





# An Efficient Deep-Sea Debris Detection Method Using Deep Neural Networks

Bing Xue , Baoxiang Huang , *Member, IEEE*, Weibo Wei , Ge Chen , Haitao Li, Nan Zhao, and Hongfeng Zhang

**Abstract**—Marine debris impacts negatively upon the marine environment and the survival of marine life because they are some difficult-to-degrade substances, and most of them will sink into the deep sea and continue to exist in the ocean. Autonomous underwater vehicles can clean up the deep-sea debris to some extent. However, the efficient detection method plays a critical role in the collection rate. This article establishes an efficient deep-sea debris detection method with high speed using deep learning methods. First, a real deep-sea debris detection dataset (3-D dataset) is established for further research. The dataset contains seven types of debris: cloth, fishing net and rope, glass, metal, natural debris, rubber, and plastic. Second, the one-stage deep-sea debris detection network ResNet50-YOLOV3 is proposed. In addition, eight advanced detection models are also involved in the detection process of deep-sea debris. Finally, the performance of ResNet50-YOLOV3 is verified by experiments. Furthermore, the applicability and effectiveness of ResNet50-YOLOV3 in deep-sea debris detection are proved by the experimental results.

**Index Terms**—Deep convolutional neural network, deep-sea debris detection, deep-sea debris detection dataset, sea floor.

## I. INTRODUCTION

THE marine environment has received more and more attention around the world, and marine debris is one of the main culprits for the harsh marine environment. Along with the expansion of human activities on the coast and ocean and the increase of garbage [1], most of the debris has been flowed to the ocean and finally sinks to deep ocean. Compared with the garbage on the ocean surface, the marine environment and the survival of organisms will be more severely threatened by the

deep-sea debris [2], [3]. Fortunately, autonomous underwater vehicles (AUVs) can complete garbage collection and cleaning on the seabed by manipulating a robotic arm, but it needs a strong deep-sea debris detection performance as a support [4]. Therefore, accurate automatic deep-sea debris detection capabilities are necessary for AUVs.

For now, some studies have been carried out around the classification and detection of marine debris. Traditional machine learning algorithms were used to classify marine plastic garbage on the beach [5]. A reversed linear spectral unmixing methodology has been applied to the detection of garbage floating in the ocean [6]. Satellite remote sensing technology [7], [8], as well as unmanned aerial vehicle systems [9], [10], is usually used to obtain image data of garbage on the sea surface and beach [11], and then, spectral feature analysis [12], plastic index [13], along with LIDAR [14] is adopted to realize the identification and detection of beach and sea surface debris. With the excellent performance of Faster R-CNN [15], SSD [16], and other detection networks on classical detection datasets [17], [18], it is also a trend that seabed garbage is detected using these networks [19]–[21], and certain effects have also been achieved.

However, most marine garbage detection only focuses on the sea surface and beach; there are few comprehensive studies on the detection of deep-sea debris. Although some detection networks have begun to be used to detect underwater garbage, these have not achieved satisfactory results. Fulton *et al.* [21] only detected the plastic waste, and the category is single; Valdenegro-Toro [19] did not use a complete deep learning detection network.

One of the reasons for the above situation is the lack of real deep-sea debris detection datasets, which has led to the scarcity of research on deep-sea debris detection. Deep-sea garbage data need to be captured in a real deep-sea environment by using professional diving equipment and high-precision cameras, which requires huge manpower and material resources. Although there are current studies that create a marine garbage dataset by simulating the deep-sea environment in a water tank and use it to train garbage detection algorithms [19], it is not clear whether it is applicable to the natural marine environment. Another important reason is the serious interclass similarity and intraclass variability of deep-sea debris [4], which brings great difficulties to research related to deep-sea garbage detection. Different from sea garbage and beach garbage, deep-sea debris is always on the bottom of the sea and is eroded by sea water, and its appearance is severely deformed. Coupled with the

Manuscript received August 6, 2021; revised October 20, 2021; accepted November 11, 2021. Date of publication November 24, 2021; date of current version December 13, 2021. This work was supported in part by the National Natural Science Foundation of China under Grant 42030406, in part by the Marine Science and Technology Fund of Shandong Province for Pilot National Laboratory for Marine Science and Technology (Qingdao) under Grant 2018SDKJ0102-8, in part by the Ministry of Science and Technology of China under Grant 2019YFD0901001, and in part by the Natural Science Foundation of Shandong Province ZR2021MD001. (*Corresponding authors: Baoxiang Huang; Weibo Wei.*)

Bing Xue, Baoxiang Huang, Weibo Wei, Nan Zhao, and Hongfeng Zhang are with the College of Computer Science and Technology, Qingdao University, Qingdao 266071, China (e-mail: xvacebv@163.com; hbx3726@163.com; weiweibo@qdu.edu.cn; ZN\_17853269733@163.com; 332303635@qq.com).

Ge Chen is with the School of Marine Technology, Institute for Advanced Ocean Study, Ocean University of China, Qingdao 266100, China (e-mail: gechen@ouc.edu.cn).

Haitao Li is with the College of Information Science and Technology, Qingdao University of Science and Technology, Qingdao 266061, China (e-mail: taohaili@sina.com).

Digital Object Identifier 10.1109/JSTARS.2021.3130238

influence of ocean light and attachments, the characteristics of the same type of seabed garbage in the capture imaging are no longer uniform, and different types of seabed garbage may form feature similarities. This is the main reason for the scarcity of research on submarine garbage and the difficulty and challenge of deep-sea garbage detection than terrestrial garbage detection [22]. Therefore, those networks that have achieved the ideal detection of ocean or beach surface garbage are also difficult to achieve accurate detection of deep-sea debris due to this characteristic of deep-sea debris [21]. Consequently, the algorithms or models that can accurately detect deep-sea debris should be further studied.

Furthermore, Considering the harshness and complexity of the deep-sea environment, it is necessary to improve the efficiency of AUVs in cleaning and collecting garbage in the deep sea. Therefore, the detection speed of the detection model assembled in AUVs for deep-sea garbage should also be considered.

In view of the above, the main work and contributions of this article are as follows.

- 1) Considering the lack of datasets that can be used for deep-sea debris detection, a 3-D dataset (deep-sea debris detection dataset) is constructed based on the online deep-sea debris database [23]. The debris categories of this dataset are divided into seven categories: cloth, fishing net and rope, glass, metal, natural debris, rubber, and plastic. The debris in the database is captured by submarine cameras in the real deep-sea environment, so the 3-D dataset has authenticity and applicability.
- 2) In order to promote the research of deep-sea debris detection methods and improve the accuracy and speed of the model for deep-sea garbage detection, the detection network ResNet50-YOLOV3 is proposed, which is a one-stage detection network with ResNet50 as the backbone (feature extractor) and YOLOV3 [24] as the feature detector.
- 3) SSD [16] and Faster R-CNN [15] detectors are used as comparative detectors, and three classic classification networks ResNet50, VGG16, and MobileNetV2 are used as the backbone. Further experimental results show that compared with the other eight detection models, the proposed ResNet50-YOLOV3 can maintain a considerable detection speed while maintaining the highest detection results. Finally, the experimental analysis also reveals the influence of different backbones on the detection results of submarine garbage.

This study is among the first that detect multiclass deep-sea debris using deep learning. The rest of this article is organized in the following format. The technical details and characteristics of the proposed ResNet50-YOLOV3 method are described in Section II. Section III reports experiments and discussions. Section IV concludes this article.

## II. METHODOLOGY

Detection networks can be summarized into two types at present. One is a two-stage network represented by R-CNN [25] and Faster R-CNN. The other is a one-stage network represented by RetinaNet [26] and SSD. The two-stage networks first

propose the proposed area and then classify the proposal and return the exact coordinates. It has high detection accuracy; the detection speed is not satisfactory. On the contrary, the one-stage networks have a faster detection speed in that it completes the recognition/regression in one time, but this is at the cost of loss of accuracy.

To produce excellent deep-sea debris detection effects in terms of speed and accuracy, the ResNet50-YOLOV3 detection network is introduced in this article to detect deep-sea debris. In this detection network, a multiscale detector called YOLOV3 (you only look once-v3) is applied, which can achieve a high detection speed while ensuring the accuracy of detection. In addition, as a residual network with strong feature abstraction ability, ResNet50 [27] is chosen as the backbone of the network, which can further improve the detection accuracy. ResNet50-YOLOV3 is an end-to-end network, i.e., a one-stage network, which can achieve the speed versus accuracy tradeoff. The detailed structure of the network will be introduced in this section.

### A. Network Structure

1) *Overall Architecture:* The structure of ResNet50-YOLOV3 is depicted in Fig. 1. It can be divided into two parts: feature extractor (backbone) and feature detector (multiscale detector). The image to be detected is mapped out a series of low-level and high-level features through the feature extractor. Then, these low-level and high-level semantic features are further encoded by the feature detector to achieve the final target detection.

The input size of the network follows default size  $416 \times 416$  of YOLOV3 [24]. The image is first subjected to  $7 \times 7$  convolution and  $3 \times 3$  max pooling to complete the preliminary processing, and then, the features flow through four kinds of ResBlock blocks. Each block contains three, four, six, and three ResBlocks, respectively. It is noted that  $1 \times 1$  convolution is required on the branch of the first ResBlock of each block to unify the scale to facilitate subsequent add operations. The batch normalization and the ReLU function are carried out after convolution.

The finally generated features with a scale of  $13 \times 13 \times 2048$  have the advanced features and the most abstract image information and are subsequently processed by YoloBlock to generate  $13 \times 13 \times 512$  features. On the one hand, the features are used to obtain the deepest prediction result with a scale of  $13 \times 13 \times 36$  through  $3 \times 3$  and  $1 \times 1$  convolution. On the other hand, it is subjected to a double upsampling operation and then to a concat operation with the penultimate layer features of the backbone. Similarly, the  $26 \times 26 \times 1280$  feature map obtained by the concat operation flows into YoloBlock, and then, a branch undergoes two convolutions to obtain a shallower prediction result with a scale of  $26 \times 26 \times 36$ . The other branch is upsampled to perform concat operation with the  $52 \times 52 \times 512$  shallow features of the third-to-last layer of the backbone, which is used to generate a shallow prediction result with a scale of  $52 \times 52 \times 36$ .

The number of channels of the detection results is 36, which can be regarded as  $3 \times (4 + 1 + 7)$ . It represents the results of seabed garbage detection and will be discussed in Section II-A3b.

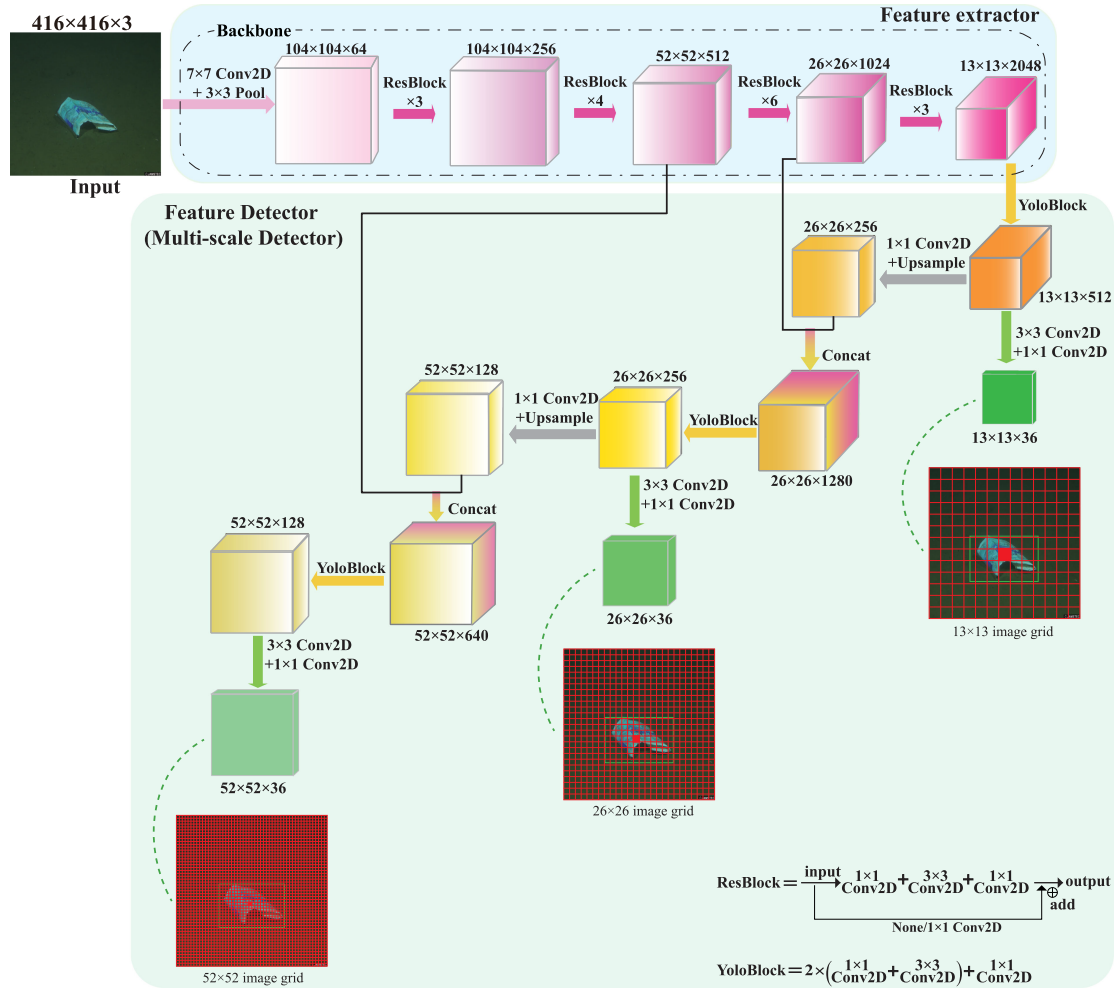


Fig. 1. ResNet50-YOLOV3 structure.

2) *Feature Extractor*: The feature extractor is the backbone. ResNet50 was adopted as the backbone of the network. ResNet50 consists of a series of residual blocks (the residual block is the ResBlock in the model in Fig. 1). Residual block mainly introduces a shortcut connection, so that the information of the previous residual block can flow into the next residual block without hindrance, which improves the flow of information. And it also avoids the vanishing gradient problem and the degradation problem caused by the deep network [27]. The residual block allows the strong feature extraction capabilities of the deep network to be reflected, which enables ResNet50 to extract more advanced features from deep-sea debris with complex features, thereby facilitating debris detection. From another perspective, the semantic information of the shallow features is strong, and the detailed information of the object is retained well. The shallow information is also directly transmitted backward through the shortcut connection of the residual block so that the detailed information of the object is retained, which is also beneficial for subsequent detection.

We avoid choosing a too deep residual network; even if the too deep residual network structure is stronger than ResNet50, it will bring too low detection speed to the detection network.

In addition, another motivation of this article is to explore the impact of different classical classification networks as backbone on the detection results. As a result, Darknet53—the original backbone of YOLOV3—has not been used for reference. Simultaneously, it allows the deep-sea debris detection algorithm to be migrated to various classic networks as much as possible to achieve easy deployment, instead of running on a specially designed backbone network.

3) *Feature Detector*:

a) *Feature pyramid network (FPN)*: The YOLOV3 method as the feature detector of this network is the core component. YOLOV3 is a multiscale detector since it uses the FPN [28] structure, as shown in Fig. 2. The FPN can combine low-resolution semantically strong features with high resolution, semantically weak features via a top-down pathway, and lateral connections and subsequently generate fusion features of different scales. These different dimensional features with enhanced high-level features and rich detailed information can have better feature expression, which is of great benefit to object detection. Based on this structure, YOLOV3 produced three scales of fusion features:  $13 \times 13$ ,  $26 \times 26$ ,  $52 \times 52$  and independently detected on the three scales of fusion feature maps using the

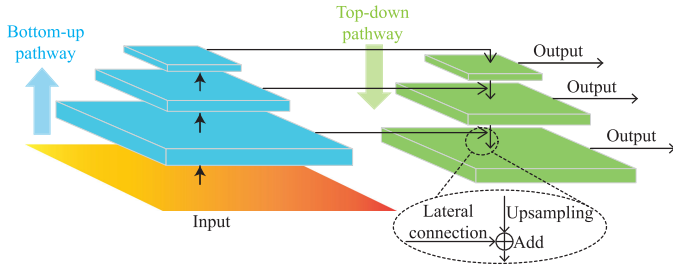


Fig. 2. FPN structure.

TABLE I  
ALLOCATION RULE OF THE DEFAULT ANCHORS

Feature map	$13_1 \hat{\times} 13$	$26_1 \hat{\times} 26$	$52_1 \hat{\times} 52$
Receptive field	big	medium	small
	$116 \times 90$	$30 \times 60$	$10 \times 13$
Assigned anchors	$156 \times 198$	$62 \times 45$	$16 \times 30$
	$373 \times 326$	$59 \times 119$	$33 \times 23$

Because the  $13 \times 13$  size feature has a larger receptive field, it is more suitable to be used to detect garbage objects with larger sizes, so larger anchors are assigned to them. The  $52 \times 52$  feature has the smallest receptive field, and the retained image detail information is the most complete, so smallest anchors are assigned to detect small-scale objects

prior boxes (also called anchors) mechanism. YOLOV3 adopts nine types of anchors and assigned three types of anchors to each predicted feature map, as described in Table I. YOLOV3 can achieve accurate detection of small- and large-scale debris objects with this multiscale detection method.

*b) Detection idea:* Specifically, the detection idea of YOLOV3 is to divide the input image into three grids of different sizes:  $13 \times 13$ ,  $26 \times 26$ , and  $52 \times 52$ , as described in Fig. 1. If the center position of the debris falls into a certain grid cell, this cell is responsible for detecting the object. The task assigned to the red cell is to predict the plastic in Fig. 1. Each cell predicts three bounding boxes by means of the assigned three anchors, and each bounding box has  $(5 + C)$  attributes. As depicted in Fig. 3, taking the  $13 \times 13$  grid as an example, 5 refers to the predicted coordinates  $t_x, t_y, t_w, t_h$  of the bounding box and the confidence  $p_0$  that the bounding box contains objects.  $C$  is equal to the number of categories ( $C = 7$  in this article), which refers to the possibility  $p_1-p_7$  that the objects contained in this bounding box belong to each category.  $p_0-p_7$  are all activated by the sigmoid function.

These attributes are integrated in the last three scale prediction feature maps:  $13 \times 13 \times 36$ ,  $26 \times 26 \times 36$ , and  $52 \times 52 \times 36$ , which are generated by encoding with  $3 \times 3$  and  $1 \times 1$  convolution kernels.

Directly predicting the coordinates of the bounding box will cause inaccuracy of the coordinates [25], [29]. YOLOV3 does not directly predict the exact coordinates of the bounding box, but predicts the offset  $t_x, t_y$  related to the upper left corner of the grid cell responsible for detecting the target, and the width  $t_w$  and height  $t_h$  of the bounding box relative to the anchors.

The final outputs of the predicted bounding box  $b_x, b_y, b_w$ , and  $b_h$  need to be refined as follows:

$$b_x = \sigma(t_x) + c_x \quad (1)$$

$$b_y = \sigma(t_y) + c_y \quad (2)$$

$$b_w = p_w e^{t_w} \quad (3)$$

$$b_h = p_h e^{t_h} \quad (4)$$

where  $c_x$  and  $c_y$  represent the coordinate position of the upper left corner of the grid where  $t_x$  and  $t_y$  are located, and  $p_w$  and  $p_h$  represent the width and the height of the anchor box corresponding to the predicted bounding box, respectively.  $\sigma$  is the sigmoid function, which scales  $t_x$  and  $t_y$  to between 0 and 1, thereby fixing  $b_x$  and  $b_y$  in the cell to avoid unstable prediction results.

Therefore, the final predicted bounding box we get is based on the anchors, but it is not necessarily equal to the anchors.

### B. Loss Metric

The intersection over union (IOU) is often used in the field of object detection to measure the similarity of two boxes, which can be described as

$$\text{IOU} = \frac{\text{area}(B_p \cap B_{gt})}{\text{area}(B_p \cup B_{gt})} \quad (5)$$

where  $B_p$  and  $B_{gt}$  represent the predicted bounding box and the ground-truth bounding box, respectively; the numerator represents the intersection area of the two boxes, and the denominator represents the union area of these boxes. The larger the IOU means the closer the two boxes are.

Since this network uses YOLOV3 as the detection method, the training strategy developed by YOLOV3 is adopted.

- 1) *Positive example:* Calculate the IOU between the ground-truth box and nine anchors corresponding to the cells where the center of the object is located. The anchor with the largest IOU is a positive example. Positive examples generate confidence loss, coordinate loss, and class loss.
- 2) *Negative example:* Anchors whose IOU with all ground-truth boxes is less than the threshold (0.5) are negative examples except positive examples. Negative cases only have confidence loss.
- 3) *Ignored example:* Anchors whose IOU with any ground-truth boxes is greater than the threshold (0.5) are ignored examples except positive examples. Ignored examples does not produce any loss.

Finally, the loss function of the network can be abstractly described as

$$\text{Loss} = \sum_{i=0}^{\text{all anchors}} (\text{Coord}_{\text{loss}} + \text{Conf}_{\text{loss}} + \text{Class}_{\text{loss}}) \quad (6)$$

where  $\text{Coord}_{\text{loss}}$  denotes the coordinate loss. The confidence loss is presented by  $\text{Conf}_{\text{loss}}$  and  $\text{Class}_{\text{loss}}$  is calculated for classifying loss. *all anchors* means the number of all anchors generated.



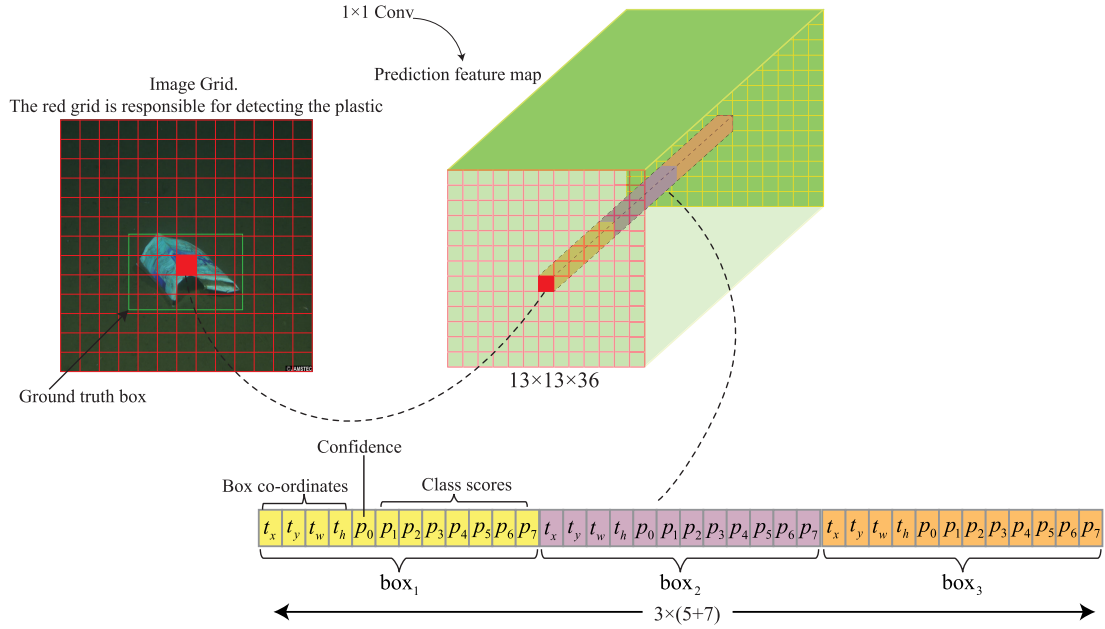


Fig. 3. On the  $13 \times 13$  feature map, the object is detected. The red cell is responsible for predicting the plastic debris. There are seven types of deep-sea debris.

### III. EXPERIMENTS

#### A. Dataset Description

The deep-sea debris database [23] provided by the Japan Agency for Marine-Earth Science and Technology (JAMSTEC) contains real deep-sea debris images and videos taken by submersibles “SHINKAI6500,” “HYPER-DOLPHIN,” etc. As shown in Fig. 4, the deep-sea debris in this database is widely distributed in the global ocean and is concentrated in the Western Pacific region. Some debris is located in the ocean depth of up to 10900 m. These videos and images in this database have not undergone secondary processing, fully showing the garbage in the real deep-sea environment. At the same time, the deep-sea garbage captured by the videos or images in this database has different forms due to the complexity of the seabed environment and light exposure. The above points mean that this database has the characteristics of authenticity and diversity.

The deep-sea debris database with few deep-sea garbage images but more deep-sea garbage videos is not suitable for direct image detection. For this reason, the deep-sea garbage images and videos downloaded from this online database need to be rearranged. We extract appropriate frames from these videos to combine with the original deep-sea garbage images to form a new deep-sea garbage image dataset. Then, the labelImg labeling tool [30] is used to label the newly formed image dataset to finally construct a deep-sea debris detection dataset called 3-D dataset.

The 3-D dataset has about 10000 images with dimensions of  $480 \times 320$  and contains seven types of debris images: cloth, fishing net and rope, glass, rubber, plastic, natural debris, and metal. Different from the previous marine garbage datasets, the 3-D dataset that inherits the characteristics of the database is derived from the real deep-sea environment and has multiple

TABLE II  
NUMBER OF OBJECTS IN THREE SCALES OF SMALL ( $\text{AREA} < 63^2$ ), MEDIUM ( $63^2 \leq \text{AREA} \leq 200^2$ ), AND LARGE ( $\text{AREA} > 200^2$ ) IN THE 3-D DATASET

Object scales	small	medium	large
Samples	3418	6764	4820

types of debris, which allows the detection algorithms trained on this dataset to be practically applied. In view of the discrete distribution characteristics of garbage individuals in the deep sea and the tendency of the database, it is more common that the image in the 3-D dataset contains a single garbage individual. Part of data of the dataset is shown in Fig. 5. It can be observed that this dataset has serious intraclass variability and interclass similarity [4]. The category distribution of objects in the dataset is depicted in Fig. 6. 3768 objects belong to the plastic category, which is the highest, followed by the cloth class. The least number is rubber and glass, 1285 and 1161, respectively. The distribution of debris scales in this dataset is described in Table II. The numbers of small-scale, medium-scale, and large-scale debris objects are 3418, 6764, and 4820, respectively.

A deep-sea garbage attribute database is also constructed based on the data provided by the database, as illustrated in Fig. 7, which divides the data into seven categories. Each category has the identification of the image or video containing the garbage belonging to the category and also includes the latitude and longitude of the garbage, the depth of the ocean, the shooting time, etc. It is convenient to follow up another research.

#### B. Comparative Methods

Faster R-CNN with the fastest speed and best accuracy in the two-stage detection networks is used as a comparison method.

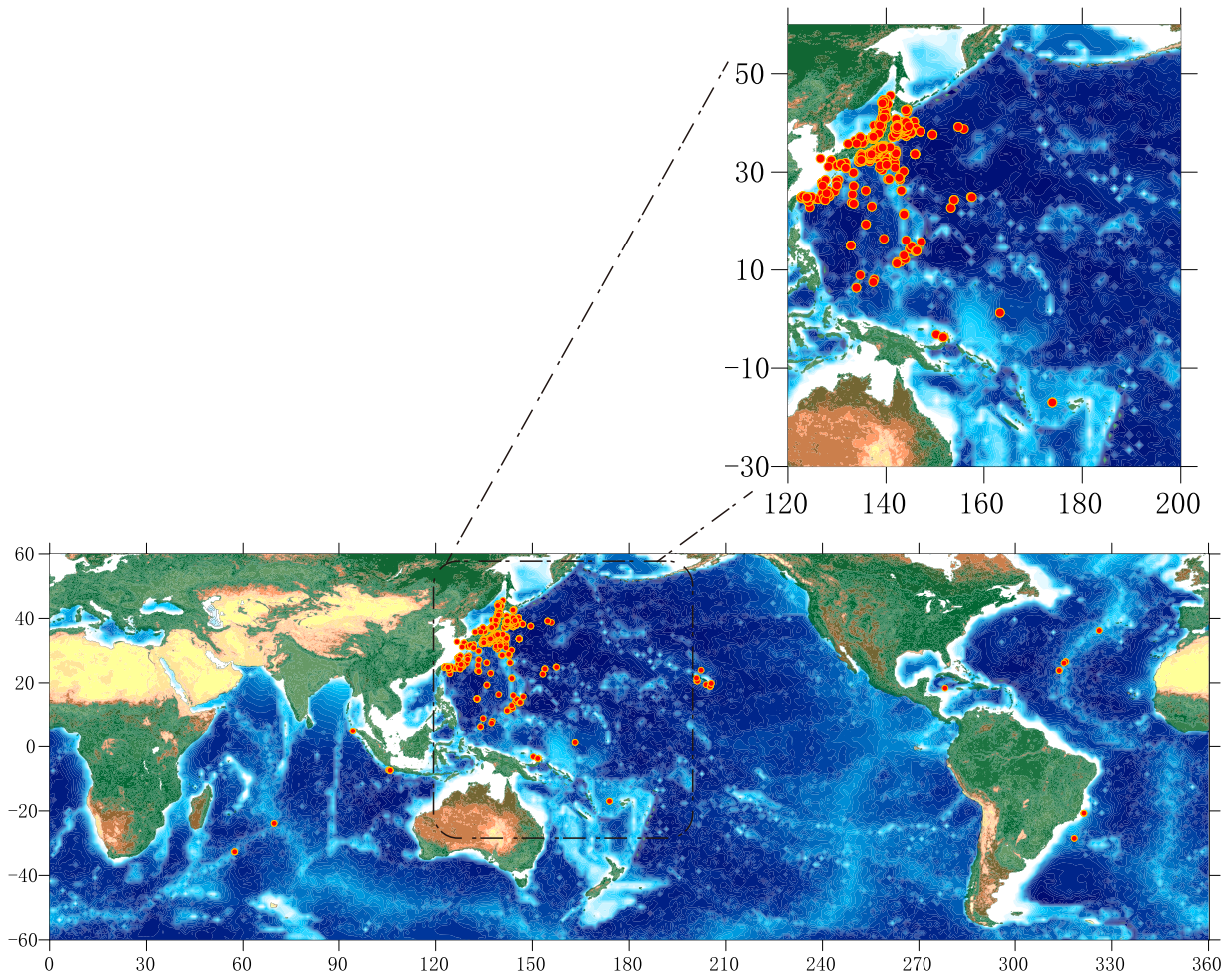


Fig. 4. Global distribution map and concentrated distribution area map of the deep-sea debris database.

Faster R-CNN abandons the previous manual selection of candidate proposals by R-CNN and Fast R-CNN, while it adopts the region proposal network in the first stage, which uses nine anchors at each point on the last feature map of the backbone to automatically generate candidate proposals, and ultimately regresses and classifies these proposals in the second stage. In addition, SSD is also used as a comparison method. It uses a pyramidal feature hierarchy [28] to generate six different features from shallow to deep, and the six feature maps use four, six, six, six, four, and four anchors successively to directly predict the results. SSD as a one-stage detection network has surpassed the detection accuracy of the two-stage detection networks for the first time, and the detection speed far exceeds the two-stage nets. Finally, in order to explore the influence of different backbone structures on detection speed and accuracy, VGG16 [31], MobileNetV2 [32], and ResNet50 are chosen as the backbone network of each detector, as described in Table III. VGG16 is a basic CNN composed of a convolutional layer and a pooling layer, with a total of 16 layers. It is a network with a simple structure and strong applicability, which is often used in various experiments. MobileNetV2 is a lightweight network that uses a deep separation convolution structure and is famous for its fast speed and better accuracy. The parameters of each

TABLE III  
EIGHT COMPARATIVE METHODS AND RESNET50-YOLOV3

Models	Parameters
VGG16-Faster R-CNN	136811934
MobileNetV2-Faster R-CNN	3040606
ResNet50-Faster R-CNN	28336798
VGG16-SSD	24547880
MobileNetV2-SSD	6380936
ResNet50-SSD	28807272
VGG16-YOLOV3	35440556
MobileNetV2-YOLOV3	23241900
ResNet50-YOLOV3	45263852

model are also depicted in Table III for reference for platform transplanting.

### C. Experimental Results

1) *Experiment Settings*: Models are built on Keras and run on the computer, which has a GeForce GTX 1080Ti GPU with a capacity of 11 G. 85% of the 3-D dataset is used for training, and 15% is used for testing. Data augmentation technology is



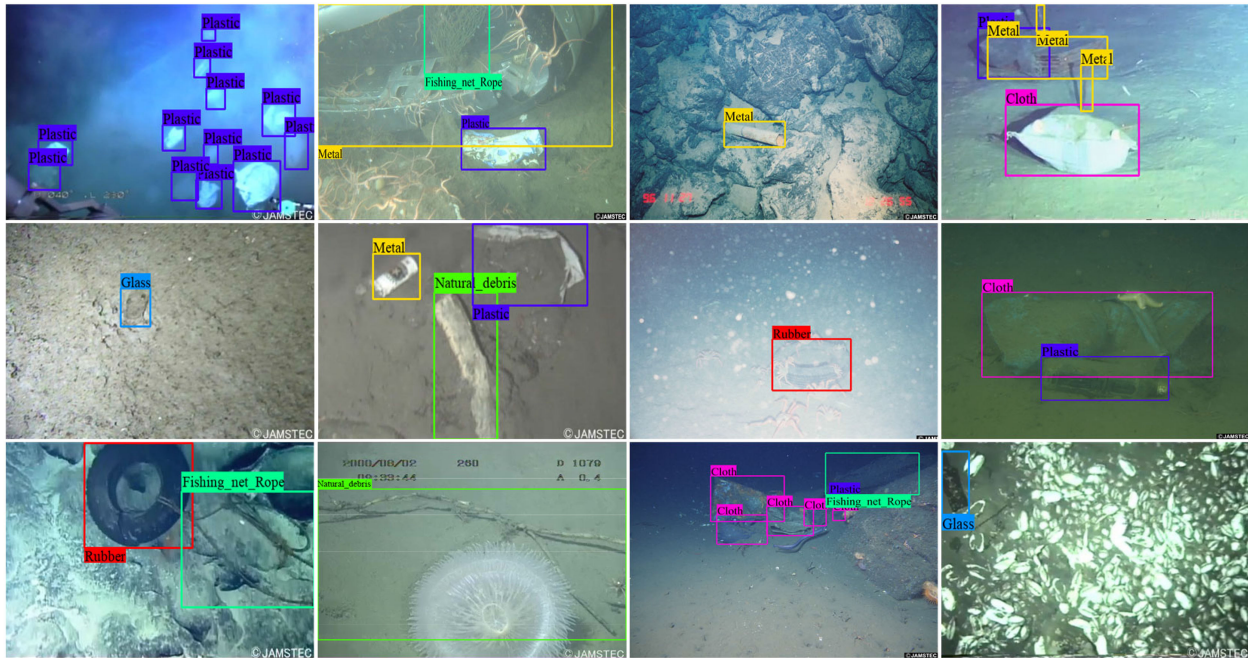


Fig. 5. Sample examples of the 3-D dataset.

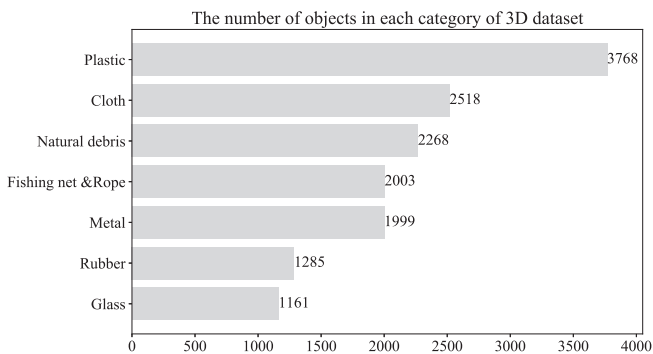


Fig. 6. Number of each type of garbage in this dataset. It contains 15 000 objects in about 10 000 pictures.

taken to avoid overfitting. Adam was selected as the optimizer for the models. For Faster R-CNN and SSD, the input image size is set to 600 and 512, respectively. The large input resolution is beneficial to the detection result [16], [29]. The scales and aspect ratios of the anchor boxes adopt the default settings of each detection network, which is feasible because the shape and size of the deep-sea debris in our 3-D dataset are basically the same as the shape and size of the objects in the COCO dataset and the PASCAL VOC dataset that made these default anchors. We have done experiments to formulate anchors through the  $k$ -means clustering method [24], [29], but the effect is not good. It can be said that these default anchors have a very good multiscale nature. Nonmaximum suppression (NMS) is adopted, which can eliminate low-confidence predicted boxes whose IOU values with other predicted boxes are higher than the threshold. The threshold is usually set to 0.5 to avoid overculling boxes and excessive redundancy.

In the training process, transfer learning is adopted, and all training is divided into two steps. In the first step, the pretraining weights of the backbone on ImageNet are loaded and are frozen to train the deeper layer of the detection network. This can speed up the convergence speed of the models because the shallow features are universal. In the second step, all layers of the model are trained to fine-tune detection networks, which will produce our final training results. When enduring the past three epochs and the performance of the model does not improve, the action that the learning rate decays by a factor of 0.3 will be triggered. Training will be terminated early when there is no improvement in model performance after ten iterations. The initial learning rate of  $1e-4$  and the batch size of 10 are used to fine-tune the ResNet50-YOLOV3 model.

2) *Experimental Results and Discussion:* MAP is the average value of AP of all classes and usually used to measure the detection quality of different models. AP is the average precision of all recall values between 0 and 1, describing the area under the precision–recall curve. The definition of precision  $\rho$  and recall rate  $r$  is

$$\rho = \frac{TP}{TP + FP} \quad (7)$$

$$r = \frac{TP}{TP + FN} \quad (8)$$

where TP means the number of predicted bounding boxes with IOU greater than the threshold  $\kappa$ , FN represents the number of predicted bounding boxes with IOU less than or equal to threshold  $\kappa$  (or the number of redundant predicted bounding boxes matching the same ground-truth bounding box), and FN is the number of ground-truth bounding boxes that are not detected.

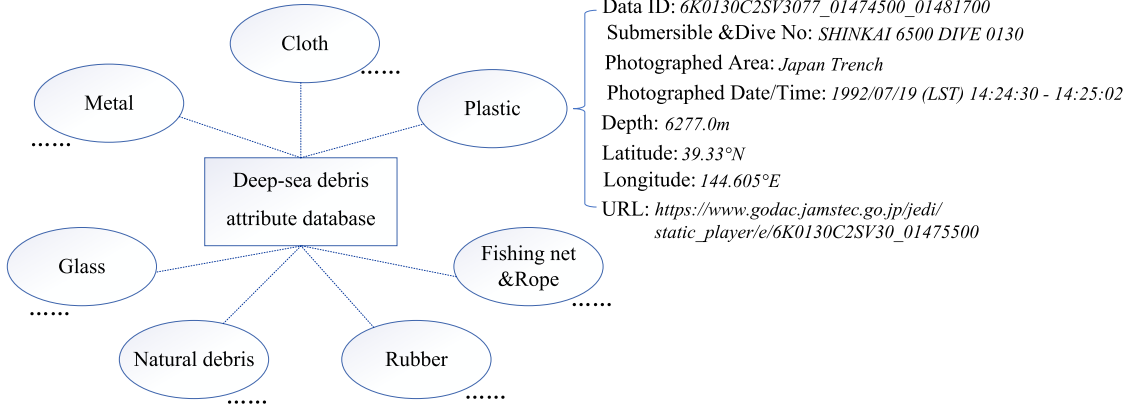


Fig. 7. Deep-sea debris attribute database.

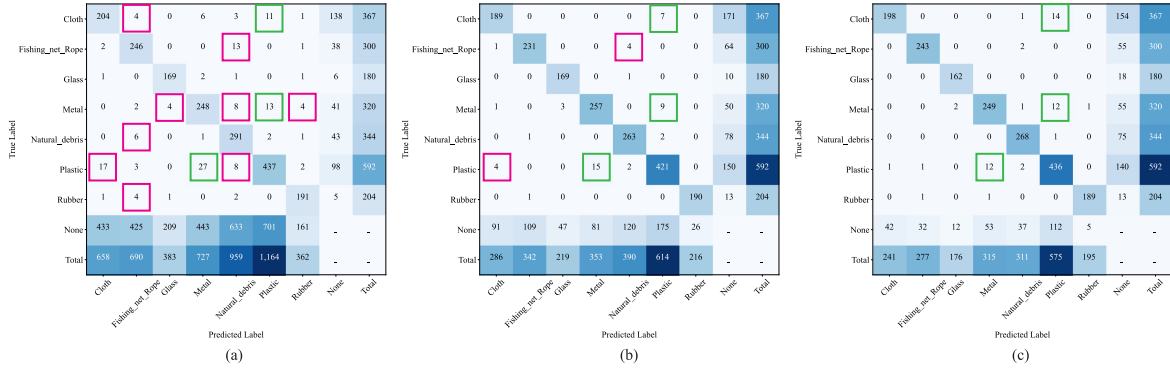


Fig. 8. Confusion matrices of ResNet50-SSD/YOLOV3/Faster R-CNN. Since predicted boxes with low confidence are often not helpful to the detection results, the boxes with a confidence lower than 0.3 were eliminated first, and then, the final predicted boxes are output by NMS processing (the same is true for the results in Figs. 9 and 10). The matching principle of the predicted bounding box in the confusion matrix: if the IOU of the ground-truth box and the predicted box is greater than or equal to 0.5, a match is found, and the predicted box is assigned its true label; if there are repeated matches, the best match is always selected (the larger IOU). Objects belonging to ground truth but not detected are included in the *None* column of the matrix; objects detected but not belonging to the confusion matrix are included in the *None* row of the matrix. (a) ResNet50-Faster R-CNN. (b) ResNet50-SSD. (c) ResNet50-YOLOV3.

Therefore, AP can be described as

$$AP = \sum_{i=0}^{n-1} (r_{i+1} - r_i) \rho_{\text{interp}}(r_{i+1}) \quad (9)$$

$$\rho_{\text{interp}}(r_{i+1}) = \max_{\tilde{r}: \tilde{r} \geq r_{i+1}} \rho(\tilde{r}) \quad (10)$$

where  $\rho(\tilde{r})$  is precision when recall rate is  $\tilde{r}$ .

Table IV shows the detection results of the models under different indicators. A high MAP value means a good detection effect. Obviously, MobileNetV2 as a backbone cannot achieve a good detection effect of deep-sea debris since it is a lightweight network that abandons accuracy and pursues speed [32]. Although the detection models using it as a backbone are very fast, it is meaningless. In addition, different from the outstanding performance of ResNet50/VGG16-YOLOV3, detection ability of MobileNetV2-YOLOV3 is not good, which is worse than MobileNetV2-SSD and MobileNetV2-Faster R-CNN that also use MobileNetV2 as the backbone. In effect, MobileNetV2-Faster R-CNN and MobileNetV2-SSD did not allow the shallow

TABLE IV  
VARIOUS EVALUATION INDICATORS FOR DIFFERENT MODELS

Models	MAP <sub>0.5</sub>	MAP <sub>0.5:0.95</sub>	MAP <sub>0.75</sub>	FPS
MobileNetV2-Faster R-CNN	65.3	36.2	35.7	16
VGG16-Faster R-CNN	71.2	41.9	44.1	20
ResNet50-Faster R-CNN	71.9	42.3	44.6	12
MobileNetV2-SSD	60.1	37.2	41.9	21
VGG16-SSD	71.2	43.7	49.1	23
ResNet50-SSD	78.7	47.7	53.7	17
MobileNetV2-YOLOV3	59	33.2	34.1	<b>37</b>
VGG16-YOLOV3	82.4	48.1	51.7	35
<b>ResNet50-YOLOV3</b>	<b>83.4</b>	<b>48.4</b>	<b>53.8</b>	30

MAP<sub>0.5</sub> (PASCAL VOC metric) is the MAP when the IOU threshold  $\kappa$  is 0.5, MAP<sub>0.5:0.95</sub> (COCO metric) is the mean value of the MAPs when  $\kappa$  is 0.5, 0.55, 0.60, 0.90, and 0.95. MAP<sub>0.75</sub> (strict metric) represents the MAP when  $\kappa$  is 0.75. FPS clarifies the number of pictures that the model can detect per second.

layers of the MobileNetV2 backbone to participate in prediction, which indicates that it is not feasible for MobileNetV2-YOLOV3 to allow the shallow features of MobileNetV2 to



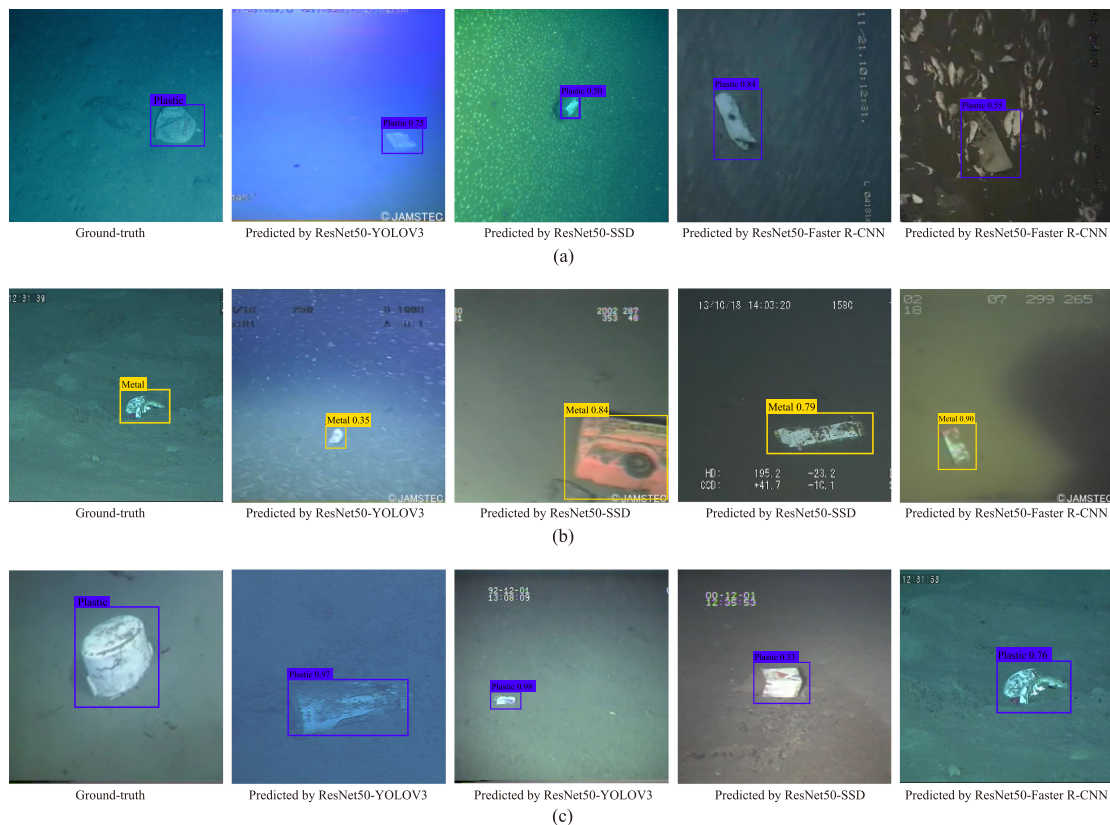


Fig. 9. Showcases of misdetection of cloth, plastic, and metal for ResNet50-SSD/YOLOV3/Faster R-CNN. In order to facilitate the description of the misdetection, only the detection case of a single object is shown here. The leftmost column displays the ground-truth box that belongs to the category which other objects are misclassified as. (a) Cloth that is detected to be plastic. (b) Plastic that is detected to be metal. (c) Metal that is detected to be plastic.

participate in prediction. This is because compared with VGG16 and ResNet50, the MobileNetV2 network tends to compress the feature channels very low to pursue lightweight, which weakens to a certain extent the characterization ability of shallow features. These weak features or useless information may cause a certain degree of interference to the high-level features after being fused with the high-level features.

ResNet50-YOLOV3 achieved the highest garbage detection results, with  $MAP_{0.5}$ ,  $MAP_{0.5:0.95}$ , and  $MAP_{0.75}$  as high as 83.4, 48.4, and 53.8, respectively. The high  $MAP_{0.5:0.95}$  and  $MAP_{0.75}$  indicate that ResNet50-YOLOV3 has a good ability to predict boundary coordinates, which means that it can more accurately frame deep-sea debris. The main reason is that with the support of ResNet50's strong feature extraction capabilities, YOLOV3 uses three different sizes of anchors on the three different scales of fusion feature maps to detect, which can make the deep-sea garbage of different sizes accurately detectable. Furthermore, the predicted box is also limited to the grid cell, which also avoids excessive offset of the predicted boxes, making the boundary prediction of the boxes more reasonable. SSD also uses different anchors on multiple feature maps of different sizes to detect deep-sea garbage, so the prediction of ResNet50-SSD is also considerable. However, even if SSD uses multiscale anchors on multiple feature maps, these feature maps have not undergone information fusion, resulting in a certain loss of shallow detail information [28], which has caused certain

obstacles to the accurate prediction of the boxes. Moreover, the SSD does not impose constraints on the prediction of coordinates, and the change of the predicted box is too large, which also leads to the inaccurate prediction of the box boundary by the SSD. Faster R-CNN only uses anchors in final features to detect deep-sea garbage. It does not integrate low-level features, which makes some details of the seabed garbage lose a lot, and the scale of anchors is simpler than that of SSD and YOLOV3, which is more unfavorable for the detection of deep-sea garbage. It should also be noted that these results achieved by SSD and Faster R-CNN are performed at a higher input image resolution (SSD:  $512 \times 512$ ; Faster R-CNN:  $600 \times 600$ ). Actually, the high resolution greatly improves the detection accuracy [16], [29]. In contrast, YOLOV3 with the lowest resolution input image ( $416 \times 416$ ) only uses three feature maps and nine anchors to achieve excellent submarine garbage detection results.

One-stage networks such as SSD and YOLOV3 have an inherent advantage in detection speed because they abandon the candidate region extraction stage of the two-stage networks. The detection speed of MobileNetV2/VGG16/ResNet50-YOLOV3 for submarine garbage crushes other networks. ResNet50-YOLOV3 achieves the best detection effect of deep-sea debris while also achieving a detection speed of 30 FPS among these models. It can be said that ResNet50-YOLOV3 achieves a balance between detection accuracy and speed.

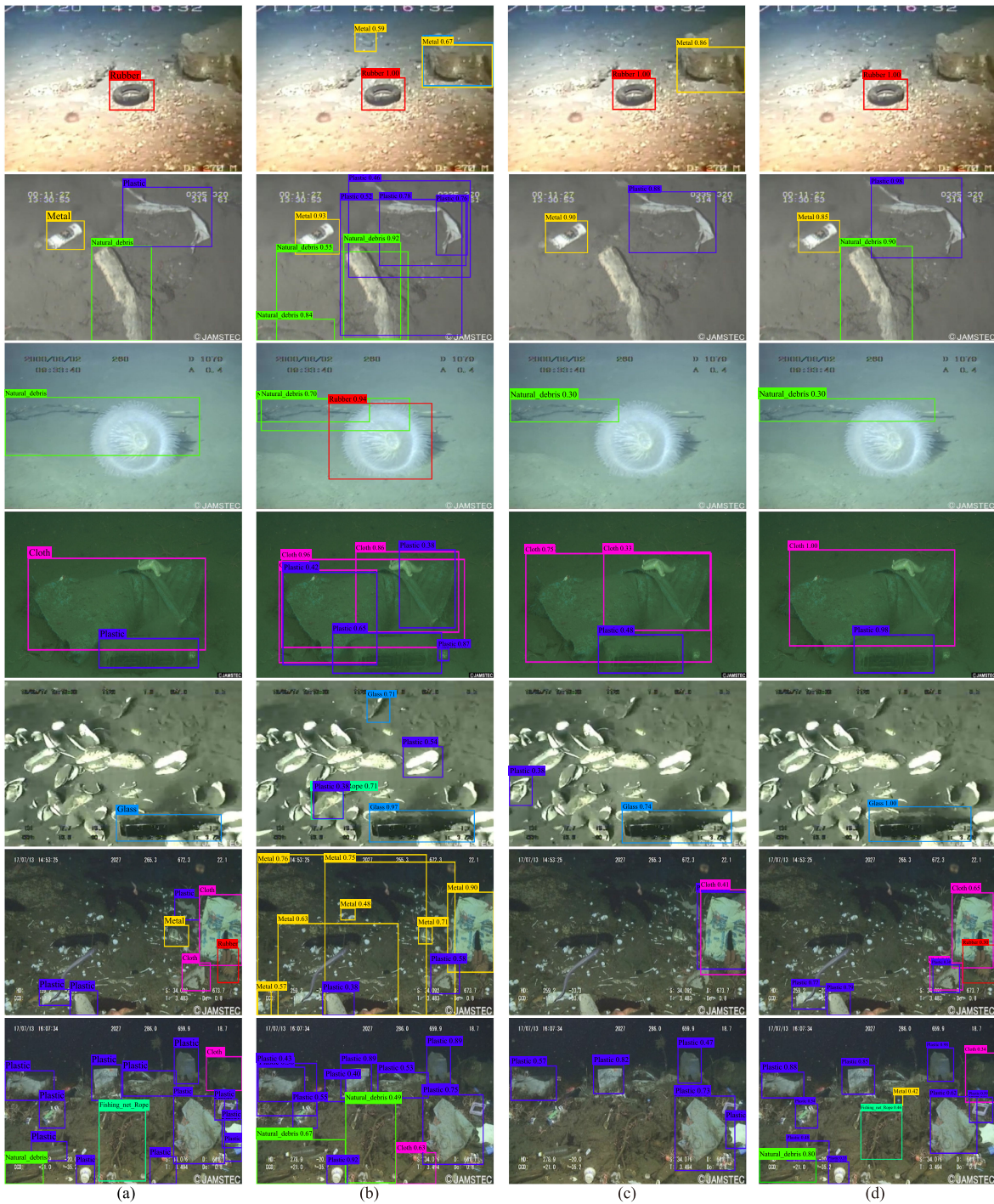


Fig. 10. Difference between the three models in the detection of deep-sea debris. The leftmost column is the real label. (a) Ground truth. (b) ResNet50-Faster R-CNN. (c) ResNet50-SSD. and (d) ResNet50-YOLOV3.

For the sake of viewing the detection of each type of deep-sea debris for each model, we show the AP value of each type of model.

It can be seen from the Table V that ResNet50-YOLOV3 almost achieved the highest AP value of all classes and achieved satisfactory detection results.

The rubber category is the easiest to detect. ResNet50-YOLOV3 has AP values of 97.6 for this category. Although

the number of rubber categories in the 3-D dataset is the least, however, due to its single shape and scale, it is easy for models to accurately detect it.

VGG16-YOLOV3 has the highest AP value for glass, reaching 93.6. VGG16 is a relatively simple network model that employs  $3 \times 3$  convolution for feature extraction. It has more reservations about target details. This is conducive to the detection of small objects such as glass. Moreover, the FPN of



TABLE V  
AP PER CLASS FOR THESE MODELS

Model	Cloth	Fishing net & Rope	Glass	Metal	Natural debris	Plastic	Rubber
MobileNetV2-Faster R-CNN	36.5	70.9	82.4	62.8	61.2	54.2	89
VGG16-Faster R-CNN	44.1	73.5	85.7	68.7	65	67.7	93.3
ResNet50-Faster R-CNN	45.5	76.3	87.3	68.9	70.7	62	92.5
MobileNetV2-SSD	31.0	58.4	82.3	50.9	58.4	53.7	86.2
VGG16-SSD	46.8	72.6	88.6	64.9	67.9	64.6	92.6
ResNet50-SSD	52.3	76.9	92.8	83.4	74.5	74.4	96.6
MobileNetV2-YOLOV3	31.4	64.4	80	46.3	60.1	46.3	82.1
VGG16-YOLOV3	61.1	83.7	<b>93.6</b>	84.6	79.7	77.9	96.1
<b>ResNet50-YOLOV3</b>	<b>61.7</b>	<b>86</b>	91.6	<b>85.2</b>	<b>82.5</b>	<b>79.4</b>	<b>97.6</b>

The AP here refers to AP<sub>0.5</sub>.

The bold entities means that this method has the best result of the comparison methods.

YOLOV3 makes the detailed information of the glass class merge and fuse with the advanced features of the class, which makes VGG16-YOLOV3 an outstanding detection effect on small-scale objects such as glass [28]. Like the rubber class, the shape, appearance, scale, and other characteristics of the glass class in the 3-D dataset have great uniformity, so all models are not inferior to the detection effect of this class.

Each model has the lowest AP value for cloth and plastic, which shows that there is a little difficulty in detecting cloth and plastic debris, even if the number of cloth and plastic debris in the 3-D dataset is the largest. The main reason is that deep-sea debris has huge intraclass variability and interclass similarity [4], which is discussed in conjunction with following confusion matrices.

As shown in Fig. 8, we select a network with the best performance from each type of detector (Faster R-CNN, SSD, and YOLOV3): ResNet50-Faster R-CNN, ResNet50-SSD, and ResNet50-YOLOV3, and show their deep-sea debris detection confusion matrices. The cases where the number of false detections is greater than 3 (not including *None*) are circled with boxes. Obviously, the false detection of ResNet50-Faster R-CNN is the most serious. The green boxes circled the common points of the three models' error detection. Fig. 9 is used to illustrate this case. Apparently, the shapes, textures, and other characteristics between these categories have strong similarity, and even, we cannot seem to distinguish the differences between them easily. The model cannot capture a characteristic boundary between them since the feature difference between these categories is not obvious. For other categories (especially glass and rubber), the features between categories are highly distinguishable, and the features within the category are specialized, so models can find the optimal boundary hyperplane in their feature space and the misdetection situation will be alleviated a lot.

Although the number of samples recalled by ResNet50-Faster R-CNN is large, as shown in the confusion matrix in Fig. 8(a), the price is that it generates too many false predicted boxes. It produced 658 predicted boxes labeled cloth, 690 labeled fishing net and rope, 383 labeled glass, 727 labeled metal, 959 labeled natural debris, 1164 labeled plastic, and 362 labeled rubber class, which are much more than the SSD and the YOLOV3.

TABLE VI  
F1 AND KAPPA CALCULATED ACCORDING TO THE THREE CONFUSION MATRICES OF FIG. 8

Models	ResNet50-Faster R-CNN	ResNet50-SSD	ResNet50-YOLOV3
Cloth	39.8	57.7	<b>65.1</b>
Fishing net & Rope	49.7	72	<b>84.2</b>
Glass	60	84.7	<b>91</b>
Metal	47.4	76.4	<b>78.4</b>
Natural debris	44.7	71.7	<b>81.8</b>
Plastic	49.8	69.8	<b>74.7</b>
Rubber	67.5	90.5	<b>94.7</b>
<b>Kappa</b>	0.907	0.966	<b>0.966</b>

The bold entities means that this method has the best result of the comparison methods.

This problem is depicted in Fig. 10. It can be seen from the figure that ResNet50-Faster R-CNN generates a lot of predicted boxes to frame all seabed debris objects as much as possible, which to a certain extent gives it a higher recall rate. However, more boxes are redundant boxes and error-detected boxes, which leads to a serious low precision of ResNet50-Faster R-CNN. ResNet50-YOLOV3 uses a more conservative approach, which avoids the generation of redundant boxes while maintaining a high detection level. The F1 value of the three detection models and the kappa value of the confusion matrix of Fig. 8 are described in Table VI. It is obvious that the comprehensive detection ability of ResNet50-YOLOV3 for deep-sea debris is still optimal.

#### IV. CONCLUSION

Deep-sea debris detection using deep learning methods has been studied in this article. Given the existing problems of deep-sea debris detection, the 3-D dataset containing seven types of deep-sea debris and the deep-sea debris attribute database is established. The eight detection models are compared with our proposed method, and the following conclusions can be drawn through experimental analysis.

- 1) It is necessary to create a 3-D dataset for deep-sea debris detection, which is conducive to the in-depth development of subsequent deep-sea debris detection. At the same time, deep-sea debris is affected by the special deep-sea environment, and the garbage has strong intraclass diversity and interclass similarity.
- 2) Compared with the other eight methods, ResNet50-YOLOV3 not only has good deep-sea garbage detection capabilities, but also maintains a faster detection speed. In addition, different backbones also have a significant impact on the seabed garbage detection effect of the model. MobileNetV2 is not suitable as the detection backbone, and ResNet50 is more suitable as the backbone than VGG16.
- 3) Compared with other categories, plastic and cloth, metal and plastic are more severely shuffled between the two. ResNet50-YOLOV3 can obtain the best comprehensive detection capabilities for deep-sea debris while maintaining a low level of confusion.

In this article, detecting various types of deep-sea debris using deep learning is the first to be carried out. In the future, improving the detection capabilities of deep-sea debris using deep learning methods will be considered, and on the basis of the research in this article, the situation of misdetection of submarine debris will be further solved. At the same time, it is also considered to actually apply detection methods to AUVs working in the deep sea to help detect and clean up garbage. Finally, debris detection in videos is also a future research direction.

#### ACKNOWLEDGMENT

The authors are grateful to JAMSTEC for providing the deep-sea debris database, anonymous reviewers for helpful and constructive comments on this manuscript.

#### REFERENCES

- [1] I. M. J. van den Beld, B. Guillaumont, L. Menot, C. Bayle, S. Arnaud-Haond, and J.-F. Bourillet, "Marine litter in submarine canyons of the Bay of Biscay," *Deep-Sea Res. II—Topical Stud. Oceanography*, vol. 145, pp. 142–152, Nov. 2017.
- [2] W. C. Li, H. F. Tse, and L. Fok, "Plastic waste in the marine environment: A review of sources, occurrence and effects," *Sci. Total Environ.*, vol. 566, pp. 333–349, Oct. 2016.
- [3] H. Yin and C. Cheng, "Monitoring methods study on the great pacific ocean garbage patch," in *Proc. Int. Conf. Manage. Service Sci.*, Aug. 2010, pp. 1–4.
- [4] M. Valdenegro-Toro, "Deep neural networks for marine debris detection in sonar images," May 2019, *arXiv:1905.05241*.
- [5] G. Goncalves, U. Andriolo, L. Goncalves, P. Sobral, and F. Bessa, "Quantifying marine macro litter abundance on a sandy beach using unmanned aerial systems and object-oriented machine learning methods," *Remote Sens.*, vol. 12, no. 16, Aug. 2020, Art. no. 2599.
- [6] K. Topouzelis, D. Papageorgiou, A. Karagaitanakis, A. Papakonstantinou, and M. A. Ballesteros, "Plastic litter project 2019: Exploring the detection of floating plastic litter using drones and Sentinel 2 satellite images," in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, Sep. 2020, pp. 6329–6332.
- [7] K. Schlining *et al.*, "Debris in the deep: Using a 22-year video annotation database to survey marine litter in Monterey Canyon, Central California, USA," *Deep-Sea Res. I—Oceanograph. Res. Papers*, vol. 79, pp. 96–105, Sep. 2013.
- [8] S. Chiba *et al.*, "Human footprint in the abyss: 30 year records of deep-sea plastic debris," *Mar. Policy*, vol. 96, pp. 204–212, Oct. 2018.
- [9] M. Kremezi *et al.*, "Pansharpening PRISMA data for marine plastic litter detection using plastic indexes," *IEEE Access*, vol. 9, pp. 61955–61971, 2021.
- [10] C. Su, W. Dongxing, L. Tiansong, R. Weichong, and Z. Yachao, "An autonomous ship for cleaning the garbage floating on a lake," in *Proc. 2nd Int. Conf. Intell. Comput. Technol. Autom.*, Oct. 2009, vol. 3, pp. 471–474.
- [11] M. Arii, M. Koiwa, and Y. Aoki, "Applicability of SAR to marine debris surveillance after the great East Japan Earthquake," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 7, no. 5, pp. 1729–1744, May 2014.
- [12] T. Acuna-Ruz *et al.*, "Anthropogenic marine debris over beaches: Spectral characterization for remote sensing applications," *Remote Sens. Environ.*, vol. 217, pp. 309–322, Nov. 2018.
- [13] K. Themistocleous, C. Papoutsas, S. Michaelides, and D. Hadjimitsis, "Investigating detection of floating plastic litter from space using Sentinel-2 imagery," *Remote Sens.*, vol. 12, no. 16, Aug. 2020, Art. no. 2648.
- [14] Z. Ge, H. Shi, X. Mei, Z. Dai, and D. Li, "Semi-automatic recognition of marine debris on beaches," *Sci. Rep.*, vol. 6, 2016, Art. no. 25759.
- [15] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 6, pp. 1137–1149, Jun. 2017.
- [16] W. Liu *et al.*, "SSD: Single shot MultiBox detector," in *Proc. Eur. Conf. Comput. Vis.*, 2016, vol. 9905, pp. 21–37.
- [17] M. Everingham, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman, "The Pascal Visual Object Classes (VOC) challenge," *Int. J. Comput. Vis.*, vol. 88, no. 2, pp. 303–338, Jun. 2010.
- [18] T.-Y. Lin *et al.*, "Microsoft COCO: Common objects in context," in *Proc. Eur. Conf. Comput. Vis.*, 2014, vol. 8693, pp. 740–755.
- [19] M. Valdenegro-Toro, "Submerged marine debris detection with autonomous underwater vehicles," in *Proc. Int. Conf. Robot. Autom. Humanitarian Appl.*, Dec. 2016, pp. 1–7.
- [20] J. I. Watanabe, Y. Shao, and N. Miura, "Underwater and airborne monitoring of marine ecosystems and debris," *J. Appl. Remote Sens.*, vol. 13, no. 4, 2019, Art. no. 044509.
- [21] M. Fulton, J. Hong, M. I. Jahidul, and J. Sattar, "Robotic detection of marine litter using deep visual detection models," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2019, pp. 5752–5758.
- [22] B. Xue, B. Huang, G. Chen, H. Li, and W. Wei, "Deep-sea debris identification using deep convolutional neural networks," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 14, pp. 8909–8921, 2021.
- [23] "Deep-sea debris database," Oct. 2018. [Online]. Available: <http://www.godac.jamstec.go.jp/catalog/dsdebris/e/index.html>
- [24] J. Redmon and A. Farhadi, "YOLOv3: An incremental improvement," 2018, *arXiv:1804.02767*.
- [25] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 580–587.
- [26] T. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 2, pp. 318–327, Feb. 2020.
- [27] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 770–778.
- [28] T.-Y. Lin *et al.*, "Feature pyramid networks for object detection," in *Proc. 30th IEEE Conf. Comput. Vis. Pattern Recognit.*, Jul. 2017, pp. 936–944.
- [29] J. Redmon and A. Farhadi, "YOLO9000: Better, faster, stronger," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jul. 2017, pp. 6517–6525.
- [30] Tzutalin/LabelImg, Git Code, 2015.
- [31] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2015, *arXiv:1409.1556*.
- [32] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L. Chen, "MobileNetV2: Inverted residuals and linear bottlenecks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 4510–4520.



**Bing Xue** is working toward the B.S. degree in computer application technology with Qingdao University, Qingdao, China.

His research interests include image processing and analysis and artificial intelligence.





**Baoxiang Huang** (Member, IEEE) received the B.S. degree in traffic engineering from the Shandong University of Technology, Zibo, China, in 2002, the M.S. degree in mechatronic engineering from Shandong University, Jinan, China, in 2005, and the Ph.D. degree in computer engineering from the Ocean University of China, Qingdao, China, in 2011.

She was an Academic Visitor with Nottingham University, Nottingham, U.K. She is currently an Associate Professor with the College of Computer Science and Technology, Qingdao University, Qingdao.

Her research interests include remote sensing image processing and analysis, big data oceanography, and artificial intelligence.



**Haitao Li** received the Doctorate of Science degree in cartography and geographic information systems from the Ocean University of China, Qingdao, China, in 2007.

In 2007, he joined the Qingdao University of Science and Technology, Qingdao, as a Professor. His research interests include digital ocean, digital environmental protection and security, smart city informatization, geographic information systems, positioning and navigation systems (Beidou and GPS), mobile Internet of Things, and software engineering.



**Weibo Wei** received the Ph.D. degree in weapon launch theory and technology from the Nanjing University of Science and Technology, Nanjing, China, in 2006.

He is currently an Associate Professor with the College of Computer Science and Technology, Qingdao University, Qingdao, China. His research interests include image processing and analysis, big data, and artificial intelligence.

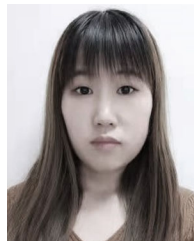


**Ge Chen** received the B.S. degree in marine physics, the M.S. degree in satellite oceanography, and the Ph.D. degree in physical oceanography from the Ocean University of China (OUC), Qingdao, China, in 1998, 1990, and 1993, respectively.

From 1994 to 1996, he was a Postdoctoral Fellow with the French Research Institute for the Exploitation of the Sea, Brest, France. Since 1997, he has been a Professor of Satellite Oceanography and Meteorology with the Ocean University of China. He is the Deputy Dean of the Institute for Advanced Marine

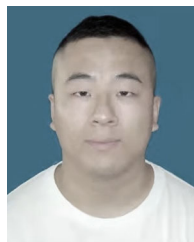
Sciences, OUC, and the Chief Scientist for Ocean Science Satellite Missions with the National Laboratory of Ocean Science and Technology, Qingdao. He is the author of more than 110 peer-reviewed scientific papers published in internationally recognized journals. His current research interests include satellite remote sensing of the ocean and big data oceanography.

Dr. Chen received the National Science Fund for Outstanding Young Scientists awarded by the Natural Science Foundation of China and became the Chair Professor of Cheung Kong Scholars Program nominated by the Chinese Ministry of Education in 2001. From 1998 to 2002, he was the Executive Secretary of the International Pan Ocean Remote Sensing Conference Association.



**Nan Zhao** is working toward the B.S. degree in computer science and technology with Qingdao University, Qingdao, China.

Her research interests include image processing and analysis and artificial intelligence.



**Hongfeng Zhang** is working toward the B.S. degree in computer application technology with Qingdao University, Qingdao, China.

His research interests include image processing and analysis and artificial intelligence.