Hindawi
Mathematical Problems in Engineering
Volume 2020, Article ID 3937580, 11 pages
https://doi.org/10.1155/2020/3937580

Research Article

Marine Organism Detection and Classification from Underwater Vision Based on the Deep CNN Method

Fenglei Han, Jingzheng Yao, Haitao Zhu, and Chunhui Wang

College of Shipbuilding Engineering, Harbin Engineering University, Harbin, China

Correspondence should be addressed to Jingzheng Yao; yaojingzheng_heu@163.com

Received 25 November 2019; Accepted 22 January 2020; Published 25 February 2020

Academic Editor: Alessandro Gasparetto

Copyright © 2020 Fenglei Han et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Seabed fishing depends on humans in common, for instance, the sea cucumber, sea urchin, and scallop fishing, which is always a very dangerous task. Considering the underwater complex environment conditions such as low temperature, dim vision, and high pressure, collecting the marine products using underwater robots is commonly regarded as a feasible solution. The key technique of the underwater robot development is to detect and locate the main target from underwater vision. This research is based on the deep convolutional neural network (CNN) to realize the target recognition from underwater vision. The RPN (Region Proposal Network) is used to optimize the feature extraction capability. Deep learning dataset is prepared using an underwater video obtained from a sea cucumber fishing ROV (Remote Operated Vehicle). The inspiration of the network structure and the improvements come from the Faster RCNN and Hypernet method, and for the underwater dataset, the method proposed in this paper shows a good performance of recall and object detection accuracy. The detection runs with a speed of 17 fps on a GPU, which is applicable to be used for real-time processing.

1. Introduction

Sea cucumbers, sea urchins, scallops, and other marine organisms live in the bottom of the sea, which are mainly fished by humans, and the fishing process is dangerous. Nowadays, the underwater robots are commonly used to replace humans due to dangerous operation. Real-time object recognition is the key technology of the fishing robot, and the underwater environment is complex and changeable, which brings many difficulties to real-time object recognition. At present, underwater fishing robots are mostly operated by humans using a remote control device, the fishing speed is slow, and the image information collected by using an underwater camera is generally vague, so it is very difficult to detect the position of sea cucumber and something else by human eyes. Image recognition techniques based on convolution neural networks can surpass the function of human eyes, which can be completed more accurately when the sharpness and contrast degree of the vision are poor. Deep learning has been one of the major breakthroughs in the field of artificial intelligence in the past decade, which is widely used in the fields of speech recognition, natural speech processing, computer vision, image analysis, and so on. Deep learning has more advantages than traditional image processing methods in terms of precision and speed of target detection, and the idea is also applied in various fields. Therefore, it is an effective and feasible way to assist the underwater robot to complete the fishing process by using the deep learning technology based on neural network to recognize the target.

The traditional image recognition method is divided into three steps. First, the proposed region is divided into the original image (Region Proposal); then, the feature is extracted from the region. Finally, the trained model is used to identify the region. The image recognition usually uses a frame of a certain size as a sliding window to traverse the whole image, which is called “anchor.” By setting different length and width ratios and sizes, the target is determined by the exhaustive method. In this way, it is not difficult to find the object, but it is difficult to realize the purpose of real-time detection. For feature extraction, it is difficult to extract the features that can adapt to the changes of shape, brightness, and background.

There are a lot of tiny particles in sea water, especially in the coastal sea area and the marine aquaculture sea area. The water quality is generally turbid, and light scattering in the water is more serious. The contrast ratio of the water vision is low, and the color attenuation and the noise are very serious. In order to deal with these problems, image processing methods have been commonly used to enhance the image quality.

Trucco and Olmos-Antillon proposed a simplified self-tuning recovery algorithm based on the Jaffe McGlamery model of radiative transfer equation [1]. Using this method, the processing and classification on a $320 * 240$ image require 3.8s. Yamashita et al. adopted a color registration method [2], which considered the attenuation of light in the water, and the degraded color information was restored visually. Dudek et al. (2005) used a color correction filter to detect the submerged reef [3], which is installed on an amphibious robot, and they also designed a visual servo system to detect human guidance through a visual sensor. Then, the detection of underwater targets using an integrated tracker, which is implemented by weighted Haar similarity detection and the recognition ability compared with the color spot tracking method, is higher. Mane and Pujari developed a video detection system based on the Gauss mixed model using a differential method to remove the background information, which can be applied to locate the moving target in the video, and the recognition rate can reach 80% [4]. Barat and Phlypo (2010) developed an automatic active contour detection system for underwater fixed object segmentation [5], which is mainly based on the difference of the color and contrast between the object and background to detect the target. Compared with the active contour segmentation method and the maximum active contour method based on the color space, the adaptability and segmentation accuracy of this method are higher. It is suitable for the online segmentation under the natural environment, but the segmentation effect is poor when the color of the target is similar to the background. Rizzini et al. (2015) proposed a multifeature target detection algorithm [6], which is applied to automatically detect the plastic pipes placed in natural water. Image recognition is the identification on the segmented regions of images. In order to realize multitarget recognition, Kim et al. (2014) presented a method of segmentation and recognition of underwater roadmap based on color feature; the weight of the feature is determined by the correlation coefficient method, in which the influence of size is removed [7]. Walther et al. (2004) developed a system to detect and label the underwater suspected targets [8], which can help oceanologists to study marine organisms. With the help of this system, the background is removed and the targets are identified and tracked by selective and significant algorithm, which can achieve underwater multitarget tracking. Then, Edgington et al. (2006) optimized the system, Bayes classifier using mixed Gauss model is applied to classify the captured targets, and the information is automatically processed and sent to realize the intelligent operation [9].

Toshihiro et al. (2011) detected the underwater tube insects in the sea area of Maki [10], and two-dimensional Fourier transform was carried out to separate the tube insects from the background. The area, volume, and height of the area were obtained by extracting the characteristic morphology of the tube insects. Morris et al. (2015) used the Autosub robot to detect the seabed in the Porcupine deep sea in the Atlantic [11], and an automatic image processing system is developed to detect the giant organisms and their distribution in the region. According to the visual characteristics of fish, Salman et al. (2016) applied the method of convolution neural network to simulate the perception of different fish shapes by means of deep learning [12]. The accuracy rate is up to 90%. In order to study the behavior of the fish group and facilitate the control of the fish culture, Boussarie et al. (2016) developed an online real-time video processing system to complete the counting and tracking for the shoal of fish [13]. Prasanna et al. (2015) applied the image segmentation algorithm to detect the Kannappan [14], through image recognition to monitor the growth of scallop. Enomoto et al. (2014) used the mean shift algorithm to remove the background and shaded area of the scallop on the seabed, and the dynamic closed value method is applied to divide the contour, and the scallop is detected by the shape feature [15].

The development of neural networks is derived from the human vision and brain research on human and animal bodies. In the 60s, through the visual study of cats, Hubel found the presence of Receptive Field in the visual system of animals, which means that visual cells can feel the stimulation of light to produce an exciting area. In 1943, the American neurophysiologist McCulloch and mathematical logician W. Pitts proposed a mathematical model based on the basic structure of neurons, which is known as the MP model. Based on this basic unit, a deeper level network model can be built to complete the machine learning process.

With the rapid development of computer hardware, the implementation of neural networks is no longer so difficult. In the 80s, the backpropagation algorithm was proposed by the American psychologist Rumelhar and the British cognitive psychologist Hinton to solve the XOR problem; from then on, the neural network is applied commonly. Many mathematical models of neural networks have been developed by many scholars. On the basis of previous research, in 2006, Professor Geoffery Hinton of Toronto University first proposed "deep belief network," in which a pretraining model is used to find the optimal solution of the weights of the neural network through iteration, and then, through the fine-tunning, the entire network is modified to save the time of training the neural network. This multilayer neural network learning framework is named "deep learning." Since then, the concept of deep neural network has been widely adopted, which has a great impact on speech recognition and image recognition.

For image recognition, Krizhevsky (2012) used the convolution neural network to deal with the classification problem, winning the championship in the world's most authoritative computer vision competition ILSVRC (ImageNet Large Scale Visual Recognition Challenge) [16]. This method reduced the error rate of top 5 to 15.3%. From then on, the convolution network has gained the greatest approval in image recognition. In 2014, Facebook expert Girshick developed the regional convolution neural network (Region-Convolutional Neural Network (R-CNN)) with the combination of the region proposal network and the convolution neural network (Convolutional Neural Network (CNN)) [17]. The detection results on the PASCAL VOC2007 dataset reached 66% mAP (mean Average Precision). On the basis of R-CNN, He et al. (2015) proposed the SPP-Net model [18], which greatly improved the detection efficiency. Girshick et al. added the loss function in the SPP-NET and established the Fast-RCNN model [19]. In the training process, the multilayer perceptron (Multilayer Perceptron (MLP)) is used instead of SVM (Support Vector Machine) to realize the classification, and the training steps and speed have been significantly improved. On the basis of Fast R-CNN, He and Girshick developed the Faster-RCNN program using RPN (Region Proposal Network) instead of selective search of Fast RCNN to define and modify the bounding box to solve the end-to-end target detection problem [20].

The deep learning method based on convolution neural network is adopted in many fields. At the present stage, RCNN, Faster RCNN, and some modified versions are widely used in various engineering research. Overall, in the field of image recognition, the development of the algorithm based on convolution neural network is fast. In order to improve the real-time efficiency of video processing and image recognition, a variety of algorithms have been put forward to enable the computer to deal with the information of the image in a very short time. The advanced technology in this field has already been developed. But in engineering and other applications, scholars in various fields have to realize further improvement. In the aspect of underwater target recognition, there are still a lot of problems that need to be solved using image processing at the present stage. The traditional methods are using contour segmentation and feature extraction to locate the target. Although these methods have already been applied in many fields, the speed and accuracy are far behind the advanced technology in image recognition. This research intends to propose a new underwater CNN recognition technology to optimize the detection program, which is used to facilitate the underwater ROV to detect and classify marine organism.

2. Methodology Applied in Marine Organism Detection

Faster RCNN is assumed to be a combination system of RPN and Fast RCNN; in Fast RCNN, the selective search is too time-consuming, so RPN is added to improve the region detection calculation. RPN is used to fulfill two improvements, through softmax to classify the anchors to obtain the foreground and background information; the other improvement is to obtain the accurate proposals through the bounding box regression offset calculation of anchors.

2.1. Recommended Loss Function. There are two fully connected output layers in the same level in Faster RCNN, the score calculation, and the BBox prediction, so this is a multitask structure, as shown in Figure 1. The score calculation layer is for classification, the output is a $k+1$ dimension array p, which includes the background probabilities of all the classes on every RoI, and p is obtained by softmax in fully connected layer:

$$p = (p_0, p_1, p_2, \ldots, p_k). \tag{1}$$

In the BBox prediction layer, the region proposals are modified to output the box regression offset, and a $4 * k$ dimension array t is calculated to deprecate the offset parameters:

$$t^k = \left(t_x^k, t_y^k, t_w^k, t_h^k\right), \tag{2}$$

where k is the class index, (t_x^k, t_y^k) is the object proposal of scale invariance offset, and (t_w^k, t_h^k) is the height and width of logarithmic space in object proposal.

The classification u is assessed by the loss calculation:

$$L_{\text{cls}}(p, u) = -\log p_u. \tag{3}$$

The bounding box loss calculation is the location assessment, which equals the difference between the real offset value and the prediction value:

$$L_{\text{loc}}(t^u, v) = \sum_{i=1}^{4} \text{smooth}_{L_1}(t_i^u - v_i), \tag{4}$$

where smooth is the loss function:

$$\text{smooth}_{L_1}(x) = \begin{cases} \dfrac{x^2}{2}, & |x < 1|, \\ |x|, & -0.5. \end{cases} \tag{5}$$

The form of the loss function has a good robustness, and the total loss function is

$$L(p, u, t^u, v) = \begin{cases} L_{\text{cls}}(p, u) + \lambda L_{\text{loc}}(t^u, v) \; u \geq 1, \\ L_{\text{cls}}(p, u), \end{cases} \tag{6}$$

where λ is the weight coefficient, $u = 0$ is the background label, the exponential function indicates the background area, and the negative sample does not participate in the regression loss.

The λ controls the balance of classification loss and regression loss. In the Fast R-CNN method, all the experimental lambda = 1, but this coefficient could be modified according to the different applications.

Faster RCNN is a multitask method, and the loss function is defined as

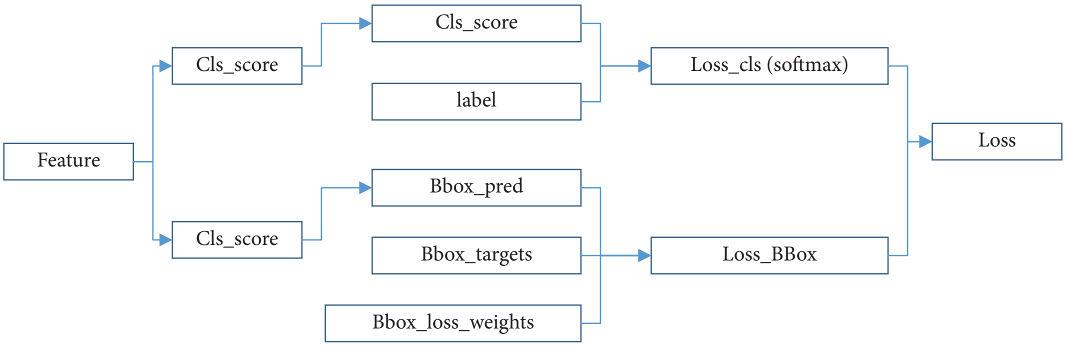


Figure 1: Loss calculation structure.

$$L(p_i, u_i) = \frac{1}{N_{\text{cls}}} \sum_i L_{\text{cls}}(p_i, p_i^*) + \lambda \frac{1}{N_{\text{reg}}} \sum_i p_i^* L_{\text{reg}}(t_i + t_i^*), \tag{7}$$

where p_i is the prediction possibility of the anchors, t_i is the vector of the bounding box parameters, t_i^* is the ground truth of the positive anchor, and GT label p_i^* is defined as follows:

$$p_i^* = \begin{cases} 0, & \text{negative}, \\ 1, & \text{positive}. \end{cases} \tag{8}$$

L_{cls} is the logarithmic loss function of the detection target:

$$L_{\text{cls}}(p_i, p_i^*) = -\log[p_i^* p_i + (1 - p_i^*)(1 - p_i)]. \tag{9}$$

L_{reg} is the regression loss:

$$L_{\text{reg}}(t_i, t_i^*) = \text{smooth}_{L_1}(t_i - t_i^*). \tag{10}$$

When the $p_i^* = 1$, there is only regression loss of the foreground information.

2.2. Bounding Box Regression. The ground truth and the region proposal are shown in Figure 2; the black box is the ground truth, the red box is the region proposal, and the red one is the detected sea cucumber, but the red box is not accurate. IoU (Intersection over Union) is less than 0.5, so this is a wrong detection, through modification on the region proposal; the detection is closer to the ground truth, and the bounding box regression is the method to realize the modification.

A 4-dimension vector (x, y, w, h) is used to deprecate the bounding box information, which is the central point location and the width and height; in Figure 3, the red frame is the original proposal, the green one is the target ground truth, and the regression aims to obtain the blue one, which is based on original proposal and closer to the labeled grounding truth (GT) after offset.

The offset is a mapping function:

$$f(p_x, p_y, p_w, p_h) = (G'_x, G'_y, G'_h, G'_w) \approx (G_x, G_y, G_w, G_h). \tag{11}$$

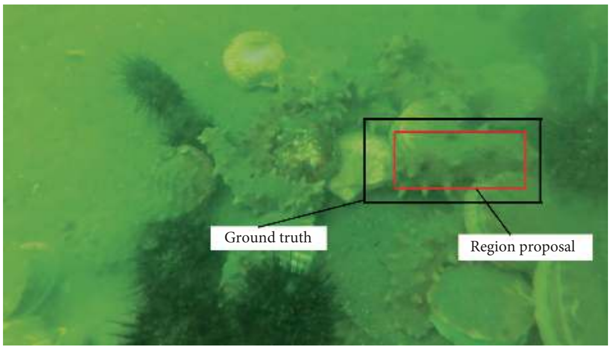


Figure 2: Bounding box and ground truth.

The offset and the scaling transformation functions are as follows:

$$\begin{aligned} G'_x &= p_w \cdot d_x(p) + p_x, \\ G'_y &= p_h \cdot d_x(p) + p_y, \\ G'_w &= p_w \cdot \exp(d_w(p)), \\ G'_h &= p_h \cdot \exp(d_h(p)). \end{aligned} \tag{12}$$

In equations above, d_x, d_y, d_h, and d_w can be calculated from deep learning, when the anchor is close to the GT, the offset is a linear transformation, and the offsets (t_x, t_y) and scales (t_w, t_h) are commutated as follows:

$$\begin{aligned} t_x &= \frac{(x - x_a)}{w_a}, \\ t_y &= \frac{(y - y_a)}{h_a}, \\ t_w &= \log\left(\frac{w}{w_a}\right), \\ t_h &= \log\left(\frac{h}{h_a}\right). \end{aligned} \tag{13}$$

The feature map Φ, obtained from convolution, and the (t_x, t_y, t_w, t_h) are the input data; the output is d_x, d_y, d_h, and d_w, the weight parameter is w, and the objective function is defined as

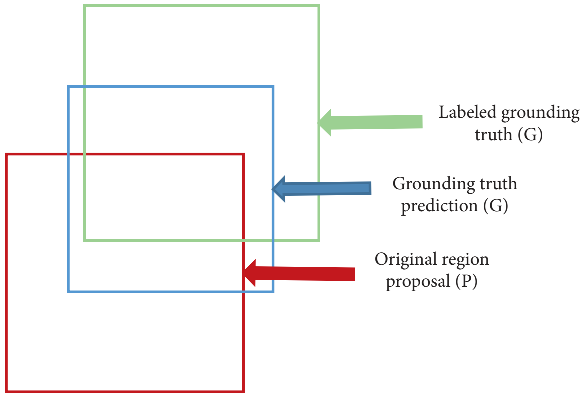

Labeled grounding truth (G)
Grounding truth prediction (G)
Original region proposal (P)

Figure 3: The regression process.

$$d_*(p) = w_*^T \cdot \Phi(p), \tag{14}$$

where $\Phi(p)$ is the eigenvector of feature map, w are the parameters obtained from deep learning, and $d(p)$ is the prediction, so the loss function can be defined as follows:

$$\text{loss} = \sum_i^N \left(t_*^i - \widehat{w}_*^T \cdot \Phi\left(p^i\right)\right)^2, \tag{15}$$

The objective function is

$$w_* = \arg\min \sum_i^N \left(t_*^i - \widehat{w}_*^T \cdot \Phi\left(p^i\right)\right)^2 + \lambda \|\widehat{w}\|^2. \tag{16}$$

3. Improvement and Modifications for Application

For marine organism detection, the objects are in small size, and because of the high complexity of the seabed environment, a lot of the objects are overshadowed by coral reef and some other things, and the detection and localization are not accurate enough to achieve the automatic fishing requirement. In order to modify the program to improve the detection, the Hypernet method proposed by Tao et al. (2016) is used to solve these problems [21], which is primarily based on an elaborately designed hyperfeature to aggregate hierarchical feature maps first and then compress them into a uniform space. The feature extraction method proposed in this paper is inspired from the hypernet method, and then the feature map is extracted by the RPN network.

3.1. Feature Extraction. The commonly used methods, RCNN, Fast RCNN, and Faster RCNN, are not applicable to detect objects in small size. The reason is that the original information loses a lot in the feature extraction process. So, we can deal with these problems by two ways: the first is to magnify the feature maps, and the other is to reduce the convolution layers. A max pooling layer is added in the shallow layers to achieve down-sampling; in the deep layer, a deconvolution layer is added to perform up-sampling; through these procedures, more semantic information is retained and the hyperfeatures are obtained by local response normalization (LRN) calculation.

In order to realize real-time detection, the net operation time is improved, but the region proposal is still very time consuming to be obtained. In order to solve this problem, RPN is used to define the possible location of the object in the image by extracting the texture, edge, color, and other features to ensure fewer windows applied with a higher IoU (Intersection over Union). The Region Proposal can be obtained using window sliding through the last convolution layer straightly. The structure is based on the neutral network, and the bounding box and softmax regression are included in output model.

For RPN, the sliding windows generate 9 anchors in different fixed size, but it is not reasonable to use the same anchors for different input proposals, and based on the hyperfeature maps, the proposals are generated in different random sizes. Considering the overlap of the region proposals, the Greedy NMS (Nonmaximum Suppression) method is applied to reduce the duplication. After the full-convolution layer, a $3 * 3$ layer is added to make the classification more accurate. The feature map dimension is reduced to speed up the computation, and the extraction neutral network is shown in Figure 4.

3.2. Object Detection Modification. The detection procedure is basically in conformance with Faster RCNN, but before the fully connected layer, a convolution layer is added to reduce the dimension of the features; at the same time, in order to improve the efficiency, the dropout ratio is changed from 0.5 to 0.25.

For the extraction, there are two parts of output. In this research, there are 3 classes; every region generates $3 + 1$ anchors, and then through the NMS method to delete the overlaps, the network structure is shown in Figure 5.

3.3. Speeding up Method. To speed up the detection time, before the RoI pooling layer, a $3 * 3 * 4$ convolution layer is added; thus, the feature maps are reduced a lot, and the sliding window classifier is simple, as shown in Figure 6.

The Region Proposal process is very time-consuming, and a large number of bounding boxes should be calculated, so a convolution layer is added before the pooling layer to realize the lightweight of the program.

4. Experimental Results

The experiment dataset is provided by the "Underwater Robot Picking Contest," organized by the National Natural Science Foundation of China. And some of the images are obtained from the camera installed on a ROV, which is recorded by an underwater robot designed by Harbin Marine Equipment Co. Ltd. The robot is a remote operated vehicle (ROV), which is designed and produced to achieve the underwater marine organisms fishing. The robot is about 1 m long and 0.8 meters wide and weighs 90 kg. The method of collecting marine products is of adsorption type, and the design and real robot are shown in Figure 7.

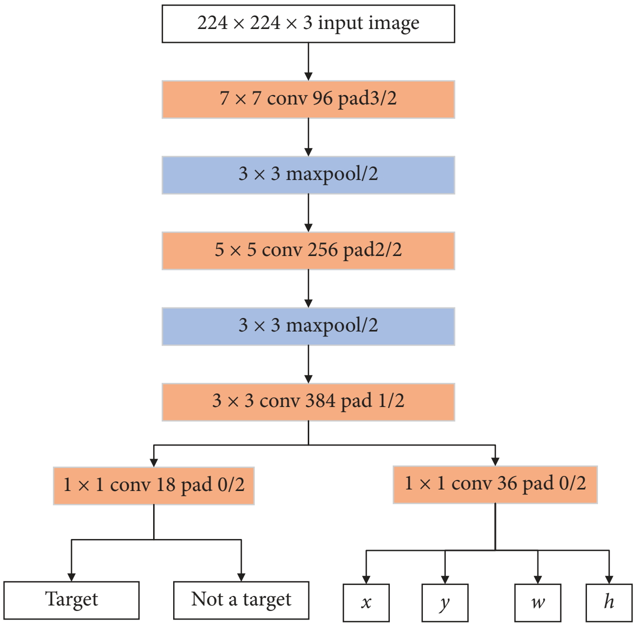


FIGURE 4: Feature extraction method.

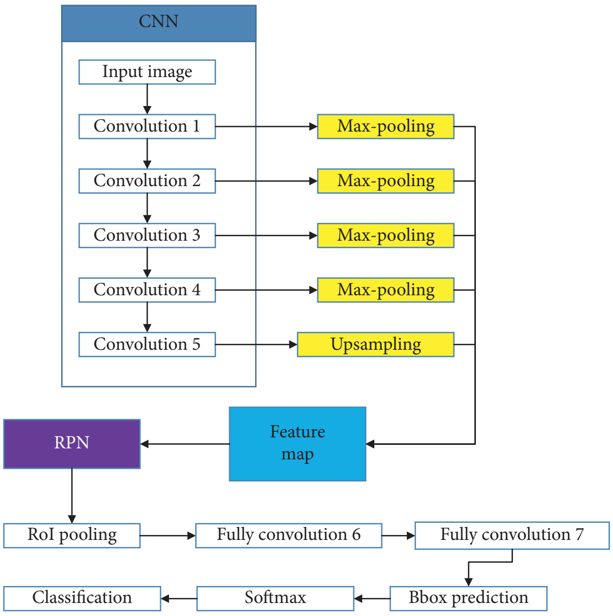


FIGURE 5: The underwater object detection CNN structure.

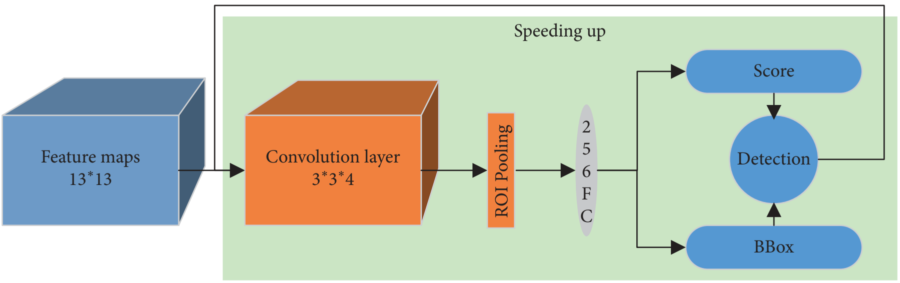


FIGURE 6: Speeding up method.

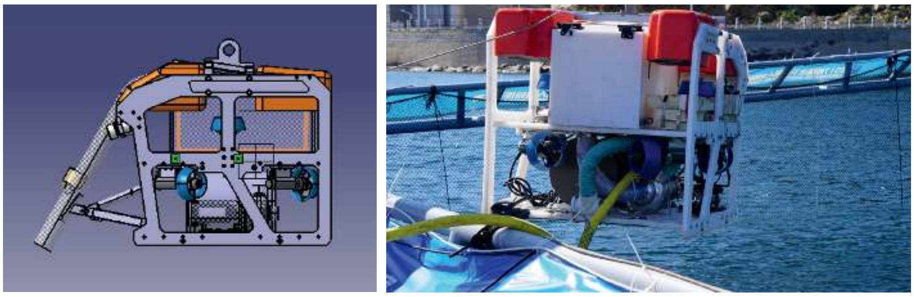
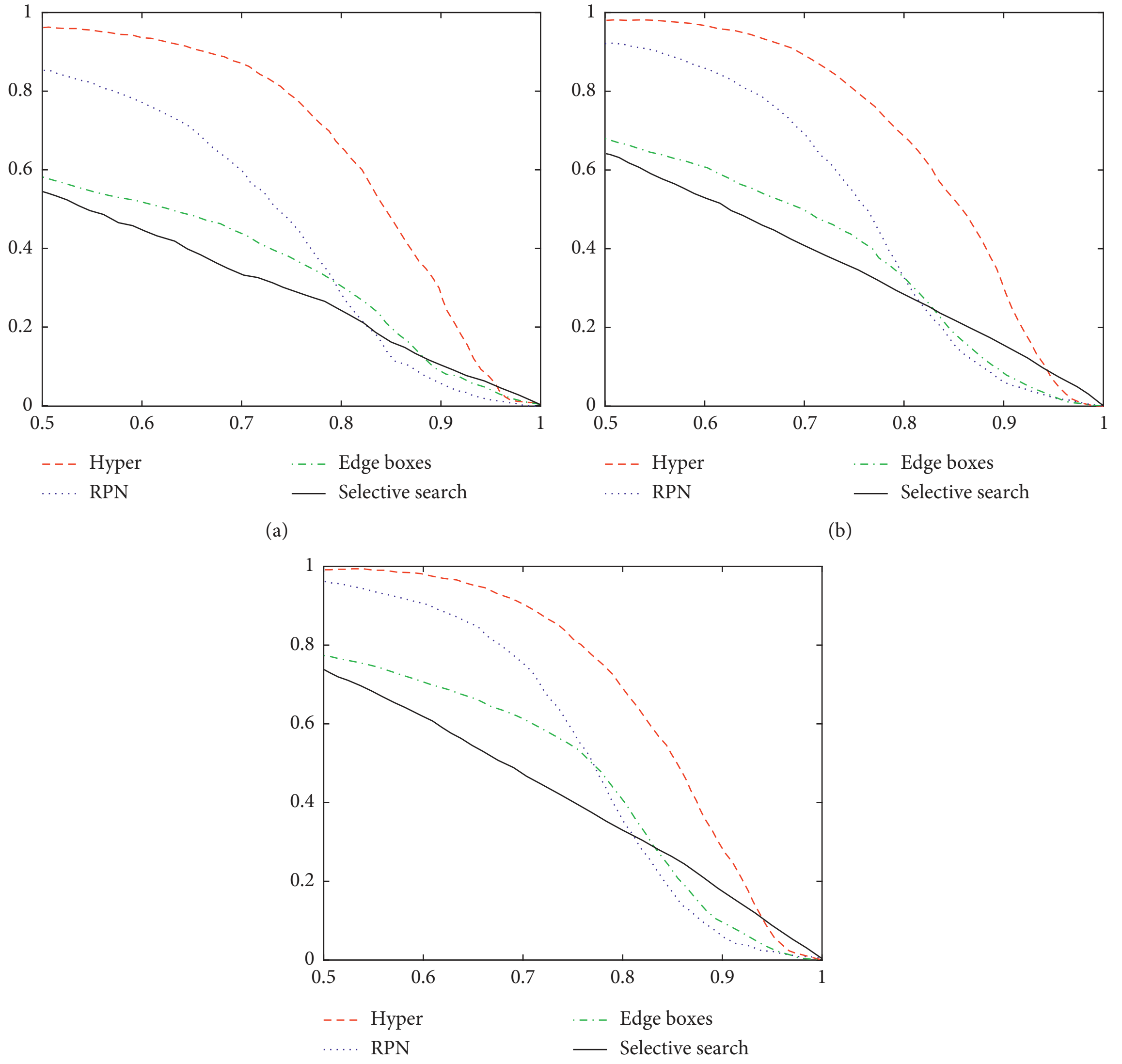


Figure 7: Underwater robot for marine organisms fishing.



(a)

(b)

(c)

Figure 8: Recall and IoU on the underwater dataset. (a) 50 region proposals. (b) 100 region proposals. (c) 200 region proposals.

4.1. Region Proposal Generation. The recall and localization accuracy are evaluated on the underwater dataset, which consists of 18,978 images with bounding box annotation for the object detection from 3 categories. The modified method is compared with the RPN, Edge boxes, and Selective Search methods; IoU is defined as the intersection divided by the union of the ground truth and bounding boxes. For a fixed number of proposals, the recall is evaluated, as shown in Figure 8.

It is clear that when the region numbers are reduced, the recall can still reach more than 95%. The recall values

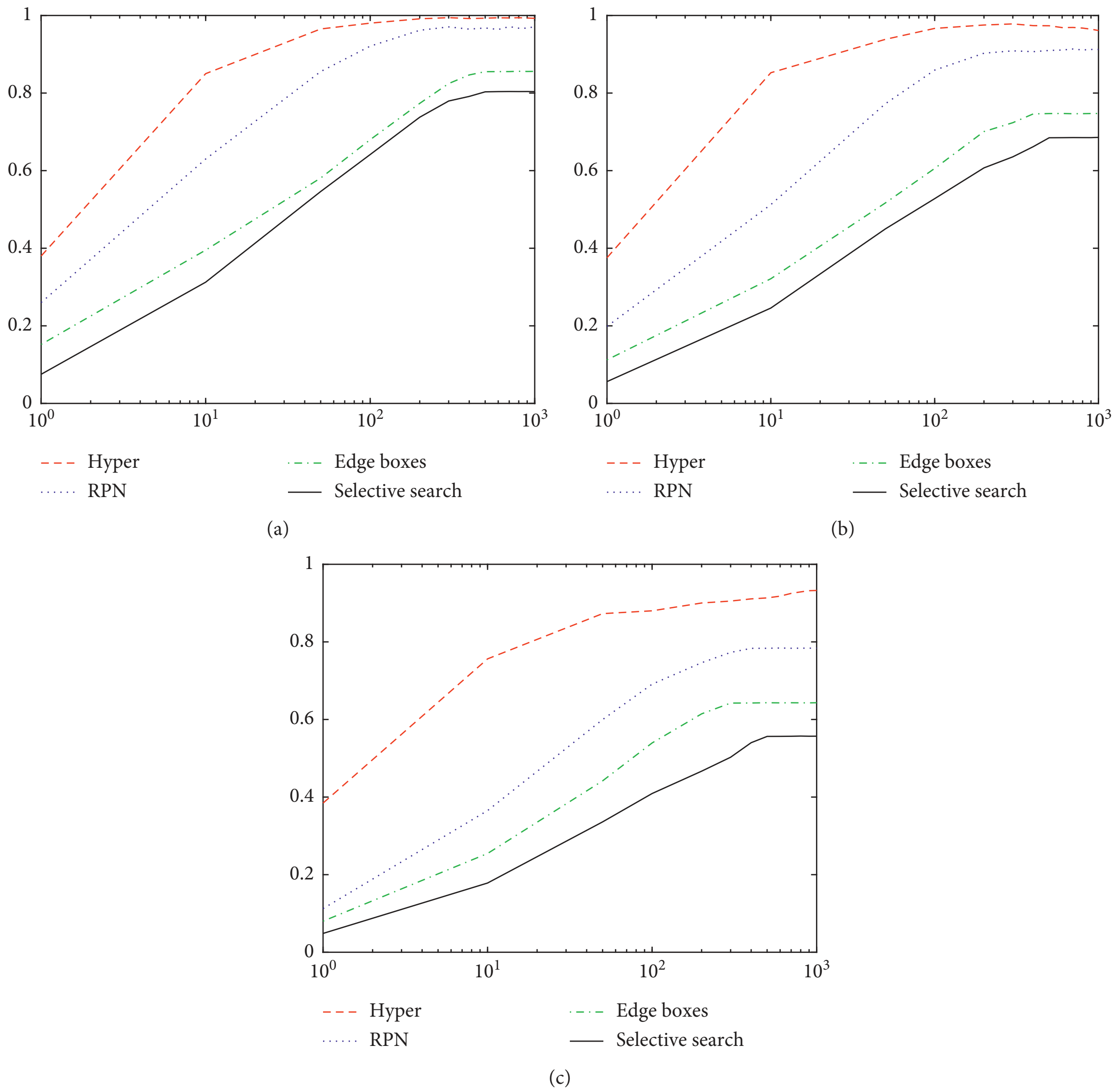


Figure 9: Recall and number of proposals on the under water dataset. (a) IoU = 0.5. (b) IoU = 0.6. (c) IoU = 0.7.

obtained by the Hypernet method with 200 proposals and IoU = 0.5 outperformed RPN by 4%, Edge boxes method by 12%, and Selective Search method by 16%.

The relationship between recall and number of proposals is shown in Figure 9. In the real application, IoU = 0.5 is not enough to achieve the accurate detection and the threshold of IoU of 0.7 is commonly used to fit the ground truth object. The results are shown in Table 1 in detail.

4.2. Underwater Dataset Results. The dataset is obtained from the video provided by the Underwater Robot Picking Contest, which contains 3 categories, and the images are labeled as the Pascal VOC form. The performance is measured by mean average precision (mAP) on the test set, which contains 8800 images. A pretrained VGG16 model is used to define the initial model coefficients. The comparison is shown in Tables 2 and 3.

When the IoU is set 0.5, the Fast RCNN with Selective Search gets an mAP of 85%, Faster RCNN gets an mAP of 88.6%, and Hyper net can achieve an mAP of 91.2%. YOLO [22] and RetinaNet [23] are the typical single-step detection methods; they are different from the proposed method and the RCNN series methods; they always give a faster detection, but the accuracies are relatively lower. The mAPs obtained on this dataset is higher; this is because the dataset is prepared from the video, which is filmed in the same sea area, and the organisms are the same categories; the training images change relatively little, so the precision is high. But for the application in ROV, which is used in the same sea area, it is in good performance.

Table 1: The recall results of different number of proposals.

Number of proposals	IoU = 0.5				IoU = 0.6				IoU = 0.7			
	Hyper	RPN	Edge boxes	Selective search	Hyper	RPN	Edge boxes	Selective search	Hyper	RPN	Edge boxes	Selective search
1	0.380	0.260	0.152	0.075	0.376	0.199	0.112	0.056	0.385	0.113	0.080	0.049
10	0.850	0.630	0.395	0.313	0.852	0.513	0.321	0.246	0.756	0.365	0.255	0.178
50	0.965	0.855	0.581	0.546	0.938	0.772	0.517	0.450	0.873	0.600	0.442	0.336
100	0.980	0.921	0.680	0.642	0.967	0.859	0.606	0.528	0.880	0.691	0.539	0.409
200	0.991	0.962	0.773	0.738	0.975	0.903	0.701	0.607	0.900	0.746	0.615	0.467
300	0.994	0.971	0.825	0.780	0.978	0.909	0.724	0.636	0.905	0.773	0.642	0.503
400	0.992	0.965	0.847	0.791	0.974	0.907	0.746	0.662	0.911	0.783	0.642	0.540
500	0.993	0.968	0.855	0.803	0.973	0.910	0.747	0.685	0.913	0.784	0.643	0.556
600	0.994	0.964	0.910	0.804	0.969	0.911	0.747	0.685	0.918	0.784	0.643	0.556
700	0.994	0.971	0.910	0.804	0.969	0.914	0.747	0.685	0.925	0.784	0.643	0.557
800	0.994	0.967	0.911	0.804	0.967	0.911	0.747	0.685	0.929	0.784	0.643	0.557
900	0.993	0.967	0.910	0.804	0.965	0.912	0.747	0.685	0.932	0.784	0.643	0.557
1000	0.993	0.971	0.910	0.804	0.961	0.912	0.747	0.686	0.933	0.784	0.643	0.557

Table 2: Results on underwater datasets with IoU = 0.5.

Method	mAP	Sea cucumber	Sea urchin	Scallop
Fast RCNN	85	89.2	86.4	79.4
Faster RCNN	88.6	90.3	89.7	85.8
RetinaNET	67.1	69.3	66.9	65.1
YOLO	71.3	72.6	70.3	71.0
Proposed method	91.2	94.5	92.6	86.5

Table 3: Results on underwater datasets with IoU = 0.7.

Method	mAP	Sea cucumber	Sea urchin	Scallop
Fast RCNN	51.2	55.6	52.3	45.7
Faster RCNN	59.6	62.8	61.4	54.6
RetinaNET	36.1	38.2	36.7	33.4
YOLO	39.4	41.3	40.3	36.6
Proposed method	70.2	75.6	73.8	61.2

Table 4: Detection time for one image by different methods.

Approach	Fast RCNN	Faster RCNN	RetinaNET	YOLO	Proposed method	Speeding up
Time cost (ms)	96	85	41	34	65	58

4.3. Detection Results. An Nvidia GTX 1080ti is used for detection; the time consumed is about 58 ms for one image detection of the speeding up method. And the Fast RCNN used 96 ms and the Faster RCNN used 85 ms, and the results are shown in Table 4.

Through the comparison, the single-step detection methods can give much more faster speed, although the accuracy is lower, but the YOLO and RetinaNET can be used in real-time detection as application requirement for other circumstance. But for our underwater research, we paid more attention on the features of the objects to improve the detection ability. For the real life engineering application, we are going to do more research on the size prediction and underwater picking technology, so the more accurate detection is more important for our future research.

The following images are used to testify the method proposed in this paper, the images shown in Figure 10 are filmed by the ROV in remote location, the objects are tiny, and some of them are overlapped by the other objects; even when the images are vague, the objects can be detected and classified accurately. As shown in Figure 11, the objects are located close to the camera, some of the objects are covered by sands, and the program can almost detect all of the objects.

In Figure 12, the background is different, and the images are filmed from a far distance when the sunshine is weak, and the objects can be detected too. When the objects are big in the images, which are filmed from a close distance, and the detection effect is in good performance too, as shown in Figure 13.

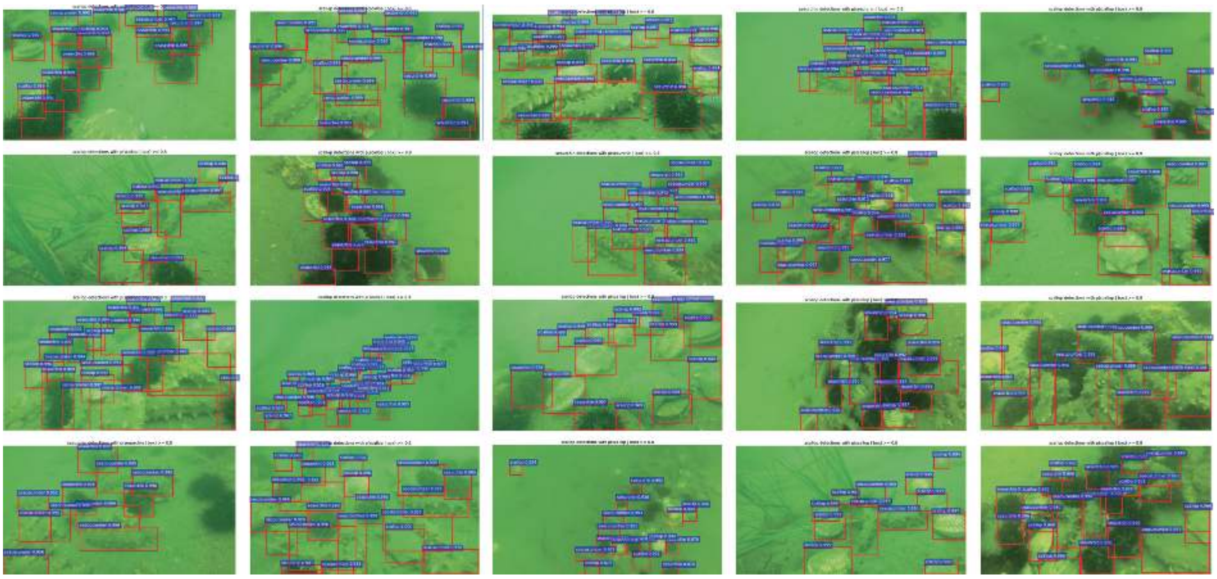


Figure 10: Detection results of the images filmed from a far distance.

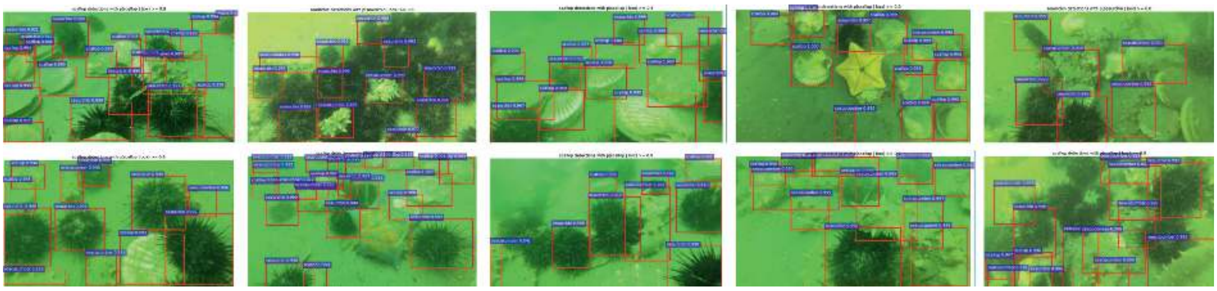


Figure 11: Detection results of the images filmed from a close distance.

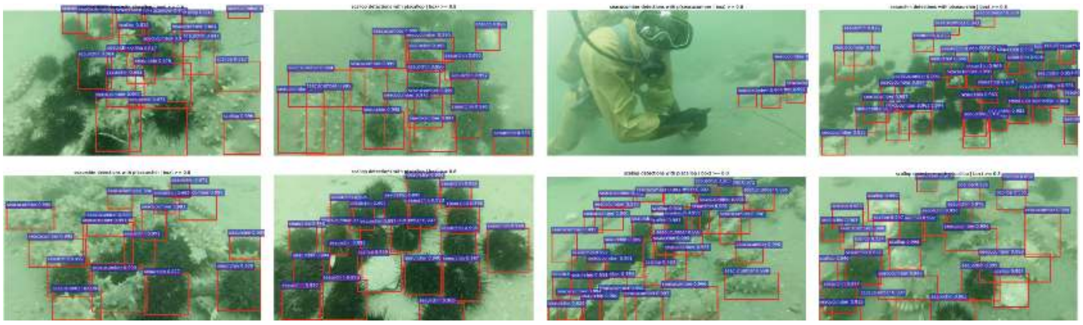


Figure 12: Detection results of the images filmed from a far distance when it is cloudy.

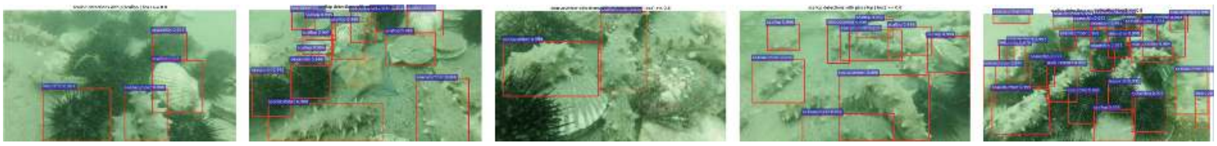


Figure 13: Detection results of the images filmed from a close distance when it is cloudy.

5. Conclusion

A deep CNN network is proposed to realize the detection and classification of marine organisms, which is based on faster RCNN and the modified method of hyper net. The modified framework is used to achieve the marine organism detection on the underwater dataset, which is obtained using the ROV and from the Underwater Robot Picking Contest. The analysis and the detection results show that the method proposed in this paper is feasible to be applied in the underwater vision detection. The mAP is more than 90% when the IoU is set to be equal to 0.7. The detection time is 58 ms running on the GPU of NVIDIA GTX 1080ti, which is enough to be used on a camera installed on the ROV to achieve the real-time detection.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that there are no conflicts of interest.

Acknowledgments

The authors would like to express their gratitude for support from the National Key R&D Program of China (Grant no. 2018YFC0309402).

References

[1] E. Trucco and A. T. Olmos-Antillon, "Self-tuning underwater image restoration," *IEEE Journal of Oceanic Engineering*, vol. 31, no. 2, pp. 511–519, 2006.

[2] A. Yamashita, M. Fujii, and T. Kaneko, "Color registration of underwater images for underwater sensing with consideration of light attenuation," in *IEEE International Conference on Robotics & Automation*, vol. 14, IEEE, Piscataway, NJ, USA, April 2007.

[3] G. Dudek, M. Jenkin, C. Prahacs et al., "A visually guided swimming robot," in *IEEE/RSJ International Conference on Intelligent Robots & Systems*, IEEE, Edmonton, Canada, August 2005.

[4] K. T. Mane and V. G. A. Pujari, "Novel approach for species detection from oceanographic video," in *Fourth International Conference on Advanced Computing & Communication Technologies*, IEEE, Piscataway, NJ, USA, 2014.

[5] C. Barat and R. Phlypo, "A fully automated method to detect and segment a manufactured object in an underwater color image," *EURASIP Journal on Advances in Signal Processing*, vol. 2010, no. 1, pp. 1–11, 2010.

[6] D. L. Rizzini, F. Kallasi, F. Oleari, and S. Caselli, "Investigation of vision-based underwater object detection with multiple datasets," *International Journal of Advanced Robotic Systems*, vol. 12, no. 6, pp. 1–13, 2015.

[7] D. Kim, D. Lee, H. Myung, and H.-T. Choi, "Artificial landmark-based underwater localization for AUVs using weighted template matching," *Intelligent Service Robotics*, vol. 7, no. 3, pp. 175–184, 2014.

[8] D. Walther, D. R. Edgington, and C. Koch, "Detection and tracking of objects in underwater video," in *IEEE Computer Society Conference on Computer Vision & Pattern Recognition*, IEEE, Washington, DC, USA, July 2004.

[9] D. R. Edgington, I. Kerkez, D. Cline, D. Davis, and J. Mariette, "Tracking and classifying animals in underwater video," in *Proceedings of the Oceans 2006*, IEEE, Boston, MA, USA, September 2006.

[10] T. Maki, A. Kume, and T. Ura, "Volumetric mapping of tubeworm colonies in Kagoshima Bay through autonomous robotic surveys," *Deep Sea Research Part I: Oceanographic Research Papers*, vol. 58, no. 7, pp. 757–767, 2011.

[11] K. J. Morris, B. J. Bett, J. M. Durden et al., "A new method for ecological surveying of the abyss using autonomous underwater vehicle photography," *Limnology and Oceanography: Methods*, vol. 12, no. 11, pp. 795–809, 2014.

[12] A. Salman, A. Jalal, F. Shafait et al., "Fish species classification in unconstrained underwater environments based on deep learning: fish classification based on deep learning," *Limnology and Oceanography: Methods*, vol. 14, no. 9, pp. 570–585, 2016.

[13] G. Boussarie, N. Teichert, R. Lagarde, and D. Ponton, "BichiCAM, an Underwater Automated Video Tracking System for the Study of Migratory Dynamics of Benthic Diadromous Species in Streams," *River Research and Applications*, vol. 32, no. 6, pp. 1392–1401, 2016.

[14] P. Kannappan, J. H. Walker, A. Trembanis, H. G. Tanner, and O. Methods, "Identifying sea scallops from benthic camera images," *Limnology and Oceanography: Methods*, vol. 12, no. 10, pp. 680–693, 2014.

[15] K. Enomoto, M. Toda, and Y. Kuwahara, "Extraction method of scallop area from sand seabed images," *IEICE Transactions on Information and Systems*, vol. E97.D, no. 1, pp. 130–138, 2014.

[16] A. Krizhevsky and I. Sutskever, "Hinton GE ImageNet classification with deep convolutional neural networks," in *International Conference on Neural Information Processing Systems*, Lake Tahoe, Nevada, December 2012.

[17] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "rich feature hierarchies for accurate object detection and semantic segmentation," in *IEEE Conference on Computer Vision & Pattern Recognition*, IEEE, Columbus, OH, USA, June 2014.

[18] K. He, X. Zhang, S. Ren, J. Sun, and M. Intelligence, "Spatial pyramid pooling in deep convolutional networks for visual recognition," in *Computer Vision—ECCV 2014*, vol. 37, pp. 1904–1916, Springer Cham, Manhattan, NY, USA, 2014.

[19] R. Girshick, "Fast R-CNN," *IEEE International Conference on Computer Vision*, Las Condes, Chile, December 2015.

[20] S. Ren, K. He, R. Girshick, J. Sun, and M. Intelligence, "Faster R-CNN: towards real-time object detection with region proposal networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 6, pp. 1137–1149, 2017.

[21] K. Tao, A. Yao, Y. Chen, and F. Sun, "HyperNet: Towards accurate region proposal generation and joint object detection," Computer Vision & Pattern Recognition, 2016, https://arxiv.org/abs/1604.00600.

[22] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You Only Look once: Unified, Real-Time Object Detection," 2015, https://arxiv.org/abs/1506.02640.

[23] T. Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollar, "focal loss for dense object detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 42, no. 2, pp. 318–327, 2020.