. MHSA concatenates several heads of self-attention calculated from I , called multi heads, and then performs a linear transformation, which is given as

$$\text{MultiHead}(Q, K, V) = \text{Concat}(\text{head}_1, \text{head}_2, \dots, \text{head}_h) W^{out}, \quad (7)$$

where Concat refers to the tensor concatenation operation, and W^{out} is a matrix for linear transformation. The head_h denotes the h th self-attention result obtained by using scaled dot-product attention which is given as:

$$\text{head}_h = \text{Attention}(Q_h, K_h, V_h) = \text{softmax}\left(\frac{Q_h \cdot K_h^T}{\sqrt{d_k}}\right) V_h, \quad (8)$$

where d_k is the variance of the dot product of Q_h and K_h to alleviate the gradient vanishing problem of softmax. The vector Q_h , K_h and V_h are given as:

$$Q_h = I W_h^Q, K_h = I W_h^K, V_h = I W_h^V, \quad (9)$$

where I is the input vector, W_h^Q is the linear transformation from I to Q of h th head.

Finally, two linear transformations after the MHSA operation are operated as shown in Figure 9, and a feature map with more attention to targets is obtained after the reshape operation.

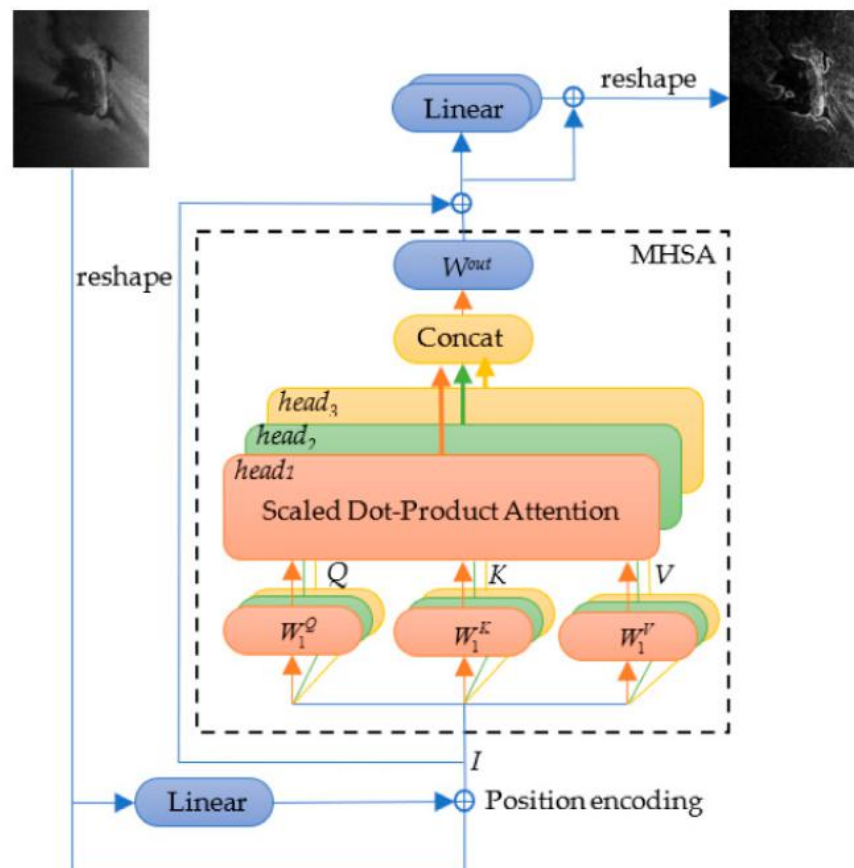


Figure 9. Schematic diagram of transformer module.

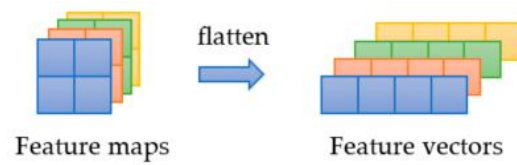


Figure 10. Reshape operation for an input feature map of the transformer module.

2.3. Target Localization

To facilitate the subsequent salvage and dredging, it is necessary to obtain the geographic locations of the detected underwater targets. In SSS measurement, the coordinates of tow-fish or SSS transducer are calculated according to the location of the shipboard GNSS antenna, the level arms of the GNSS antenna and the tow point in the vessel coordinate system, the orientation of surveying ship and the horizontal length of tow cable (Figure 11). It should be noted that during the measurement process, changes in both the direction and speed of the vessel will lead to changes in the spatial position of the towed fish in relation to the vessel, which, together with the effects of tidal currents and inaccurate measurements of the angle and length of the cable, will result in inaccurate positioning results. However, the target positioning is only to obtain an approximate geographical position to guide the subsequent work of the maritime department. Therefore, the error caused by the above is acceptable.

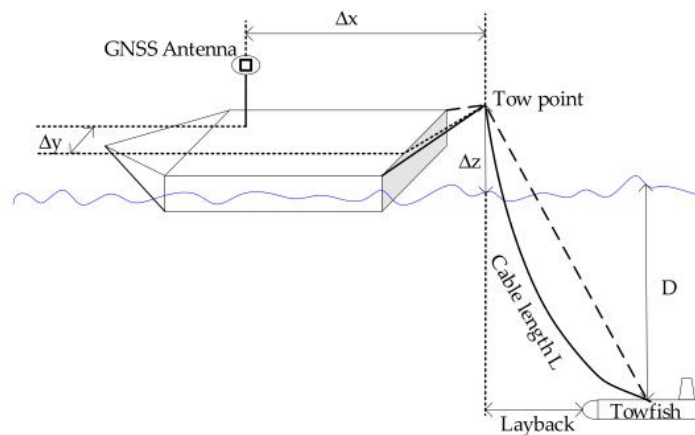


Figure 11. Schematic diagram for calculating the geographic coordinates of the tow-fish.

Assuming that the sailing orientation of tow-fish is also the orientation of surveying ship, the geographic coordinates of tow-fish $(X_T, Y_T)_{GRF}$ are calculated by:

$$\begin{pmatrix} X_T \\ Y_T \end{pmatrix}_{GRF} = \begin{pmatrix} X_{GNSS} \\ Y_{GNSS} \end{pmatrix}_{GRF} + \mathbf{R}(A) \begin{pmatrix} \Delta x + Layback \\ \Delta y \end{pmatrix}_{VFS-T-GNSS}, \quad (10)$$

where, $(X_{GNSS}, Y_{GNSS})_{GRF}$ are the coordinates of GNSS antenna in the geographic coordinate system (GCS), Δx and Δy refer to the coordinate offset between the tow point and GNSS antenna in the ship coordinate system which is given as:

$$\begin{pmatrix} \Delta x \\ \Delta y \end{pmatrix}_{VFS-T-GNSS} = \begin{pmatrix} x_T \\ y_T \end{pmatrix}_{VFS} - \begin{pmatrix} x_{GNSS} \\ y_{GNSS} \end{pmatrix}_{VFS}, \quad (11)$$

and *Layback* is the horizontal length of tow cable which is given as:

$$Layback = \sqrt{(\lambda \times L)^2 - (\Delta z + D)^2}, \quad (12)$$

where L , D and Δz denote the length of tow cable, the depth of tow-fish and the vertical distance between the tow point and water surface, and λ is the coefficient to convert the

cable length into the hypotenuse, which is usually set to 0.9. $\mathbf{R}(A)$ is the rotation matrix formed with the ship orientation A .

$$\mathbf{R}(A) = \begin{pmatrix} \cos A & -\sin A \\ \sin A & \cos A \end{pmatrix}, \quad (13)$$

Then, the location of the detected target in GCS can be obtained by:

$$\begin{pmatrix} X_{tg} \\ Y_{tg} \end{pmatrix}_{GRF} = \begin{pmatrix} X_T \\ Y_T \end{pmatrix}_{GRF} + \begin{pmatrix} \sqrt{S^2 - D^2} \cos(A \pm \frac{\pi}{2}) \\ \sqrt{S^2 - D^2} \sin(A \pm \frac{\pi}{2}) \end{pmatrix}_{GRF}, \quad (14)$$

where S denotes the slant range between the detected target and the tow-fish. Assume that the seabed is flat relative to the boat and sea surface, S is given as:

$$S = \frac{T \times v}{2}, \quad (15)$$

where v and T denote the sound speed and the time of two-way propagation of sound wave from SSS transducer to the detected target respectively.

2.4. Evaluation of the Recognition Model

The evaluation indexes of model performance used here are mean average precision (mAP), F_β score and floating point operations (FLOPs), which reflect the performance of the model in accuracy and complexity respectively.

The mAP can be intuitively reflected by the area under curve (AUC) of P - R curve. To obtain the mAP, it is necessary to calculate the AP value of each class of underwater targets at a fixed Intersection over Union (IOU) threshold firstly, and then calculate the mean value of AP values of all classes. Wherein, when calculating the AP value of a certain class of target, the Precision (P) and Recall (R) should be obtained first as:

$$\begin{aligned} P &= TP / (TP + FP) \\ R &= TP / (TP + FN) \end{aligned} \quad (16)$$

Taking the shipwreck target as an example, TP , True Positive, means the content in the bounding box predicted by the model is the shipwreck and the result is correct; FP , False Positive, means that the content in the predicted position is a shipwreck but it is not in fact; TN , True Negative, means that non-shipwrecks are predicted by the model and there are not shipwrecks in reality; FN , False Negative, means that the model missed the target.

Different recall values can be obtained by setting different confidence thresholds, and the corresponding precision values can be calculated. Then, taking precision as the longitudinal axis and the recall value as the horizontal axis to draw the P - R curve, the AP value can be approximated by the area under the P - R curve.

$$AP = \sum_{k=2}^{n-1} (R_k - R_{k-1}) P_k, \quad (17)$$

Both the precision and the recall values are required at the same time, but the recall is more important than the precision. F_β score also reflects the relationship of P and R by:

$$\frac{1}{F_\beta} = \frac{1}{1 + \beta^2} \times \left(\frac{1}{P} + \frac{\beta^2}{R} \right), \quad (18)$$

where β is a constant to change the importance of precision and recall, and it is set to 2 to enhance the recall. More importantly, for multi-class target recognition, we need to calculate the macro- F_β based on the macro- P and macro- R , which are the mean value of P and R respectively in different confidence thresholds.

Besides, FLOPs of convolution are given as:

$$\text{FLOPs} = 2 \times H \times W \times (C_{in} \times K^2 + 1) \times C_{out}, \quad (19)$$

where H , W and C_{in} are the height, width and channel of the input feature map respectively, and K and C_{out} are the size of convolution kernel and channel of the output feature map.

3. Results

3.1. Dataset

To verify the proposed real-time SSS ATR method, an SSS image set consisting of shipwrecks and submarine container targets was collected, amplified and divided into the training set, validation set and test set, and then used for the detector building.

Besides, the real-time underwater target recognition experiments were carried out with three sets of SSS data including two shipwreck data as shown in Section 3.1.2 and one submarine container data as shown in Section 3.1.3.

3.1.1. SSS Image Set for Detector Building

We collected two SSS image sets, A and B, for detector training and testing. Dataset A mainly includes the images from Google retrieval that have been post-processed such as radiometric correction, slant range correction, etc., and dataset B mainly includes the original grayscale images without any optimization. Some samples of dataset A and B are shown in Figure 12.

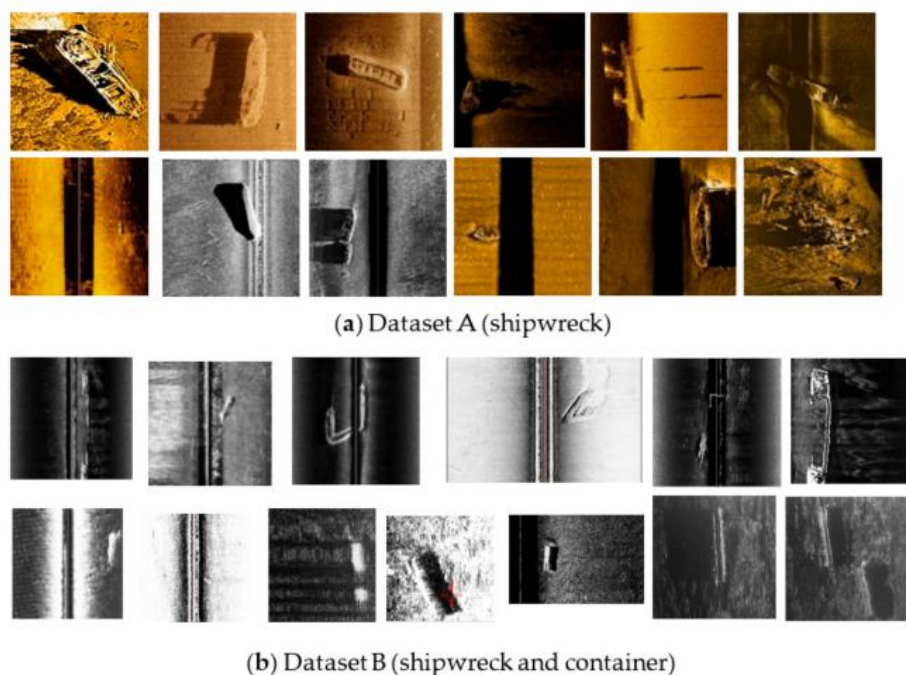


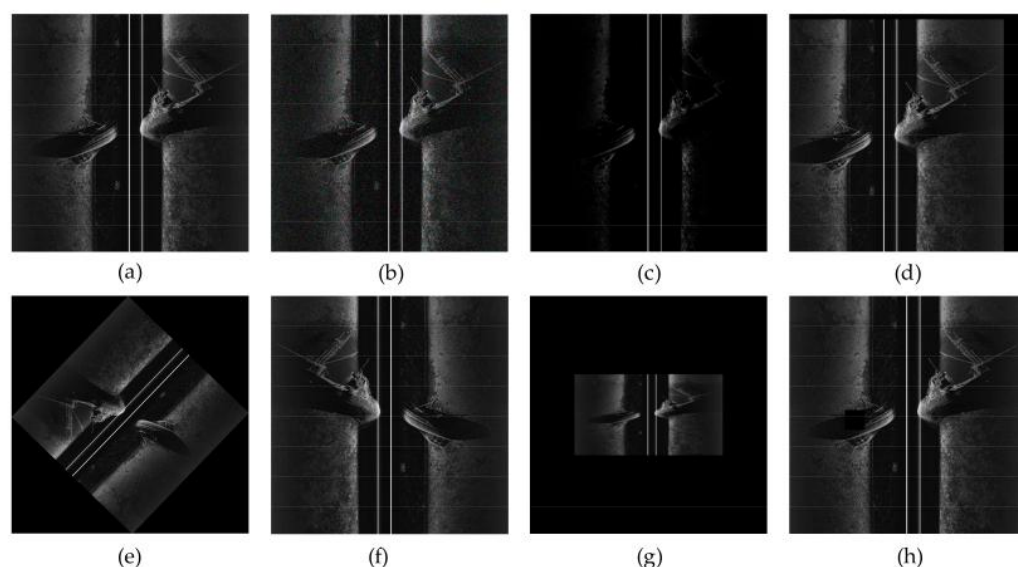
Figure 12. Some samples of data set A and data set B: (a) denotes the data set A which only consists of shipwreck targets; (b) denotes the data set B. The images in the first row of (b) are shipwreck targets and the second row are container targets.

Dataset A consists of 313 shipwreck images and dataset B consists of 65 shipwreck images and 97 container images. To maintain the original distribution of the data, we mixed up the images with the same class of targets at first and then divided them into three subsets, namely the training set, the validation set and the test set by a ratio of 6:2:2 (Table 1).

Table 1. The training set, the validation set and the test set.

Target \ Data Set	Training Set			Validation Set			Test Set		
	A	B	Total	A	B	Total	A	B	Total
Shipwreck	194	34	228	56	20	76	63	11	74
Container	0	58	58	0	19	19	0	20	20
Total	194	92	286	56	39	95	63	31	94

The deep learning algorithms are data-driven, so the network performance is easily affected by the amount and distribution of the training dataset. For the underwater target recognition model training, there were two problems: the lack of enough images of shipwrecks and containers and the quantity unbalance in different targets will limit the performance of deep learning algorithms. To solve the two problems, the data augmentation technique was introduced to generate more SSS images with shipwrecks and containers. The data augmentation algorithms, such as the Gaussian noise, the brightness change, the image translation, the rotation, the flip, the zoom and the cutout, etc. are commonly used [13]. Figure 13 shows the results achieved by different data augmentation algorithms.

**Figure 13.** Data augmentation. (a) denotes the raw SSS image; (b–h) are the results of data augmentation by the Gaussian noise, the brightness change, the image translation, the rotation, the flip, the zoom and the cutout.

By randomly combining several data augmentation methods, the training set and the validation set were amplified and the ratio between shipwrecks and containers was balanced (Table 2). The test set was not amplified because the purpose of data augmentation is merely for detector training and hyper-parameters adjustment, and once the data augmentation was used in the test set, the distribution of the test set would be changed and be revealed to some degree which was supposed to be invisible in the training period.

Table 2. The division of the training set, validation set and test set after data augmentation.

Target \ Data Set	Training Set			Validation Set			Test Set		
	A	B	Total	A	B	Total	A	B	Total
Shipwreck	970	170	1140	280	100	380	63	11	74
Container	0	1160	1160	0	380	380	0	20	20
Total	970	1330	2300	280	480	760	63	31	94

3.1.2. Description of Shipwreck Data

The first shipwreck data was collected at the Bohai Sea, China. In this experiment, the SSS of Edgetech 4200-MP was utilized to obtain SSS waterfall images and find the shipwrecks in real-time. The location of the measurement area and the sailing routes are shown in Figure 14.

The other shipwreck data was collected in the Wusong River, China. There are three shipwrecks in this area. An SSS of Shark-S150D was utilized to obtain SSS waterfall images and find the shipwrecks in real-time, the measurement area and the sailing routes are shown in Figure 15.

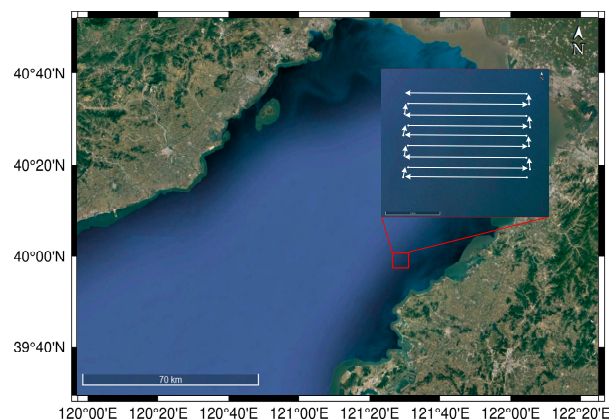


Figure 14. The location of the experiment and the sailing routes of SSS measurement.



Figure 15. The location of the experiment and SSS sailing routes.

3.1.3. Description of Submarine Container Data

The submarine container data was collected in the Yangtze River, China. There are dozens of containers in this area. The SSS of Benthos SIS-1624 was utilized to observe SSS waterfall images and find the submarine containers in real-time, and the schematic diagram of the sailing routes is shown in Figure 16.

3.2. Detector Building

3.2.1. Detector Training

Experimental training and testing were conducted using Python programming based on Pytorch. The experimental environment is Windows 10 system, running on an AMD Ryzen 7 5800X 8-Core Processor@3.80GHz, with an NVIDIA GeForce RTX 2070 SUPER.

In the training process, the training samples were input into the detector to generate feature maps by Backbone at first. Then, the feature maps were enhanced and fused by Neck for a better representation, and then mapped into the label space from the feature space by the Detect Head. The predicted label vectors had 7 channels, including the position information (4 channels), confidence (1 channel) and class (2 channels due to 2

classes of targets). After that, three loss values [25] were calculated between the predicted label vector and the ground truth to assess the performance of the detector's parameters on predicting the position, confidence and class information of the targets. Finally, the parameters of the detector were updated by using the stochastic gradient descent (SGD) algorithm.

To accelerate the convergence speed and enhance the generalization ability of the proposed TR-YOLOv5s model, the tricks of anchor clustering and transfer learning were used in the detectors training. The anchor sizes of YOLOv5s and TR-YOLOv5s were reset to (21, 21), (29, 71), (53, 52), (45, 121), (95, 94), (73, 225), (223, 77), (148, 152) and (246, 201) by using k-means clustering algorithm based on the bounding boxes in the training set. For TR-YOLOv5s, a pre-trained weighting was used to initialize the detector for better generalization ability.

Some training hyper-parameters were set as: the epochs were 300, the batch size was 16, the momentum of SGD was 0.937 and the initial learning rate was 0.01 which will be changed during training by using a cosine annealing algorithm.

The loss curves and mAP curves are shown in Figure 17.



Figure 16. The schematic diagram of the sailing routes of SSS measurement.

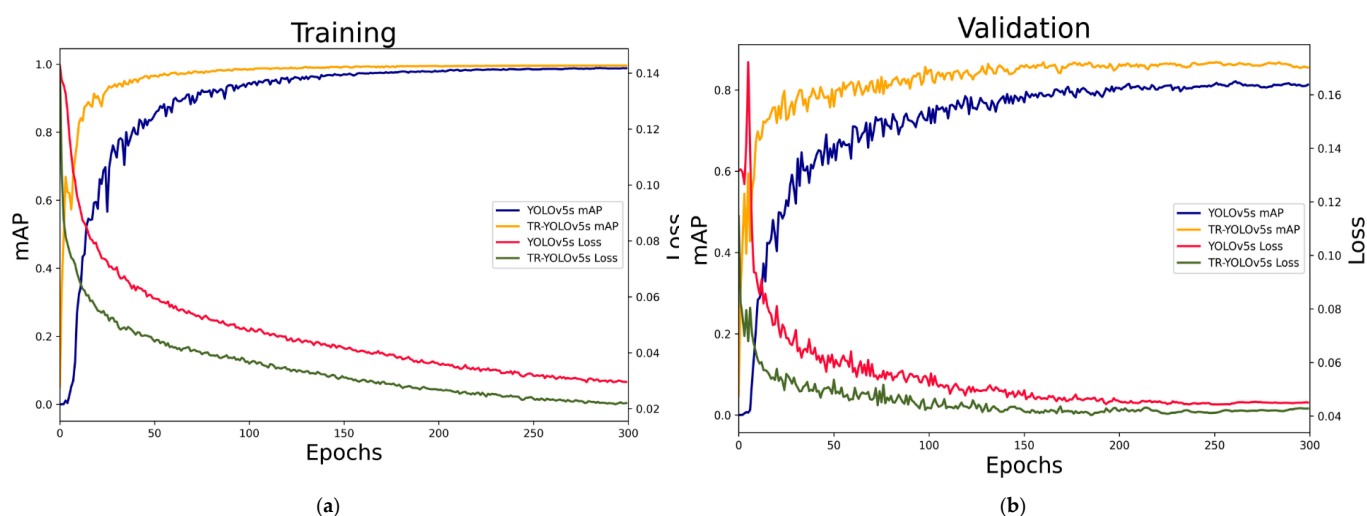


Figure 17. The mAP curves and Loss curves of detectors in the training set and validation set: (a) the mAP curves and training loss curves of two detectors; (b) the mAP curves and validation loss curves of two detectors.

Both the detectors converged at about 250th epochs, and the validation loss curve of pre-trained TR-YOLOv5s is smoother and lower than YOLOv5s. In conclusion, pre-trained TR-YOLOv5s has a faster convergence rate and lower loss value than raw YOLOv5s.

3.2.2. Detector Evaluation

The two detectors were tested on the test set, and the comparison of AP, F_2 score and GFLOPs is shown in Table 3. For the F_2 score, we calculated and compared the maximum macro- F_2 score of detectors at their confidence thresholds.

Table 3. The comparison of AP, macro- F_2 and GFLOPs between YOLOv5s and pre-trained TR-YOLOv5s.

Detector	Target	AP@0.5 ¹	mAP@0.5 ¹	macro- F_2	GFLOPs ²
YOLOv5s	Shipwreck Container	79.5% 66.8%	73.1%	77.2%@0.60 ³	16.3
Pre-trained TR-YOLOv5s	Shipwreck Container	84.6% 86.7%	85.6%	87.8%@0.23 ³	16.2

¹ AP@0.5 and mAP@0.5 denote the AP and mAP when the IOU threshold is set to 0.5. ² GFLOPs denotes 1 billion FLOPs. ³ @0.60 and @0.23 denote the YOLOv5s and pre-trained TR-YOLOv5s get their maximum F_2 scores when the confidence thresholds are set to 0.60 and 0.23 respectively.

It can be found from Table 3 and Figure 18 that:

- Pre-trained TR-YOLOv5s achieved a better performance in detection quality and model complexity than YOLOv5s.
- The AUC of pre-trained TR-YOLOv5s, which is equal to the mAP value, is more than that of YOLOv5s. The same conclusion can also be drawn according to the AP values. The macro- F_2 curves and the curves of each class target achieved by pre-trained TR-YOLOv5s are above the curves achieved by YOLOv5s. Pre-trained TR-YOLOv5s has better precision and recall than YOLOv5s at almost all the IOU thresholds.

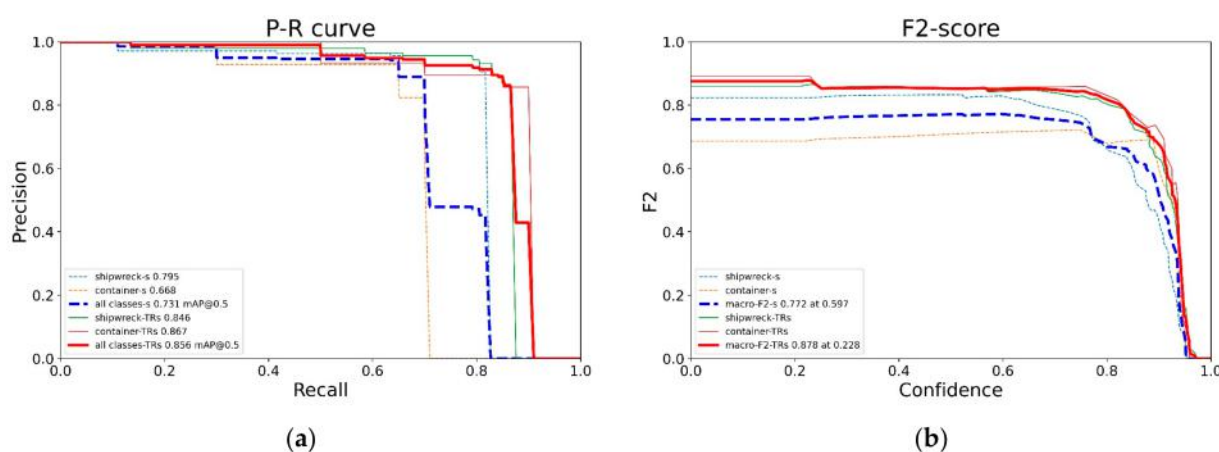


Figure 18. The P-R curves and F_2 score curves of YOLOv5s and pre-trained TR-YOLOv5s: (a) the P-R curves of two detectors in shipwreck and container detection; (b) the F_2 score curves of two detectors in shipwreck and container detection.

3.2.3. Ablation Study

To access the increase of detector performance caused by the pre-trained weighting and the transformer module, an ablation study was done. We compared the precision, recall, mAP, macro- F_2 score and GFLOPs in four situations, including YOLOv5s trained from scratch, YOLOv5s initialized by pre-trained weighting, TR-YOLOv5s trained from scratch and TR-YOLOv5s initialized by pre-trained weighting, on the test set in Table 4.

Table 4. Influence of pre-trained weighting and transformer module on the YOLOv5s recognition performance, including precision, recall, mAP, macro-F2 and GFLOPs.

Pre-Trained	Transformer	Precision	Recall	mAP@0.5 ¹	Macro-F ₂	GFLOPs ²
		87.3%	74.6%	73.1%	77.2%@0.60 ³	16.3
	✓	92.2%	75.3%	81.6%	81.5%@0.24	16.2
✓		91.3%	77.8%	79.1%	81.8%@0.31	16.3
✓	✓	90.9%	84%	85.6%	87.8%@0.23	16.2

¹ mAP@0.5 denotes the mAP value when the IOU threshold is set as 0.5. ² GFLOPs denotes 1 billion FLOPs. ³ @0.60 denotes that the detector get the maximum macro-F₂ score when the confidence threshold is set to 0.60.

According to Table 4, the best mAP 85.6% and macro-F₂ score 87.8% were achieved by pre-trained TR-YOLOv5s due to the transformer module and the prior knowledge from the weighting trained on the open-source dataset.

For the transformer module, no matter the pre-trained weighting is used or not, TR-YOLOv5s outperforms YOLOv5s in mAP and macro-F₂, and the improvement is mainly achieved by increasing the recall. Therefore, the transformer module is helpful to find the target from the background.

For the pre-trained weighting, the performance of pre-trained detectors was better than the trained one from scratch. Hence, the pre-trained parameters or the knowledge learned from the open-source dataset are beneficial to the detector performance.

3.2.4. Qualitative Results and Analysis

For a better visual effect, we visualized the feature maps by converting the first channel of the feature maps from the last three C3 modules which connect to the Detect Heads to pseudo-color images. The first channel of the feature map was normalized to 0–1 by using a sigmoid function and then re-quantized to 0–255. Then, we used the applyColorMap function of Opencv to generate the pseudo-color images and blended them with the original SSS images. The colors in the feature map denote the response degree of the feature extracted by the filter in this layer. The large value in the feature map or the color closed to red means that more features and information are extracted in this area, and the color closed to blue denotes that this area has little feature and useful information. The feature maps S, M and L have different sizes of receptive fields because they are from different depths of layers with different downsampling multiples. For a network, the deeper the layers are, the larger the downsampling multiple is, and the larger the receptive field is. A large receptive field of the feature map means that one pixel in this feature map represents a large area in the original image. Feature maps S, M and L denote the feature maps obtained based on 8, 16 and 32 times downsampling of the raw image, respectively.

Figure 19 shows the feature maps and recognition results by the two detectors. It should be noticed that the feature maps here are generated by the detectors trained from scratch to get rid of the influence of pre-trained weighting because YOLOv5s can transfer all the layers parameters pre-trained from the open-source dataset but TR-YOLOv5s cannot.

For the shipwreck in Figure 19a, TR-YOLOv5s predicted the shipwreck's position and class information correctly with 80% confidence, while YOLOv5s missed the shipwreck target. According to the feature map S, it is obvious that TR-YOLOv5s paid more attention to the shipwreck target and the area near it, while YOLOv5s missed the target influenced by the complex noise of background in the small receptive field. In terms of the feature map M, YOLOv5s extracted more features from the area near the shipwreck target because the shipwreck was a medium-size target which matched the size of feature map M, while TR-YOLOv5s obtained a better feature map because it focused on the shipwreck target with extracting fewer features from the background. However, neither of the two detectors distinguished the target from the background based on the feature map L. We visualized the feature map L0 from the last C3 module without sigmoid operation and compared the L0 of the two detectors. Although neither of the two detectors focused on the target,

YOLOv5s extracted more features from the background due to the barren features of the shipwreck compared with the background with complicated noise, while TR-YOLOv5s noticed some information around the shipwreck because the transformer module calculated the self-attention among the various areas in the SSS image.

For the container in Figure 19b, TR-YOLOv5s predicted the container's position and class information correctly with 68% confidence, while the confidence of the YOLOv5s result was only 36%. According to the feature map S and M, a conclusion similar to the shipwreck target was drawn, and the feature map L still could not indicate the diversity between the two detectors. So we visualized L0 and found that YOLOv5s extracted features from almost the whole image on the large scale, while TR-YOLOv5s concentrated on the container target and the junction of seabed and water column due to their conspicuousness.

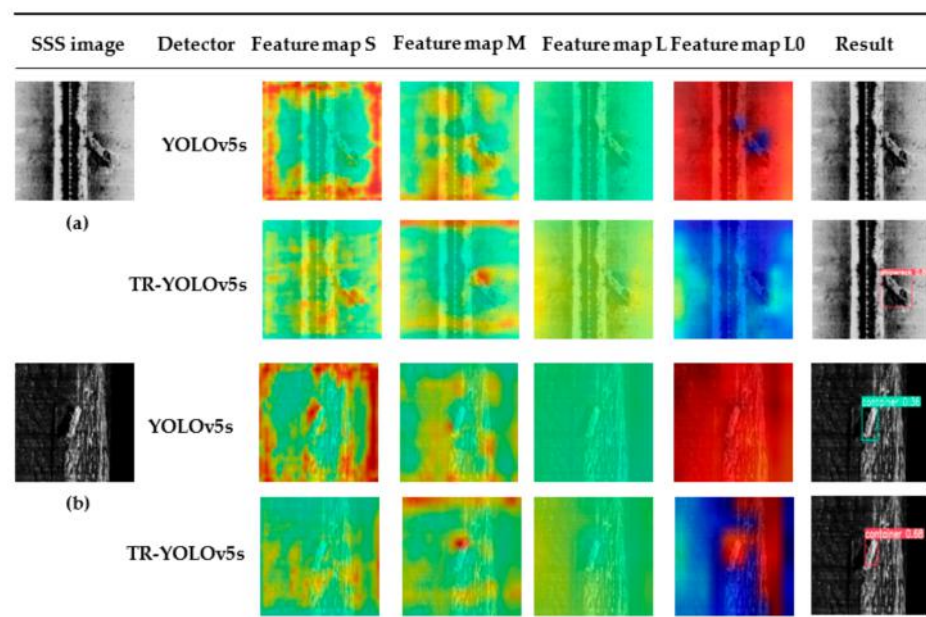


Figure 19. Comparison of the feature maps and recognition results of the shipwreck target in (a) and the container target in (b) between YOLOv5s and TR-YOLOv5s. The feature map S and M are from the antepenultimate and penultimate C3 modules (Figure 6) for detecting small and medium targets. The feature maps L and L0 are from the last C3 modules for detecting large targets.

To sum up, with the help of the transformer module, TR-YOLOv5s can extract more useful and reliable information and features and focus on the targets and the area around them, thus has better performance in the recognition and location of the detected targets.

3.3. Single Shipwreck ATR

The shipwreck data at the Bohai Sea was processed with the procedures detailed in Section 2. In this trial, the surveying ship speed was about 5 knots. The scanning width of Edgetech 4200-MP was 150 m (slant range) on each side. The pulse length and frequency were set to 20 ms and 100 kHz, and the sampling number was 8528 on each side of the ship. The sampling interval between two adjacent pings was about 0.2 s.

3.3.1. SSS Image Preprocessing and Sampling

The first step in the processing was to quantize the sonar data to 0–255 (8 bits) and do sampling in the along-track direction to generate an SSS image when a new ping updates. Figure 20a is an SSS image obtained by sampling once in the along-track direction. The width of the raw SSS image was 17,056 pixels, which was the same as the scanning width of SSS, and the height d was 320 pixels, the standard input size of the detector.

Then, to make the mapping resolution of the cross-track direction the same as that of the along-track direction, the sampling multiple n was calculated according to Equation (2).

$$n = \frac{v \times T}{L/N} = \frac{5 \times 0.514 \times 0.2}{300 \div 17056} \approx 29, \quad (20)$$

where 0.514 denotes the conversion constant from knot to meter per second.

Therefore, the width of the SSS image needed to be resized from 17,056 pixels to 588 pixels, and to facilitate the subsequent processing, the raw SSS image was resized to 600×320 pixels as shown in Figure 20b. According to the detection results of the raw SSS image and the image after downsampling in the cross-track direction (Figure 21), the shipwreck target was recognized correctly (Figure 21d) by the downsampling operation but missed in the raw SSS image (Figure 21a,c). Therefore, it is necessary to do downsampling in the cross-track direction in real-time detection.

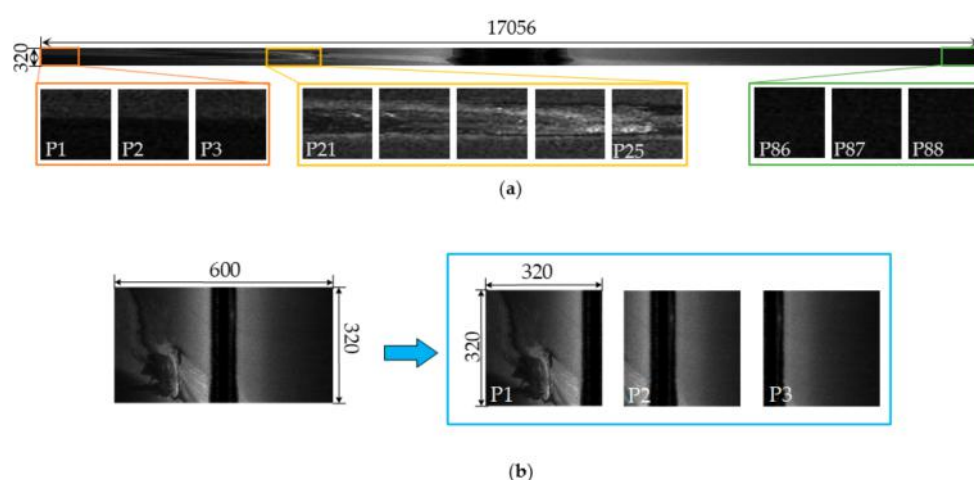


Figure 20. SSS image sampling: (a) patches of raw SSS image; (b) patches of SSS image sampled in the cross-track direction. The patch size is 320×320 , which is the standard input size of the detector.

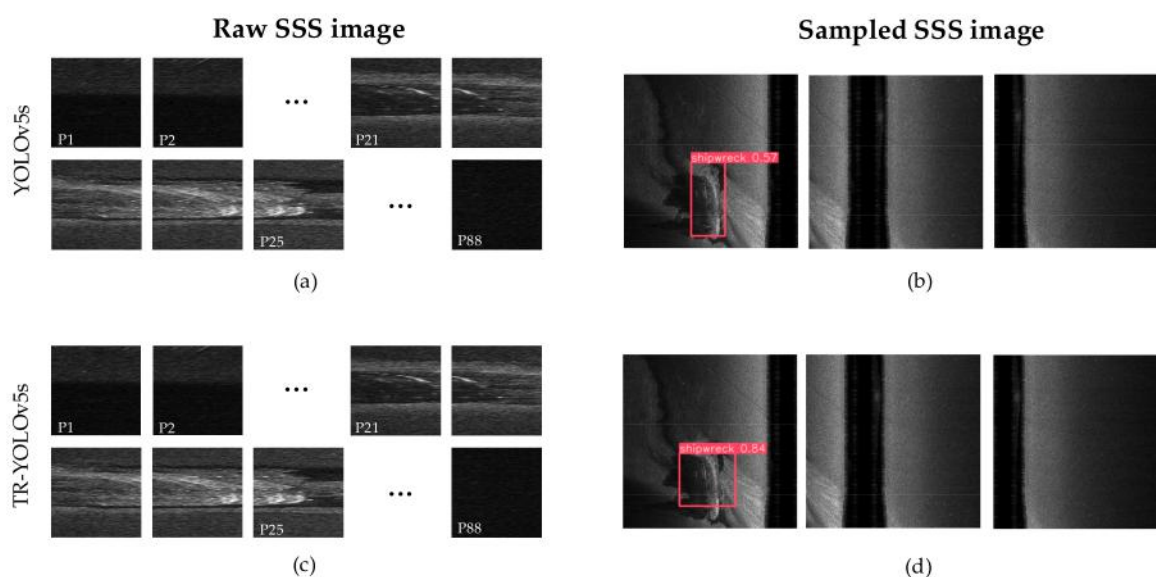


Figure 21. Detection results for SSS image in Figure 20: (a) the detection result of raw SSS image by YOLOv5s; (b) the detection result of sampled SSS image by YOLOv5s; (c) the detection result of raw SSS image by TR-YOLOv5s; (d) the detection result of sampled SSS image by TR-YOLOv5s.

Finally, use the sliding window of size 320×320 pixels to divide the sampled SSS image and generate patches with a common area for better use of the local feature in detection as shown in Figure 20. The common coverage of two adjacent patches is a manually set parameter, and it was set to 40% of patch size in this paper.

3.3.2. Performance of the Detectors

The SSS image was divided into several patches and input into the detector. The detection results from all patches were performed a Non-Maximum Suppression (NMS) in the SSS image coordinate system to delete the repeated bounding boxes.

Qualitative analysis. To make a qualitative analysis of the performance of two detectors, the detection results of the SSS images in Figure 20 were compared with those in Figure 21. According to Figure 21, TR-YOLOv5s and YOLOv5s could not detect the shipwreck in the raw SSS image because the shipwreck was overstretched in the cross-track direction and lost its true aspect ratio due to the difference of the resolution between the along-track direction and the cross-track direction. For the sampled SSS image, though both detectors could detect the shipwreck successfully, TR-YOLOv5s had higher confidence than YOLOv5s due to the transformer module.

Quantitatively analysis. To compare the two detectors more effectively, the results were quantitatively analyzed as shown in Table 5. As the ship sails, the SSS data was constantly updated. In the SSS images generated by sampling along the track direction, the relative position of the shipwreck was constantly changing from invisible to appeared and then disappeared. Based on the dense sampling in the along-track direction, 603 SSS images were generated in total. There were 348 consecutive pings among the whole measurement that the shipwreck was visible, that is, 348 SSS images were containing the shipwreck. Therefore, the precision and recall of both detectors were calculated based on these 348 images, and the confidence threshold of NMS was set to 0.2.

Table 5. The comparison of precision and recall between YOLOv5s and TR-YOLOv5s.

Detector	TP	FP	TP + FN	Precision	Recall	Time
YOLOv5s	250	30	348	89.3%	71.8%	0.033
TR-YOLOv5s	329	0	348	100%	94.5%	0.036

In Table 5, TP denotes the number of images that the detector could recognize the shipwreck target, FP refers to the number of images that the detector misidentifies the shipwreck as a container. TP + FN denotes the total number of SSS images containing the shipwreck.

According to Table 5, TR-YOLOv5s detected the shipwreck successfully in 329 SSS images with no misrecognition, while YOLOv5s only recognized 250 shipwrecks and misidentified 30 shipwrecks as containers. It took 0.033 s for YOLOv5s to finish recognition per image (320×320 pixels), and TR-YOLOv5s needed 0.036 s to complete single image recognition. Therefore, TR-YOLOv5s had a better detection accuracy with an efficiency similar to YOLOv5s.

3.3.3. Target Localization

Based on the procedures above, TR-YOLOv5s recognized the shipwreck target successfully. In the recognition, merely 0.01 s was used for detecting each patch of one sampling image, and the entire time consumption including sampling, target recognition and target localization was about 0.068 s, which met the requirement of real-time recognition. The location of the shipwreck target was calculated as Section 2.3, and the shipwreck is marked in Figure 22.

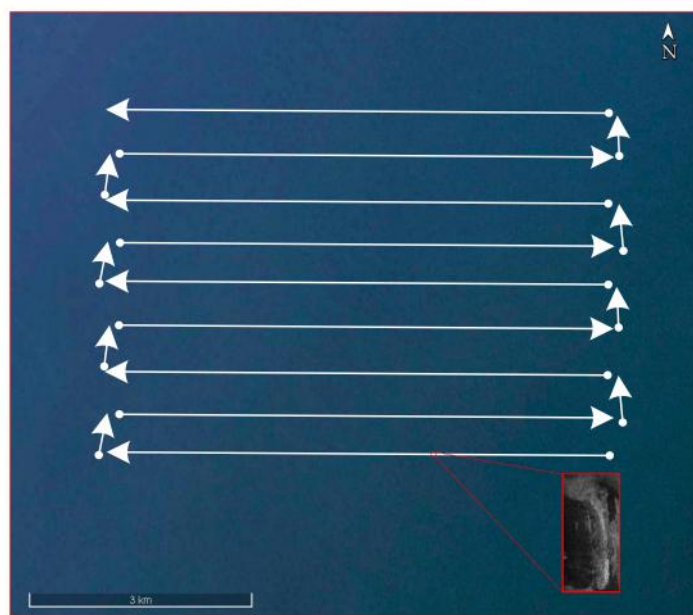


Figure 22. The recognition result of the shipwreck target.

3.4. Multi-Target ATR

3.4.1. Multi-Shipwreck ATR

The shipwreck data in the Wusong River was processed with procedures detailed in Section 2. In this trial, the shipping speed was about 3 knots. The scanning width of Shark-S150D was set to 45 m on each side (slant range). The pulse length was set to 20–1000 μ s and the frequency was set to 150 kHz, and the sampling number was 6000 on each side of the ship. The sampling interval between two adjacent pings was about 0.2 s.

The three shipwrecks are all detected successfully by the proposed method as shown in Figure 23. In this experiment, TR-YOLOv5s achieved 80.2% precision and 86.6% recall, and it took about 0.01 s to finish recognition per patch on average. Both precision and recall were reduced compared with Section 3.3 because of the changing of SSS image quality and other underwater targets resembling shipwrecks. Shipwreck 2 was recognized twice in different directions because of the common coverage area of sailing routes.

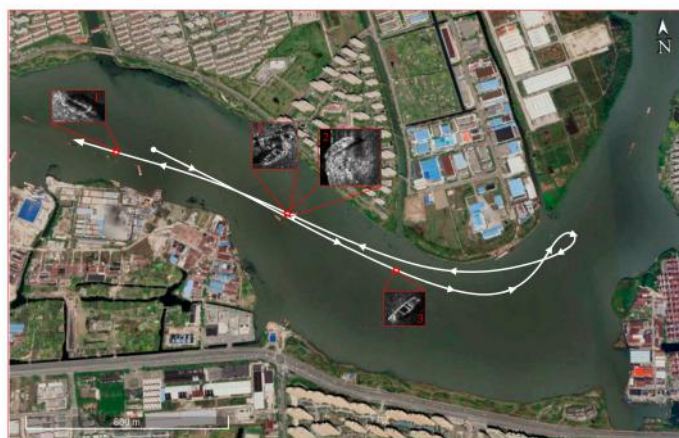


Figure 23. The recognition result of shipwreck target.

3.4.2. Multi-Container ATR

The container data at the Yangtze River was processed with procedures detailed in Section 2. In the container ATR trial, the shipping speed during the sweeping survey was

about 4 knots. The pulse length and frequency were set to 25 μ s and 123 kHz, and the scanning width of Benthos SIS-1624 was 200 m on each side (slant range).

Six submarine containers were detected successfully based on the proposed method and the locations of the containers were calculated and marked in the sailing route image as shown in Figure 24.

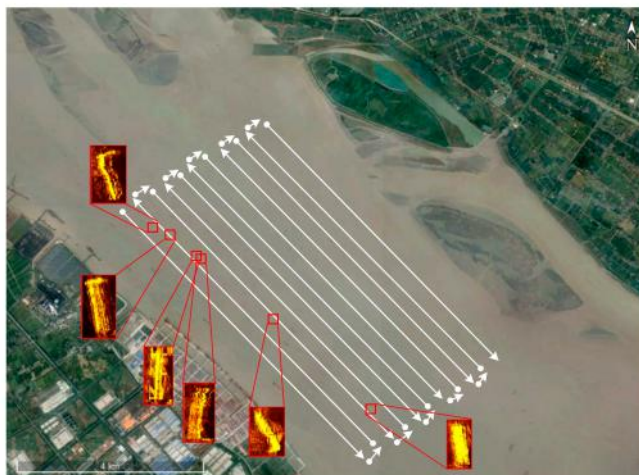


Figure 24. The recognition result of container target.

4. Discussion

4.1. Significance of the Proposed Method

The proposed real-time SSS ATR method provides an effective way to replace manual interpretation in the underwater target recognition, which is essential to achieve the real-time ATR in channel dredging, shipwreck and container salvage, maritime search and rescue and maritime archaeology. Besides, the real-time image detected can come from the data processing on the raw SSS observation information or the screenshot on the waterfall images produced by some business software of SSS data collection, which greatly facilitates the real-time target detection.

The introduced transformer module improved the recognition precision with fewer GFLOPs than YOLOv5s. It is because the given TR-YOLOv5s combines the transformer module and YOLOv5s to deal with the target-sparse and feature-barren characteristics of SSS images. In our experiments, the transformer module with four heads of self-attention paid more attention to the targets and the area near the targets, not the background. Because it can change the weightings of each pixel based on the self-attention from four different subspaces, which is to adapt the distribution of input images to the training data actually. Therefore, the transformer module or the attention mechanism is useful to distinguish the targets from the background even if the SSS images are target-sparse and feature-barren so as to improve the accuracy and efficiency of the underwater target recognition.

The downsampling used in the proposed method improved the recognition precision and reduced computing costs by downsampling in the cross-track direction. It can maintain the rough aspect ratio and shape information of underwater targets which are lost to some degree because the difference of mapping resolution between two directions will lead to a distortion of a target.

4.2. Sensitivity Analysis

4.2.1. The Heads Number in the Transformer Module

The transformer module is made up of several self-attentions, named multi-head self-attention structure (Section 2.2.2). Therefore, the number of heads in the transformer module can be changed as a hyper-parameter in the training process. Each head of

self-attention can be viewed as a feature subspace mapped from the feature maps from Backbone to adapt the distribution of the images to the training set.

To access the influence of the heads number on the detector performance, a sensitivity analysis was conducted by changing the heads number to 2, 4, 8, 16, 32 and 64. All the numbers of heads were the multiple of 2 because the heads number should be divisible by the channel of the feature maps from the previous layer, which was 1024 in TR-YOLOv5s. The comparison of mAP and macro-F₂ values are shown in Table 6.

Table 6. Performance of TR-YOLOv5s with different numbers of heads in transformer module.

Index \ Head	2	4	8	16	32	64
mAP@0.5	83.3%	85.6%	82.2%	82.2%	79.9%	81.8%
Macro-F ₂	84.8%	87.8%	84.4%	85.3%	82.6%	83.9%

According to Table 6, it achieved the best mAP 85.6% and macro-F₂ 87.8% when the heads number was set to 4, which is the default value in our method. Furthermore, the performance of 2 heads, 8 heads and 16 heads were almost the same with a little decrease compared with 4 heads. Nevertheless, when the number of heads deviated from 4, the performance of TR-YOLOv5s degrades gradually. Therefore, TR-YOLOv5s is not very sensitive to the change of heads number in the transformer module.

4.2.2. The Complexity of the SSS Images

The complexity of the SSS images, especially for the texture features, is different due to the difference of types of SSS, water environment and topography of the seafloor, etc. The SSS images consists of a target and smooth seabed are easy to be detected correctly, but it may be not the same when the target is surrounded by the rock or sand ripples. Therefore, to study the sensitivity of our method for different image complexity, we divided the test set into five subsets based on the proposed complexity descriptor and compared the detection performance among these subsets. Some samples of the five subsets are shown in Figure 25.

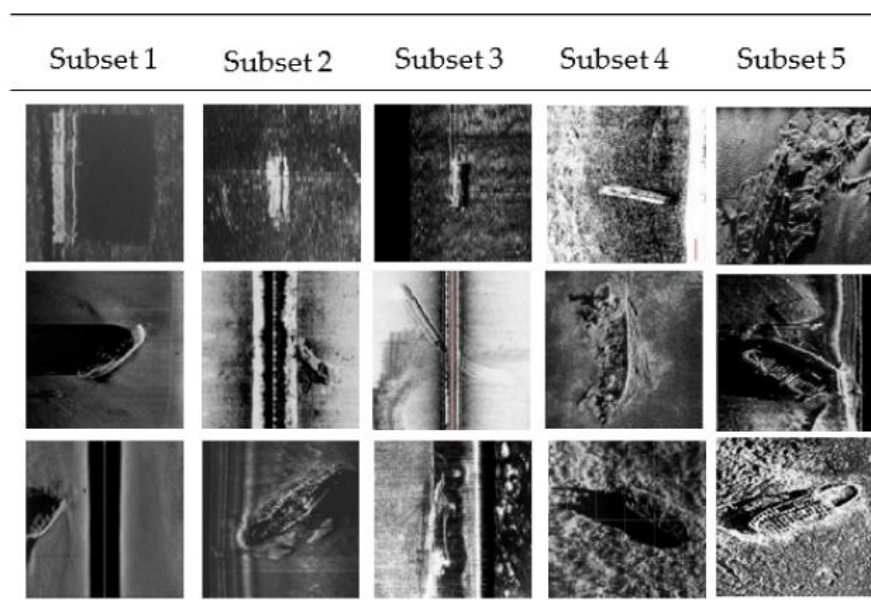


Figure 25. Some samples of 5 subsets divided from the test set.

Firstly, a descriptor of image complexity was built by calculating the sum of the horizontal and vertical gradients with the help of a Sobel operator. Then, all the 94 images

in the test set were represented using the proposed complexity descriptor and sorted as the complexity value.

After that, every 20 images were divided into a subset and the fifth subset had 14 images. The complexity of the 5 subsets are in the range 0 to 170, 170 to 270, 270 to 360, 360 to 530 and 530 to 1600, respectively.

Finally, the subsets were tested by TR-YOLOv5s and the results are shown in Table 7.

Table 7. Performance of TR-YOLOv5s in different complexity images.

Index \ Subset	1	2	3	4	5
Complexity	0–170	170–270	270–360	360–530	530–1600
Numbers	20	20	20	20	14
mAP@0.5	90.1%	91.7%	77.4%	89.7%	81.4%

According to the test results, the proposed method achieved a good detection effect. It proves that the transformer module is helpful in the vast majority of cases due to the attention to the targets and the area around the targets. Relatively, the detection accuracy is low in subsets 3 and 5 because of complex image background and noise, which indicates that the image quality has a certain influence on the proposed method. The problem can be solved by reducing the ship speed in SSS measurement and improving the image quality by image preprocessing. In addition, collecting more samples and improving the performance of the detector is also an important way.

4.2.3. The Measurement Conditions

In real-time detection, the measurement conditions such as the performance of SSS, ship speed, water depth, water environment and so on will influence the imaging quality, thereby the distributions of the SSS images put into the detector is changed and the detection results may be disturbed.

Theoretically, the way in which measurement conditions influence the imaging quality and style mainly includes target status, noise, resolution and texture changes. It is hard to collect the target images under various conditions. A possible way to get different style images is to simulate SSS measurement by considering SSS imaging mechanism, water environment, seabed background, and detected target status on seabed, etc.

4.3. Performance of Detector in Unbenign Seabed

To verify the performance of our method in the areas of unbenign sections of the seabed, we selected three SSS images whose targets are in the high clutter and sand ripples areas. Because the container data we have are always in areas of benign sections of the seabed, the three images selected are all the shipwreck targets. The detection results are shown in Figure 26.

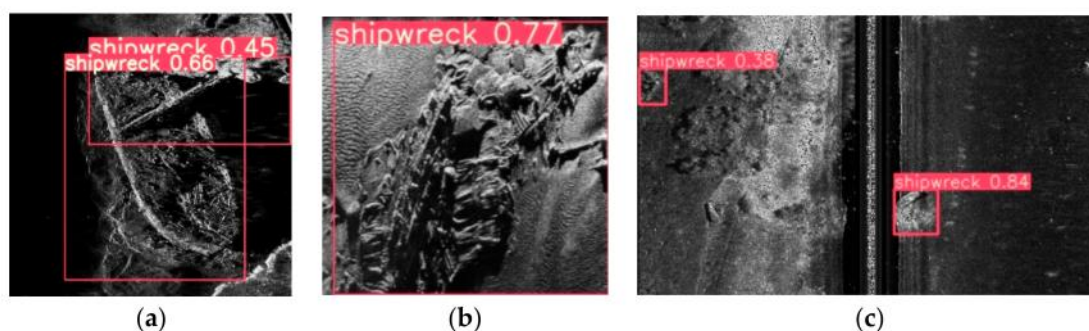


Figure 26. Detection results of targets in unbenign sections of the seabed: (a) shipwreck in a complicated environment; (b) eroded shipwreck in an area of sand ripples; (c) shipwreck in a high clutter area.

For the image in Figure 26a, the detector was affected by the complicated environment around the shipwreck target. Although the detector recognized the shipwreck successfully, it predicted a redundant bounding box of the shipwreck with lower confidence around the correct bounding box.

For the image in Figure 26b, the detector predicted the eroded shipwreck target in the sand ripples area correctly with 77% confidence.

For the image in Figure 26c, the shipwreck targets are in a high clutter area. On the one hand, the detector predicted a wrong result in the upper left corner and thought it is a shipwreck with 0.38 confidence but an unknown target that is not a shipwreck; on the other hand, in spite that the detector found the shipwreck target, the predicted position was not precise because the predicted bounding box contained the high clutter area around the shipwreck target.

Therefore, our method can detect the targets in the unbenign seabed in spite that the predicted confidence and position are not very precise with some low-confidence wrong results.

4.4. Comparison with Existing Methods

Compared with object detection after SSS measurement, the proposed real-time detection is very efficient for quick object detection and positioning on the spot. To further evaluate the performance of this proposed method, the proposed method was also compared with the mail existing real-time underwater object detection methods and listed in Table 8.

Table 8. Comparison of our object detection method performance with existing real-time underwater object detection methods.

Num	Method	Algorithm	Task	Object	Accuracy	Efficiency
1	Our method	TR-YOLOv5s	Detection	Shipwreck Container	85.6% (mAP, Laboratory)	0.068 milliseconds per 100 pixels (Ship)
2	Song et al. (2019) [22]	Self-Cascaded CNN	Segmentation	Highlight Shadow Seafloor	57.6~97.1% (mIOU ¹ , Laboratory)	0.067 s per ping (AUV)
3	Wu et al. (2019) [23]	Depth-Wise Separable Convolution	Segmentation	Object Background	66.2% (mIOU, Laboratory)	0.038 milliseconds per 100 pixels (Laboratory)
4	Burguera et al. (2020) [24]	Fully convolutional neural network	Segmentation	Rock Sand Other	87.8% (F ₁ score, AUV)	4.6 milliseconds per 100 pixels (AUV)

¹ mIOU, mean Intersection over Union, the index for segmentation evaluation.

According to Table 8, the existing methods mainly realize real-time detection through image segmentation. The segmentation can obtain the detailed information of each pixel, and distinguishes the object areas (method 2 and 3) from background or realizes seafloor description (method 4), but cannot classify the target. The proposed method in this paper can recognize and locate the detected target. Besides, the proposed method is faster than method 4, and a little bit slower than method 3 that just calculated the time for segmentation. Comparatively, the proposed method has better performance.

4.5. Limitations of the Proposed Method

The proposed TR-YOLOv5s achieved a better performance than YOLOv5s in the above experiments, but still needs improvement in mAP and macro-F₂. In the construction of the recognition model, we have only 286 original samples for training. The lack of samples makes it easier for the detectors to overfit because the few samples are not enough for the detectors to learn excellent representation or features, which limits the

recognition accuracy. Theoretically, for optical images, it needs no less than 1500 images or 10,000 targets per class in the training dataset to get the best performance of YOLOv5. For SSS images, the number of training samples required is less than that of optical images due to the target-sparse and feature-barren characteristics of SSS images. Moreover, even though 2300 samples obtained by the data augmentation are used for the recognition network training, the representative of these samples is not complete due to adopting the image transformation techniques but ignoring the SSS imaging mechanism, which also affects the target recognition accuracy. The problem can be solved by collecting more original target samples and adopting better data augmentation algorithms by considering the SSS imaging mechanism.

The efficiency of the proposed method may be affected by dense sampling in the along-track direction. To decrease the burden of the real-time target recognition and improve the detection efficiency, the sampling interval is recommended as one-third of the size of the detected target.

5. Conclusions and Recommendations

The proposed method, the integration of the raw image preprocessing, the sampling, a novel detector of TR-YOLOv5s and the localization, fulfills automatically underwater maritime object detection from side-scan sonar waterfall images with high accuracy and efficiency and can meet requirement of real-time underwater target detection.

It is very necessary to give some corresponding measures, such as the transformer module and the downsampling according to the characteristics of SSS images and target distributions in the detected images, which is very beneficial to improve the accuracy and efficiency of real-time underwater target detection.

In order to achieve a better performance, sonar data augmentation, few sample learning even zero-shot-learning methods should be considered, and more flexible network structures such as model compression technology could be studied in real-time detection. Additionally, the special module or underlying calculation mode based on the characteristic of SSS images is supposed to be considered.

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

Funding: This research was funded by National Natural Science Foundation of China under Grant 42176186, in part by National Key R&D Program of China under Grant 2016YFB0501703. This research was also funded by the Key Research & Development Program of New Energy Engineering Limited Company of China Communications Construction Company Third Harbor Engineering Limited Company under Grant 2019-ZJKJ-ZDZX-01-0349 and Class-A project of New Energy Engineering Limited Company of China Communications Construction Company Third Harbor Engineering Limited Company under Grant 2020-04.

Data Availability Statement: Access to the data will be considered upon request by the authors.

Acknowledgments: We would like to thank the editor and the anonymous reviewers for their valuable comments and suggestions that greatly improve the quality of this paper, and thank Shanghai Maritime Surveying and Mapping Center of Donghai Navigation Safety Administration for providing valuable SSS data. Thank our colleagues for their helpful suggestions during the experiment, and. Finally, we would like to thank the open-source project of Ultralytics-Yolov5.

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

References

1. Celik, T.; Tjahjadi, T. A Novel Method for Sidescan Sonar Image Segmentation. *IEEE J. Ocean. Eng.* **2011**, *36*, 186–194. [[CrossRef](#)]
2. Zheng, G.; Zhang, H.; Li, Y.; Zhao, J. A Universal Automatic Bottom Tracking Method of Side Scan Sonar Data Based on Semantic Segmentation. *Remote Sens.* **2021**, *13*, 1945. [[CrossRef](#)]

3. Li, S.; Zhao, J.; Zhang, H.; Bi, Z.; Qu, S. A Novel Horizon Picking Method on Sub-Bottom Profiler Sonar Images. *Remote Sens.* **2020**, *12*, 3322. [CrossRef]
4. Barngrover, C.; Althoff, A.; DeGuzman, P.; Kastner, R. A brain computer interface (BCI) for the detection of mine-like objects in sidescan sonar imagery. *IEEE J. Ocean. Eng.* **2016**, *41*, 123–138. [CrossRef]
5. Lehardy, P.K.; Moore, C. Deep ocean search for Malaysia airlines flight 370. In *2014 Oceans—St. John's, St. John's*; IEEE: Piscataway, NJ, USA, 2014; pp. 1–4.
6. Neupane, D.; Seok, J. A Review on Deep Learning-Based Approaches for Automatic Sonar Target Recognition. *Electronics* **2020**, *9*, 1972. [CrossRef]
7. Zheng, L.; Tian, K. Detection of Small Objects in Sidescan Sonar Images Based on POHMT and Tsallis Entropy. *Signal. Process.* **2017**, *142*, 168–177. [CrossRef]
8. Fakiris, E.; Papatheodorou, G.; Geraga, M.; Ferentinos, G. An Automatic Target Detection Algorithm for Swath Sonar Backscatter Imagery, Using Image Texture and Independent Component Analysis. *Remote Sens.* **2016**, *8*, 373. [CrossRef]
9. Xiao, W.; Zhao, J.; Zhu, B.; Jiang, T.; Qin, T. A Side Scan Sonar Image Target Detection Algorithm Based on a Neutrosophic Set and Diffusion Maps. *Remote Sens.* **2018**, *10*, 295. [CrossRef]
10. Guillaume, L.; Sylvain, G. Unsupervised extraction of underwater regions of interest in sidescan sonar imagery. *J. Ocean. Eng.* **2020**, *15*, 95–108.
11. Zhu, B.; Wang, X.; Chu, Z.; Yang, Y.; Shi, J. Active Learning for Recognition of Shipwreck Target in Side-Scan Sonar Image. *Remote Sens.* **2019**, *11*, 243. [CrossRef]
12. Nargesian, F.; Samulowitz, H.; Khurana, U.; Khalil, E.B.; Turaga, D. Learning Feature Engineering for Classification. In Proceedings of the 26th International Joint Conference on Artificial Intelligence, Melbourne, Australia, 19–25 August 2017; pp. 2529–2535.
13. Nguyen, H.; Lee, E.; Lee, S. Study on the Classification Performance of Underwater Sonar Image Classification Based on Convolutional Neural Networks for Detecting a Submerged Human Body. *Sensors* **2019**, *20*, 94. [CrossRef]
14. Bore, N.; Folkesson, J. Modeling and Simulation of Sidescan Using Conditional Generative Adversarial Network. *IEEE J. Ocean. Eng.* **2020**, *46*, 195–205. [CrossRef]
15. Steiniger, Y.; Kraus, D.; Meisen, T. Generating Synthetic Sidescan Sonar Snippets Using Transfer-Learning in Generative Adversarial Networks. *J. Mar. Sci. Eng.* **2021**, *9*, 239. [CrossRef]
16. Lee, S.; Park, B.; Kim, A. Deep Learning based Object Detection via Style-transferred Underwater Sonar Images. In Proceedings of the 12th IFAC Conference on Control Applications in Marine Systems, Robotics, and Vehicles CAMS 2019, Daejeon, Korea, 18–20 September 2019; pp. 152–155.
17. Kim, J.; Choi, J.; Kwon, H.; Oh, R.; Son, S. The application of convolutional neural networks for automatic detection of underwater object in side scan sonar images. *J. Acoust. Soc. Korea* **2018**, *37*, 118–128.
18. Einsidler, D.; Dhanak, M.; Beaujean, P. A Deep Learning Approach to Target Recognition in Side-Scan Sonar Imagery. In Proceedings of the MTS/IEEE Charleston OCEANS Conference, Charleston, SC, USA, 22–25 October 2018; pp. 1–4.
19. Nayak, N.; Nara, M.; Gambin, T.; Wood, Z.; Clark, C.M. Machine learning techniques for AUV side scan sonar data feature extraction as applied to intelligent search for underwater archaeology sites. *Field Serv. Robot.* **2021**, *16*, 219–233.
20. Huo, G.; Wu, Z.; Li, J. Underwater Object Classification in Sidescan Sonar Images Using Deep Transfer Learning and Semisynthetic Training Data. *IEEE Access* **2020**, *8*, 47407–47418. [CrossRef]
21. Li, C.L.; Ye, X.F.; Cao, D.X.; Hou, J.; Yang, H.B. Zero shot objects classification method of side scan sonar image based on synthesis of pseudo samples. *Appl. Acoust.* **2021**, *173*, 107691. [CrossRef]
22. Song, Y.; He, B.; Liu, P. Real-Time Object Detection for AUVs Using Self-Cascaded Convolutional Neural Networks. *IEEE J. Ocean. Eng.* **2019**, *46*, 56–67. [CrossRef]
23. Wu, M.; Wang, Q.; Rigall, E.; Li, K.; Zhu, W.; He, B.; Yan, T. ECNet: Efficient Convolutional Networks for Side Scan Sonar Image Segmentation. *Sensors* **2019**, *19*, 2009. [CrossRef]
24. Burguera, A.; Bonin-Font, F. On-Line Multi-Class Segmentation of Side-Scan Sonar Imagery Using an Autonomous Underwater Vehicle. *J. Mar. Sci. Eng.* **2020**, *8*, 557. [CrossRef]
25. Ultralytics-Yolov5. Available online: <https://github.com/ultralytics/yolov5> (accessed on 1 January 2021).
26. Xu, R.; Lin, H.; Lu, K.; Cao, L.; Liu, Y. A Forest Fire Detection System Based on Ensemble Learning. *Forests* **2021**, *12*, 217. [CrossRef]
27. Lofte, S.; Szegedy, C. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In Proceedings of the 32nd International Conference on Machine Learning (ICML), Lille, France, 6–11 July 2015; pp. 448–456.
28. Elfving, S.; Uchibe, E.; Doya, K. Sigmoid-weighted linear units for neural network function approximation in reinforcement learning. *Neural Netw.* **2018**, *107*, 3–11. [CrossRef] [PubMed]
29. Wang, C.; Liao, H.M.; Wu, Y.; Chen, P.; Hsieh, J.; Yeh, I. CSPNet: A New Backbone that can Enhance Learning Capability of CNN. In Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Seattle, WA, USA, 14–19 June 2020; pp. 1571–1580.
30. Lin, T.; 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.

-
31. Liu, S.; Qi, L.; Qin, H.; Shi, J.; Jia, J. Path Aggregation Network for Instance Segmentation. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 8759–8768.
 32. Redmon, J.; Farhadi, A. YOLOv3: An incremental improvement. *arXiv* **2018**, arXiv:1804.02767.
 33. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, L.; Polosukhin, I. Attention is all you need. In Proceedings of the 31st International Conference on Neural Information Processing Systems (NIPS), Long Beach, CA, USA, 4–9 December 2017; pp. 6000–6010.