Dual Refinement Underwater Object Detection Network

Baojie Fan[1(✉)], Wei Chen[1], Yang Cong[2], and Jiandong Tian[2]

[1] College of Automation and College of Artificial Intelligence, Nanjing University of Posts and Telecommunications, Nanjing 210023, China
jobfbj@gmail.com, no1chenwei@gmail.com
[2] Shenyang Institute of Automation (SIA), Chinese Academy of Sciences, Shenyang 110016, China
{congyang,tianjd}@sia.cn

Abstract. Due to the complex underwater environment, underwater imaging often encounters some problems such as blur, scale variation, color shift, and texture distortion. Generic detection algorithms can not work well when we use them directly in the underwater scene. To address these problems, we propose an underwater detection framework with feature enhancement and anchor refinement. It has a composite connection backbone to boost the feature representation and introduces a receptive field augmentation module to exploit multi-scale contextual features. The developed underwater object detection framework also provides a prediction refinement scheme according to six prediction layers, it can refine multi-scale features to better align with anchors by learning from offsets, which solve the problem of sample imbalance to a certain extent. We also construct a new underwater detection dataset, denoted as UWD, which has more than 10,000 train-val and test underwater images. The extensive experiments on PASCAL VOC and UWD demonstrate the favorable performance of the proposed underwater detection framework against the states-of-the-arts methods in terms of accuracy and robustness. Source code and models are available at: https://github.com/Peterchen111/FERNet.

Keywords: Underwater object detection · Feature enhancement · Anchor refinement · Underwater dataset

1 Introduction

At present, underwater robots are used in many fields, such as underwater target capture, underwater investigation, and underwater search. As the key technology of underwater robots, underwater object detection still faces severe challenges (e.g., blur, texture distortion, imbalanced illumination, etc.). The above issues restrict the development of underwater robot object detection.

B. Fan and W. Chen—The first two authors contribute equally to this work.

A. Vedaldi et al. (Eds.): ECCV 2020, LNCS 12365, pp. 275–291, 2020.
https://doi.org/10.1007/978-3-030-58565-5_17

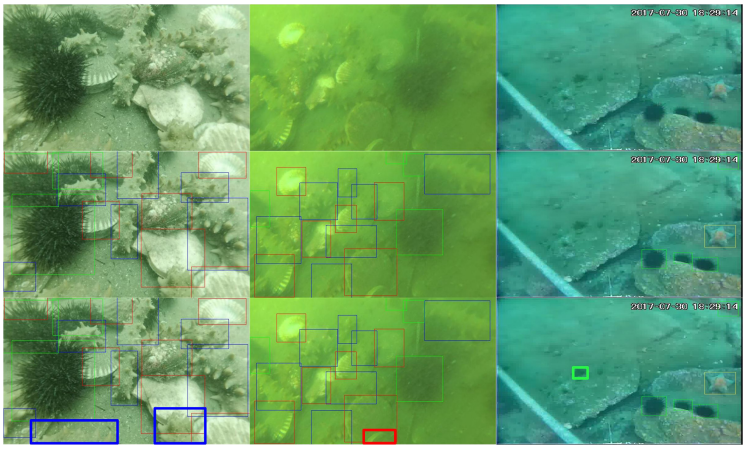


Fig. 1. Comparison of the baseline and our algorithm. (Best viewed in color and with zoom in) The upper part is the original test image. The middle part is the detection result of the baseline and the lower part is the result of our algorithm, the areas with obvious contrast are marked by bold lines. (Color figure online)

In recent years, generic object detection based on Convolutional Neural Network (CNN) [30] occupies a dominant position in object detection research. The mainstream object detectors can be divided into two categories: (1) the one-stage object detectors [18,24,25] and (2) the two-stage object detectors [6,11,23,26]. One-stage object detectors can directly localize objects by matching the large number of prior boxes, which been densely sampling on the input image at different scales and ratios. This method has a strong advantage in efficiency, but accuracy is usually low. In contrast, two-stage detectors can obtain more accurate results by generating object proposals first and then further calculate classification scores and regression bounding-box. In this work, we will focus on a one-stage object detection framework.

To deal with some real-time object detection tasks, a variety of one-stage object detection methods [15,18,24,36] have been introduced. In these methods, the Single Shot Multi-box Detector (SSD) [18] gains popularity because of its excellent performance and high speed. The standard SSD framework uses VGG16 [27]as backbone and adds a series of extra layers at the end of it. These additional layers and several former convolutional layers are used to predict the objects. Due to the use of a pyramid structure [1,14,22], each prediction layer conducts independent predictions with a specific scale in the standard SSD. SSD possesses high detection efficiency, but its accuracy performance still behind modern two-stage detectors.

During our research, we find that when many superior generic object detection frameworks are directly applied to the underwater task, they can hardly maintain high accuracy and robustness (Fig. 1). For example, Faster-RCNN [26] is affected by the invariance of the CNN scale. It is difficult to deal with the

problem of scale variation under the water. Due to the existence of the Regional Proposal Network (RPN), it can hardly meet the real-time requirements. SSD [18] can detect at a high speed, but there will be a problem of missing detection for small and blur objects under the water. Despite generic object detectors encounter some problems, they still have inspiration for the detection research in the underwater scene. Most approaches [1,15,36] adopt a top-down pyramid representation, which injects high-level semantic information into a high-resolution feature map to solve the scale problem. To process the occlusions problem, the data augmentation method called Mix up [35] becomes popular. This method can simulate occlusion samples during the training phase, thereby enhancing the ability of the model to discriminate occluded objects. In this work, we are devoted to improving a generic one-stage object detection algorithm to adapt it in underwater detection tasks.

Motivated by the works above, we propose a one-stage underwater object detection algorithm named **FERNet**. Our contributions are mainly as follows:

- To deal with blurring and texture distortion problems in underwater dataset, we introduce a Composite Connection Backbone (CCB) to enhance the feature representation, rather than finding a brand-new deeper backbone.
- To solve the problem of scale variation and sample imbalance, we introduce a Receptive Field Augmentation Module (RFAM) to enrich multi-scale contextual features and provide the Prediction Refinement Scheme (PRS) to align features with anchors.
- We have collected and integrated a large number of relevant images from the Internet, then form a brand-new UnderWater Dataset.

To sum up, we integrate and expand the existing underwater dataset. In the algorithm, we connect two pre-trained backbones to enhance feature extraction capabilities. The combination of the top-down pyramid structure and the receptive field enhancement module can instill multi-scale semantic features into the network. We also introduce RFAM to enrich multi-scale contextual features. Finally, PRS first performs binary classification to distinguish fore-background and then conducts preliminary localization. Afterward, refining previous results to get the final classification scores and bounding-box regression.

2 Related Work

One-Stage Object Detection. The current mainstream one-stage detectors have mostly followed the work based on YOLO [24] or SSD [18]. YOLO uses a forward convolutional network directly to predict object categories and locations on the dense feature map. It is the first work to achieve end-to-end detection. On this basis, there are many developments [25,31] in the follow-up. Different from YOLO, SSD introduces anchors and dense multi-scale feature maps into one-stage object detectors. It uses a pyramidal hierarchical structure to do prediction. Through this structure, the shallow texture information and deep semantic information can be combined to make the network achieve stronger

representation capabilities. Meanwhile, the dense anchor boxes also bring the overwhelming easy background samples, which limit the accuracy of one-stage object detectors. To solve this problem, RetinaNet [15] utilizes a novelty loss function named Focal Loss to down-weight the contribution of easy samples and makes network focus on the difficult samples. RefineDet [36] proposes a cascade prediction method to remove the background anchors in advance and then refine the anchors to boost detection performance. FCOS [29] uses the anchor-free method, which fundamentally avoids the impact of dense anchors.

Underwater Detection and Its Challenges. Underwater object detection [10,12,21] is generally achieved by sonar, laser and camera. The sonar is sensitive to the geometric information of the object, but can only show the difference in the distance between scanning points. It always omits other factors (e.g. visual characteristics). The laser can provide high performance to accurately model underwater objects but too expensive. In contrast, the camera is low cost and it can catch more types of visual information with high temporal and spatial resolution. Certain prominent objects can be identified by color, texture, and contour visual features. With the development of computer vision and underwater robots, vision-based underwater object detection [2,4,5,13] becomes more and more popular.

The images obtained by underwater cameras often have problems like low contrast, distorted texture, and uneven illumination. Besides, affected by living habits, underwater creatures are densely distributed and vary in size. The camera acquisition will encounter serious occlusion and scale variation problems, which pose a challenge to CNNs with scale invariance. To deal with these problems. Lv *et al.* [20] proposes a weakly supervised object detection method, which improves the accuracy by the strategy of weak fitting the foreground-background segmentation network first and refining proposals. Considerable accuracy has been achieved by this method, but it is difficult to achieve real-time performance due to the deep feature extraction network. Lin *et al.* [16] improves Faster-RCNN and proposes an enhanced strategy called Roimix to simulate overlapping and occluded objects in the training phase. This method endows the model stronger generalization ability and improves the accuracy in the occlusion scene. However, the performance of this data augmentation strategy on the one-stage detector is limited. Different from the above research, we hope to improve the one-stage underwater detector through a structural method.

Methods for Anchor Refinement. The accuracy of traditional one-stage detectors is often inferior to two-stage detectors. The main reason is that two-stage detectors have a fine-tuning process for the initial anchors but this process is omitted in one-stage detectors. Therefore, a large number of anchors caused the problem of anchor imbalance. In order to solve this problem, RefineDet [36] uses two-stage regression to get more refined results. It filters out a large number of negative anchors through the first time classification so that the positive and negative samples can be balanced, and then refines anchors based on the first time regression to obtain more accurate results. Although RefineDet can perform regression and classification of multiple stages, the features of different stages

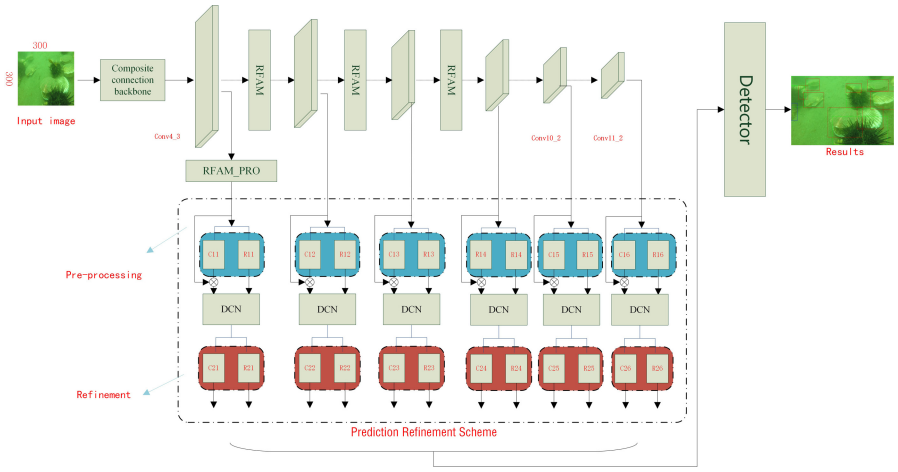


Fig. 2. Overall architecture of our framework. It specifically shows the down-sampling process after conv4_3, the composite connection backbone, the receptive field enhancement module and the prediction refinement scheme. (C_{1x}, R_{1x}) represents the results of the pre-processing phase and (C_{2x}, R_{2x}) represents the final result after the refinement process.

are the same. In fact, the anchors have changed after the first regression, and subsequent operations should rely more on the updated anchor. Therefore, AlignDet [3] learns the offset before and after the regression through the Deformable Convolution Network (DCN) [7], thereby solving the problem of feature misalignment to a certain extent. Reppoints [33] uses weak supervision to locate key points and predict their offsets, which is used as the offsets of DCN to convolution the original feature map so that the features are aligned with the object area.

3 Method

Our improved underwater object detection algorithm is based on standard SSD structure, which consists of the following components (see Fig. 2): **(1)** Composite Connection Backbone (CCB); **(2)** Receptive Field Augmentation Module (RFAM); **(3)** Prediction Refinement Scheme (PRS). The composite connection backbone network combines two common backbones. With the purpose of reducing time costs in searching for a new powerful backbone, we make up two existing backbones by a new way of composite connection to maximize the potential of them. The combined powerful feature extraction network has a stronger ability to represent the detailed features of the underwater object, which mainly deals with underwater blur problems. The RFAM is used to process the extracted information. Through RFAM, the reception field can be increased by individual kernels in different dilation rate and the multi-scale contextual features are

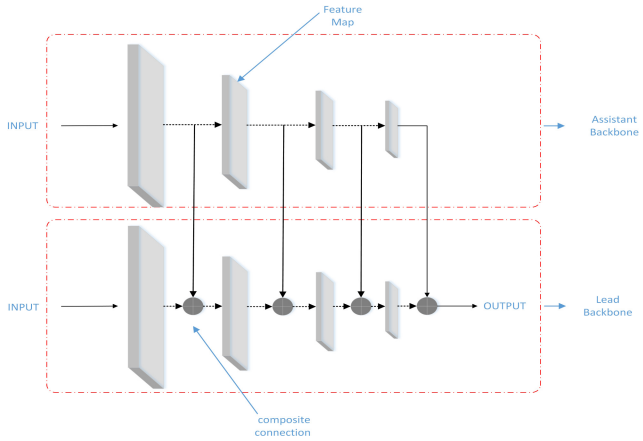


Fig. 3. Composite Connection Backbone. It shows the implementation details of our composite connection. The $\oplus$ means the fusion of two different features. In fact, we only have three layers of composite connection in actual use.

better expanded, which makes the information involved in prediction more discriminative. Our prediction refinement scheme is used to perform regression and classification operations on the prediction anchors. This scheme can both refine the anchors and features. In this step, PRS can roughly distinguish the foreground and background, giving the location on the whole, and then refine the anchors to get the final improvement results.

As we can see in the overall architecture, we utilize the new structure of composite connection to replace the VGG16 in the original standard SSD, and the input image size is 300×300. After the backbone, RFAM is interspersed between extra layers of standard SSD. In PRS, we use DCN to correct the offset of anchors after the first classification and regression, the outputs of DCN guide the second classification and regression and finally output the more accurate results.

3.1 Composite Connection Backbone

The underwater dataset has severe blurring and texture distortion problems. These problems often make it difficult for some networks to extract key feature information and affect the discrimination ability of the classifier. To this end, a feature extraction network with stronger representation capabilities is desperately needed. We first rule out the use of deeper feature extraction backbones, as this would slow down the speed of the one-stage detector, but redesigning a new and effective structure is difficult and time-consuming. So we explored the relationship between the extracted features of different backbones. Inspired by CBNet [19], we combine the existing characteristic backbone networks and get more performance than the single backbone.

The proposed composite connection backbone is shown in Fig. 3. The whole new backbone is divided into two parts: the lead backbone and the assistant backbone. The lead backbone still uses the standard VGG16 structure, and we use ResNet50 structure as the assistant backbone. Our proposed method is to replace the original backbone network with a composite connection form of these two basic backbones. In the assistant backbone, the result of each stage can be regarded as a higher-level feature. The output of each feature level is a part of the lead backbone input and flows to the parallel phase of subsequent backbones. In this way, multiple high-level and low-level features are fused to generate richer feature representations. This process can be expressed as:

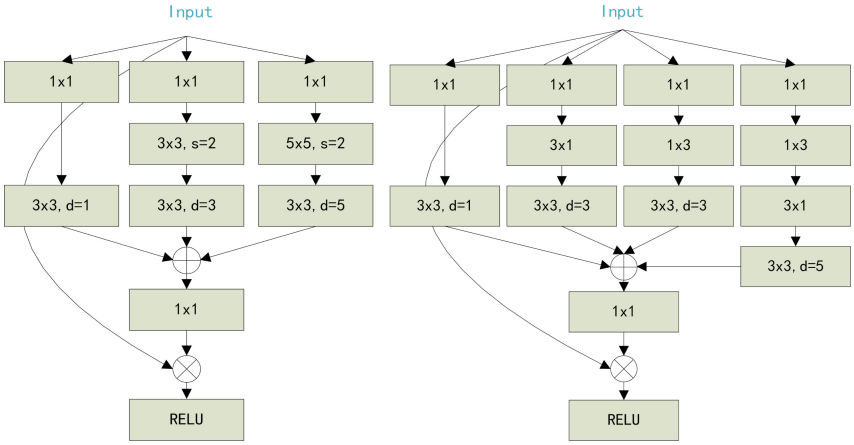
$$F_{out} = F_l \oplus F_a \tag{1}$$

$$F_{OUT} = \varepsilon(F_{out}) \tag{2}$$

where $\oplus$ is the process of feature fusion, F_l denotes the output features of the lead backbone at the current stage and F_a denotes the output features of the assistant backbone, we use F_{out} to show the results of feature fusion and F_{OUT} is used as the input value of the next layer in the lead backbone. The process from F_{out} to F_{OUT} goes through the channel adjustment. As is shown in Eq. 2, ε work as a convolution operation of 1×1. In theory, we can use this kind of composite connection method at each layer of the backbone, and our experiment only uses one of the most basic and useful composite connection methods. In fact, the connection between the lead backbone and the assistant backbone can be designed more complicated. We can also select feature layers of different sizes on the lead backbone and the assistant backbone, and bilinearly interpolate to the same size for the composite connection. This shows that our composite connection method is not limited by the size of the feature size. In order to simplify the operation, we select 150×150, 75×75, and 38×38 characteristic layers on the lead backbone, which corresponds to the output of three layers of ResNet50.

3.2 Receptive Field Augmentation Module

Figure 4 shows the receptive field augmentation module we introduced, which reproduces the work of RFB [17]. In order to imitate the design idea of ResNext [32] and the Inception structure [28], RFAM has multiple branches structure. First, the multiple branches of structure processes input data in parallel. Each branch is composed of 1×1 convolution and several other simple convolutions with different kernel sizes, and finally, each branch forms a structure similar to the bottleneck. The convolution kernel size of each branch changes slightly, which is conducive to capture the multi-scale contextual information. Aiming to expand the receptive field, we use the dilated convolution [34] with different dilation rate to enhance the multi-scale features, then the features of multiple branches are fused, after that, we use 1×1 convolution to adjust the channel size. Finally, we also simulate the residual structure using the shortcut connection method, weighting the input and feature fusion results, then obtain the



(a) RFAM (b) RFAM_PRO

Fig. 4. Receptive Field Augmentation Module. RFAM is used alternately in the downsampling layer to expand the receptive field of the feature map. RFAM_PRO is used on shallow feature maps to help detect small objects.

final output through ReLU. To adapt to a variety of situations, we proposed two RFAM structures, RFAM and RFAM_PRO. RFAM_PRO has more branches than RFAM and use many small convolution kernels, which is friendly to small object detection. RFAM_PRO replaces a 5×5 convolution with two superimposed 3×3 convolution. This can reduce the number of parameters to decrease the computational complexity, and increase the nonlinearity of the model. Furthermore, we replace the original 3×3 convolution with a 1×3 convolution and 3×1 convolution. The whole process of RFAM can be expressed by Eq. 3:

$$X_{out} = \tau(X_{in} \otimes \epsilon(Br1 \oplus Br2 \oplus Br3) \times scale) \tag{3}$$

here X_{in} represents the input feature, Br1, Br2, and Br3 denote the output of three branches, $\oplus$ is the operation of feature fusion. We use ϵ to represent the process of adjusting the number of channels through 1×1 convolution, the value of scale is the weight of linear operation in shortcut, here we take 0.1. $\otimes$ represents the element-wise addition, and finally, τ is the activation function of ReLU.

3.3 Prediction Refinement Scheme

Our prediction refinement scheme mainly includes two steps: Pre-processing and Refinement. As shown in Fig. 2, this process uses two-step treatment to refine the prediction of object's locations and sizes, which is good for the challenging underwater scenarios, especially for the small objects. The prediction refinement scheme mainly performs initial binary classification and regression in the pre-processing stage, and then the refinement module obtains the final result

based on the pre-processing results. The main process will be explained in detail below. Different from RefineDet [36], our prediction refinement scheme uses six feature prediction layers for refinement. Moreover, PRS can aggregate important features through a designed attention mechanism and refine anchors by learning from offsets. We will confirm the advantages of our structure through later experiments.

Pre-processing: In the pre-processing phase, the prediction value obtained by the Receptive Field Enhancement Module (RFAM) and the extra layer is processed first. In Fig. 2, starting from the last layer conv4_3 of the composite connection backbone, downsampling through the additional layers of the standard SSD and the RFAM to reach the size required by the prediction layer. What is special is that conv4_3 is followed by an RFAM_PRO to strengthen the detection ability of shallow features to small objects. We believe that adding RFAM_PRO to large-scale feature maps can fully extract the semantic information of high-resolution feature maps, so operating on high-resolution feature maps is conducive to the detection of the small underwater objects. Finally, binary classification and box regression are performed on the information of the six enhanced feature layers. Filter the obvious background first in preparation for the refinement module. The output C_{1x} is used to distinguish the foreground and background. R_{1x} includes four important values, which are used to locate the anchors.

Refinement: In this stage, we perform the max-pooling operation along the channel axis for the pre-processing result C_{1x} and then carry out the Sigmoid function to gain better features. The result of this process is recorded as S_{1x}. S_{1x} obtained through max-pooling and Sigmoid operations can highlight the position of the object, which is used to enhance the result X_{out} of six prediction layers. S_{1x} and X_{out} are multiplied element by element and then added to X_{out}. The result is recorded as X_{end}. Generally speaking, we replace the RefineDet's TCB module with an attention mechanism module, making the network pay more attention to the object itself. This process can be expressed by Eq. 4:

$$X_{end} = (X_{out} \odot S_{1x}) \otimes X_{out} \tag{4}$$

where $\odot$ is element-wise multiplication, $\otimes$ means element-wise addition, and X_{end} denotes the amount of enhancement of existing foreground position information. In the previous R_{1x} regression, four output values are obtained: $\triangle x$, $\triangle y$, $\triangle h$ and $\triangle w$. The first two values ($\triangle x$, $\triangle y$) represent the spatial offsets of the center point of the anchor and the last two values ($\triangle h$, $\triangle w$) represent the offsets of the size. To align features, we fine-tune the anchor frame through the DCN. Specifically, we compute the kernel offsets by $\triangle x$ and $\triangle y$ in location offsets layers, which combine with the X_{end} as the input of DCN. We also use dilated convolution in deformable convolution to enhance the semantic relevance of context. About the classification and regression in the refinement stage, C_{2x} no longer simply performs binary classification but performs multiple classification tasks. We gain the final positioning result R_{2x} through the output of DCN.

On the whole, in order to obtain more fine-grained positioning results, we adopt a strategy similar to RefineDet. We apply DCN to this process and the results of the pre-processing stage are used to calculate the feature offsets and then send to DCN to align the features. The refinement phase is fine-tuned for the best results.

4 Experiments

We perform experiments on PASCAL VOC 2007 [9] and UWD. Mean Average Precision (mAP) is adopted for the evaluation metric. Our underwater object detection algorithm has the advantages of strong feature extraction capabilities, multi-scale detection, and anchor refinement to solve the underwater issues, below we will focus on three parts: implementation details, detection performance, and ablation experiment. Our algorithm is mainly oriented to the underwater environment, so the experimental result of the UWD is used as the main evaluation criteria. To verify the feasibility of the framework, we also performed experiments on PASCAL VOC 2007 benchmarks.

4.1 Implementation Details

Our framework utilizes the composite connection of VGG16 and ResNet50 as the backbone. Both VGG16 and ResNet50 are pre-trained on ImageNet [8]. About the experiments on two datasets, we keep the consistent initial experiment settings and choose the same optimizer (SGD). Our six prediction branches of PRS use the anchor scale of [6, 6, 6, 6, 4, 4] and aspect ratio of 2:3 and 2:2. During the training phase, we adopt a warm-up strategy. The learning rate of the first six epochs is randomly selected between 10^{-6} and 4×10^{-3}, and gradually approach the basic learning rate 0.002. After that, it decreases ten times each time. The PASCAL VOC 2007 and underwater dataset decrease to the lowest learning rate at the last 10 epochs respectively. Non-maximum Suppression (NMS) with Intersection over Union (IoU) threshold 0.5 is adopted for post-processing. To simulate the occlusion problem in the underwater environment, we also add a random erasing strategy to the data augmentation during the training session. In our experiments, PASCAL VOC 2007 and underwater dataset were trained for 160 epochs and the batch size was set to 32. Besides, we used two Nvidia RTX2080Ti for training, our code is based on the deep learning framework of PyTorch.

4.2 Detection Performance

PASCAL VOC 2007: In order to verify the rationality of our underwater detection framework, we perform experiments on the PASCAL VOC 2007 dataset and compared it with the general detection methods in existing papers. We both perform experiments with input sizes of 300×300 and 512×512. Table 1 shows the test results of PASCAL VOC 2007. We can see that our algorithm achieves

Table 1. Results on PASCAL VOC 2007 testset, trained on 07 and 12 train-val dataset. VGG16* represents our composite connection backbone.

Method	Backbone	Input size	mAP
Two-Stage Detectors:			
Faster RCNN	VGG16	1000 × 600	73.2
Faster RCNN	ResNet101	1000 × 600	76.4
MR-CNN	VGG16	1000 × 600	78.2
R-FCN	ResNet101	1000 × 600	80.5
CoupleNet	ResNet101	1000 × 600	82.7
Single-Stage Detectors:			
SSD300	VGG16	300 × 300	77.2
YOLO	GoogleNet	448 × 448	63.4
YOLOV2	DarkNet-19	544 × 544	78.6
RON320++	VGG16	320 × 320	76.6
DSSD321	ResNet101	321 × 321	78.6
RefineDet320	VGG16	320 × 320	80.0
DES300	VGG16	300 × 300	79.7
DFPR300	VGG16	300 × 300	79.6
RFBNet300	VGG16	300 × 300	80.5
EFIPNet	VGG16	300 × 300	80.4
FERNet (Ours)	VGG16*	300 × 300	80.2
SSD512	VGG16	512 × 512	79.5
DSSD512	ResNet101	513 × 513	81.5
DES512	VGG16	512 × 512	81.7
RefineDet512	VGG16	512 × 512	81.8
DFPR512	VGG16	512 × 512	81.1
EFIPNet512	VGG16	512 × 512	81.8
RFBNet512	VGG16	512 × 512	82.1
FERNet (Ours)	VGG16*	512 × 512	81.0

the result of 80.2 mAP on PASCAL VOC 2007, leading most of the one-stage detectors, and even surpassing most two-stage detection algorithms. However, from the data in Table 1, the accuracy of our algorithm has not reached the state of art. The main reasons may be as follows: Firstly, the PASCAL VOC 2007 has a few cases of blurred, occlusion, and scale variation, so our algorithm doesn't improve much. Secondly, the detection accuracy fluctuates due to the influence of hardware equipment and the Anaconda environment. For example, we use RFBNet source code for training, and the detection accuracy can only reach 80.0 mAP. Our algorithm is 0.2 mAP higher than the RFBNet. Compared with RefineDet, although it has only a 0.2 map improvement, our input size is

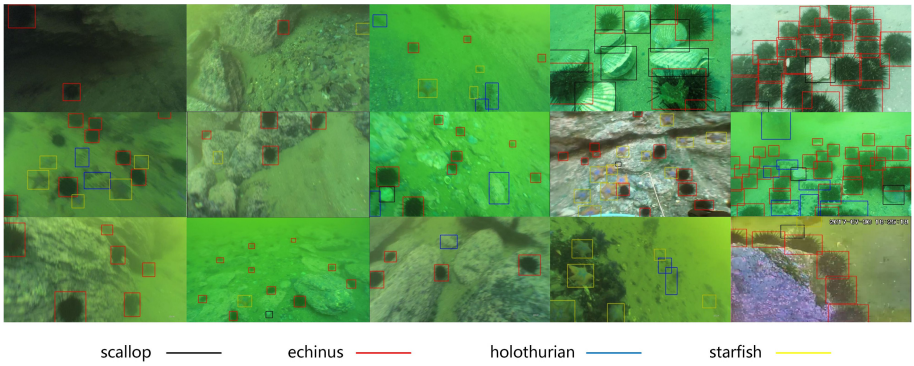


Fig. 5. Qualitative detection results of our detector on UWD dataset. Each color of an anchor belongs to an object class.

300 × 300, which is less than 320 × 320 of the RefineDet input size. Our methods can save computing resources and time while still achieving accuracy improvements. In a word, the result in Table 1 demonstrates that despite our detection framework does not reach the current best accuracy, it can still maintain a high detection level in general object detection.

UWD: We comprehensively evaluate our method on the UnderWater Dataset (UWD). We collect and integrated relevant underwater pictures on the Internet, and perform manual data annotation through Label Image[1] to expand the URPC dataset[2], and finally form our dataset. Our underwater dataset contains 10 thousand train-val and test images, it contains four classes: holothurian, echinus, scallop, and starfish. In the experiments, we use the same network structure and parameter settings like that in the PASCAL VOC experiment and make the result of RFBNet as the baseline. Table 2 shows that the accuracy of the baseline algorithm on UWD can only reach 60 mAP, but FERNet can reach an accuracy of 74.2 mAP. Compared with the baseline algorithm, our algorithm has improved by nearly 14.5% points on the UWD dataset. The algorithm has certain feasibility when dealing with complex underwater environments.

Figure 5 is the detection result of our underwater detection framework on the UWD dataset. Four classes of holothurian, echinus, scallop, and starfish are marked by blue, red, black, and yellow boxes. As is shown from the image, our detection framework can maintain a high performance when facing problems like underwater occlusion, blur, color shift, and uneven lighting.

4.3 Ablation Experiment

In this part, we will perform ablation experiments on various functional blocks of FERNet to verify their effectiveness. We gradually add functional modules

[1] Datasets Annotation Tool. https://github.com/tzutalin/labelImg.
[2] Underwater Robot Picking Contest. http://www.cnurpc.org/.

Table 2. Detection results on underwater dataset UWD. The baseline represents the result of RFBNet. RE represents the trick of random-erasing. FERNet^{+} represents that we use focal loss on the basis of FERNet, FERNet* means we use VGG16 with BN as our lead backbone. The above results are obtained by experiments on images with an input size of 300 × 300.

Method	mAP	CCB	RE	Focal loss	PRS	Holothurain	Echinus	Scallop	Starfish
Baseline(BL)	60.0					56.0	77.1	35.9	71.2
BL+C	63.8	√				63.4	78.3	38.2	75.3
BL+F	63.2			√		60.2	79.0	38.1	75.4
BL+C+F	66.6	√		√		61.9	83.3	42.9	78.5
BL+C+R+F	66.7	√	√	√		62.1	83.5	42.7	78.6
FERNet	**74.2**	√	√		√	**71.4**	**91.5**	**52.2**	82.0
FERNet^{+}	73.0	√	√	√	√	71.1	90.8	48	**82.2**
FERNet*	73.0	√	√		√	68.3	90.7	51.2	81.8

Table 3. Ablation experiments on the number of prediction layers. The results of PASCAL VOC and UWD show that six prediction layers are better than that of four.

Datasets	Num	mAP
PASCAL VOC	4	79.8
PASCAL VOC	6	**80.2**
UWD	4	73.3
UWD	6	**74.2**

based on the baseline to observe the changes in the results. Besides, we also added some tricks like random erasing and BatchNormalization to participate in comparing the results.

Rationality of Three Functional Modules. In order to prove the rationality of these functional blocks proposed in this paper, we have carefully designed multiple comparative experiments as shown in Table 2. We gradually add them to observe the changes in the experimental results. Firstly, we added the Composite Connection Block (CCB) module to the baseline algorithm. We can see that a single composite connection function block can provide 3.8 mAP gains. To maximize the potential of a single composite connection block module, we use tricks of Random-Erasing [37] and focal loss, which can finally achieve a gain of 6.7 mAP compared to the baseline algorithm. Secondly, we continue to increase the functional modules of the prediction refinement scheme (PRS) on this basis and reach the best accuracy of 74.2 mAP. The process above can prove the rationality of our proposed functional modules.

BatchNorm in Backbone. BatchNorm is widely used to enable fast and stable training of deep neural networks [38]. To investigate whether BatchNorm has an improvement on the backbone, we add the BatchNorm operation to each

convolution layer in the VGG16 network, denoted as FERNet*. As shown in Table 2, FERNet* only gain the accuracy of 73.0 mAP, 1.2 mAP lower than VGG16 without BatchNorm. This indicates that the BatchNormalization layer does not significantly improve the accuracy of our algorithm.

Number of the Prediction Layers. Our refinement module is inspired by the RefineDet and it is similar but essentially different. The selection and design details of our prediction layers have been explained before. The ablation experiment here is mainly to analyze the rationality of the number of the prediction layers. We know that in RefineDet, the author selected four feature layers for prediction, and confirm that the selection of the four layers can achieve the best accuracy. In our experiments, we believe that small feature maps are also necessary to preserve, because, for our underwater dataset, high-level semantic information can help us identify something more detailed and benefit for those blurry and distorted image detection. Therefore, we perform another small ablation experiment on selecting the number of prediction layers in our experiment. We use four prediction layer structures and six prediction layer structures for experiments. Table 3 proves that six prediction layers are better than four prediction layers on both UWD and PASCAL VOC 2007 datasets.

In summary, the experiments above show that all the functional modules we proposed have a significant improvement in detection accuracy, especially the PRS. On this basis, we find that using PRS with focal loss together can not get accuracy improvement, probably because Online Hard Example Mining (OHEM) has been used in PRS. Strangely, the improvement of our algorithm on PASCAL VOC 2007 is not obvious. We guess that it is because the PASCAL VOC 2007 dataset rarely has blur, scale variation, and occlusion problems.

5 Conclusion

In this paper, we analyze the challenging problems which affect the performance of object detection in the underwater environment. To address these issues, we propose a one-stage underwater detection framework named FERNet. We combine existing feature extraction backbones by the form of a composite connection to propose a backbone with stronger feature expression capabilities. Further, we introduce the receptive field enhancement module, which is used to enrich the receptive field, expand multi-scale contextual features, and boost the discrimination ability of the entire detection network. Finally, we utilize the prediction refinement scheme to align the features with anchors to deal with the problem of sample imbalance and feature misalignment to some extent. Experiments show that our detection algorithm has great improvements compared to the baseline algorithm on the underwater dataset.

Acknowledgments. This work is supported by the Ministry of Science and Technology of the People's Republic of China (2019YFB1310300), National Natural Science Foundation of China (No. 61876092), State Key Laboratory of Robotics (No. 2019-O07) and State Key Laboratory of Integrated Service Network (ISN20-08).

References

1. Cao, J., Pang, Y., Li, X.: Triply supervised decoder networks for joint detection and segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 7392–7401 (2019)
2. Chen, X., Lu, Y., Wu, Z., Yu, J., Wen, L.: Reveal of domain effect: how visual restoration contributes to object detection in aquatic scenes. arXiv. Computer Vision and Pattern Recognition (2020)
3. Chen, Y., Han, C., Wang, N., Zhang, Z.: Revisiting feature alignment for one-stage object detection. arXiv preprint arXiv:1908.01570 (2019)
4. Chen, Z., Zhang, Z., Dai, F., Bu, Y., Wang, H.: Monocular vision-based underwater object detection. Sensors **17**(8), 1784 (2017)
5. Cong, Y., Fan, B., Hou, D., Fan, H., Liu, K., Luo, J.: Novel event analysis for human-machine collaborative underwater exploration. Pattern Recogn. **96**, 106967 (2019)
6. Dai, J., Li, Y., He, K., Sun, J.: R-FCN: object detection via region-based fully convolutional networks. In: Advances in Neural Information Processing Systems, pp. 379–387 (2016)
7. Dai, J., et al.: Deformable convolutional networks. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 764–773 (2017)
8. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: ImageNet: a large-scale hierarchical image database. In: 2009 IEEE Conference on Computer Vision and Pattern Recognition, pp. 248–255. IEEE (2009)
9. Everingham, M., Van Gool, L., Williams, C.K., Winn, J., Zisserman, A.: The pascal visual object classes (VOC) challenge. Int. J. Comput. Vis. **88**(2), 303–338 (2010). https://doi.org/10.1007/s11263-009-0275-4
10. Galceran, E., Djapic, V., Carreras, M., Williams, D.P.: A real-time underwater object detection algorithm for multi-beam forward looking sonar. IFAC Proc. Vol. **45**(5), 306–311 (2012)
11. He, K., Gkioxari, G., Dollár, P., Girshick, R.: Mask R-CNN. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 2961–2969 (2017)
12. Henriksen, L.: Real-time underwater object detection based on an electrically scanned high-resolution sonar. In: Proceedings of IEEE Symposium on Autonomous Underwater Vehicle Technology (AUV 1994), pp. 99–104. IEEE (1995)
13. Li, C., Anwar, S., Porikli, F.: Underwater scene prior inspired deep underwater image and video enhancement. Pattern Recogn. **98**, 107038 (2020)
14. Lin, T.Y., Dollár, P., Girshick, R., He, K., Hariharan, B., Belongie, S.: Feature pyramid networks for object detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2117–2125 (2017)
15. Lin, T.Y., Goyal, P., Girshick, R., He, K., Dollár, P.: Focal loss for dense object detection. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 2980–2988 (2017)
16. Lin, W.H., Zhong, J.X., Liu, S., Li, T., Li, G.: RoIMix: proposal-fusion among multiple images for underwater object detection. arXiv preprint arXiv:1911.03029 (2019)
17. Liu, S., Huang, D., Wang, Y.: Receptive field block net for accurate and fast object detection. arXiv preprint arXiv:1711.07767 (2017)
18. Liu, W., et al.: SSD: single shot MultiBox detector. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016. LNCS, vol. 9905, pp. 21–37. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46448-0_2

19. Liu, Y., et al.: CBNet: a novel composite backbone network architecture for object detection. arXiv preprint arXiv:1909.03625 (2019)
20. Lv, X., Wang, A., Liu, Q., Sun, J., Zhang, S.: Proposal-refined weakly supervised object detection in underwater images. In: Zhao, Y., Barnes, N., Chen, B., Westermann, R., Kong, X., Lin, C. (eds.) ICIG 2019. LNCS, vol. 11901, pp. 418–428. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-34120-6_34
21. Mullen, L.J., et al.: Modulated laser line scanner for enhanced underwater imaging. In: Airborne and In-Water Underwater Imaging, vol. 3761, pp. 2–9. International Society for Optics and Photonics (1999)
22. Pang, Y., Wang, T., Anwer, R.M., Khan, F.S., Shao, L.: Efficient featurized image pyramid network for single shot detector. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 7336–7344 (2019)
23. Purkait, P., Zhao, C., Zach, C.: SPP-Net: deep absolute pose regression with synthetic views. arXiv preprint arXiv:1712.03452 (2017)
24. Redmon, J., Farhadi, A.: YOLO9000: better, faster, stronger. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 7263–7271 (2017)
25. Redmon, J., Farhadi, A.: YOLOv3: an incremental improvement. arXiv preprint arXiv:1804.02767 (2018)
26. Ren, S., He, K., Girshick, R., Sun, J.: Faster R-CNN: towards real-time object detection with region proposal networks. In: Advances in Neural Information Processing Systems, pp. 91–99 (2015)
27. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition (2014)
28. Szegedy, C., Ioffe, S., Vanhoucke, V., Alemi, A.A.: Inception-v4, Inception-ResNet and the impact of residual connections on learning. In: Thirty-First AAAI Conference on Artificial Intelligence (2017)
29. Tian, Z., Shen, C., Chen, H., He, T.: FCOS: fully convolutional one-stage object detection. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 9627–9636 (2019)
30. Touretzky, D.S., Mozer, M.C., Hasselmo, M.E.: Advances in Neural Information Processing Systems 8: Proceedings of the 1995 Conference, vol. 8. MIT Press, Cambridge (1996)
31. Wong, A., Famuori, M., Shafiee, M.J., Li, F., Chwyl, B., Chung, J.: YOLO Nano: a highly compact you only look once convolutional neural network for object detection. arXiv preprint arXiv:1910.01271 (2019)
32. Xie, S., Girshick, R., Dollár, P., Tu, Z., He, K.: Aggregated residual transformations for deep neural networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1492–1500 (2017)
33. Yang, Z., Liu, S., Hu, H., Wang, L., Lin, S.: RepPoints: point set representation for object detection, pp. 9657–9666 (2019)
34. Yu, F., Koltun, V.: Multi-scale context aggregation by dilated convolutions. arXiv preprint arXiv:1511.07122 (2015)
35. Zhang, H., Cisse, M., Dauphin, Y.N., Lopez-Paz, D.: Mixup: beyond empirical risk minimization. arXiv preprint arXiv:1710.09412 (2017)

36. Zhang, S., Wen, L., Bian, X., Lei, Z., Li, S.Z.: Single-shot refinement neural network for object detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 4203–4212 (2018)
37. Zhong, Z., Zheng, L., Kang, G., Li, S., Yang, Y.: Random erasing data augmentation. arXiv preprint arXiv:1708.04896 (2017)
38. Zhu, R., et al.: ScratchDet: training single-shot object detectors from scratch. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2268–2277 (2019)