A Deep Learning-Based Embedding Framework for Object Detection and Recognition in Underwater Marine Organisms

Jinde Zhua (✉ 61201916@fjut.edu.cn)  
Fujian University of Technology  https://orcid.org/0000-0003-0359-1267

Research Article

Keywords: Deep learning, Computer vision, Multi-class classification, Fish detection, YOLOv4

Posted Date: November 2nd, 2021

DOI: https://doi.org/10.21203/rs.3.rs-986437/v1

License: ☺️ 📧 This work is licensed under a Creative Commons Attribution 4.0 International License. Read Full License
A deep learning-based embedding framework for object detection and recognition in underwater marine organisms

Jinde Zhu

Fujian Provincial Key Lab of Big Data Mining and Applications, School of Computer Science and Mathematics, Fujian University of Technology, Fuzhou 350118, China

Article history:

Keywords:
- Deep learning
- Computer vision
- Multi-class classification
- Fish detection
- YOLOv4

The detection of marine organisms is an important part of the intelligent strategy in marine ranch, which requires an underwater robot to detect the marine organism quickly and accurately in the complex ocean environment. Based on the latest deep learning arithmetic, this paper put forward to find the marine organism in a picture or video to construct a real-time objective invention system for marine organisms. The neural network arithmetic: YOLOv4 was employed to extract the deep features of marine organisms, implementing the accurate detection and size detection of different fish can use arithmetic for evaluation in fisheries. Furthermore, improving the architecture of the backbone and the neck connection is called YOLOv4-embedding. As a result, compared with other object detection arithmetic, YOLOv4-embedding object detection arithmetic was better at detection accuracy—higher detection confidence and higher detection ratio than other one-stage object detection arithmetic, EfficientDet-D3 example. The consequence demonstrates that the suggested instrument could implement the rapid invention of different varieties in marine organisms. Compared to the YOLOv4, the mAP75 of the YOLOv4-embedding achieves an improvement of 2.92% for the marine organism dataset at a rapid rate of ~51 FPS on RTX 3090, 60.8% AP50 for the MS COCO dataset.

Introduction

With the development of fish farms and artificial intelligence, robots have fascinated wide advertence aquaculture, [1], among which the underwater robot is one of the most popular ones. Underwater robots have significantly advanced recently in velocity and accuracy, and aquaculture robots are applied to harvest fish and marine organisms.

Computer vision (CV) is also exploited for fish recognition and weight estimation. The computer vision system is the key to achieve automatic harvest, and accurate detection is the precondition of follow-up operations. However, there is still a great challenge to realize robust and efficient object detection in marine organisms due to the resemblance of
occlusion among objects to detect or realize detection between objects and background. It even makes this kind of task more difficult to consider the complexity and the uncertainty of the ocean circumstance.

Traditionally, classic machine learning methods with multi-dimensional input features such as Adaboost, SVM, or Random Forest to realized object detection, that have been used to complete knowledge pattern mining. Before the detection step, image feature extraction operations such as Haar[2], SIFT, or HOG have to be adopted to establish the input features. Thus, one can find that the two processes, object detection, and feature extraction, are completely independent.

Since traditional image feature extraction operators are basically based on subjective judgments and are fully separated from the detection step, there are shortcomings of difficulty in extracting task-dependent features and weak generalization ability in object detection.

Researchers turn to deep learning to combine feature extraction and object detection into one operation and achieve end-to-end learning. Convolutional Neural Network (CNN) lays the foundation. Most of the current deep learning methods improve the learning quality by increasing the depth of CNN, which achieved great success in tasks for images and texts. Different from traditional neural networks, CNN introduces convolution and pooling operations to extract automatic features. Generally, CV-related CNN algorithms can be plotted into two types according to the application purpose: classification networks[3], detection networks[4].

Liu et al. [5] proposed DenseNet for weight sharing and substantially decreased the number of parameters in 2017. Khan et al.[6] present a novel structure enhancement of “Channel Boosting” in a deep CNN. This uses both the channel dimension of CNN and Transfer learning in 2018. Misra shown activation function MISH which has better performance than ReLU in the most recently proposed deep networks on benchmark datasets in 2019. Channel Boosting CNN has also been assessed on the medical graphics dataset (Aziz et al.) in 2020, where it has shown better results.

The detection networks can be further separated into two types, one is the foundation of the candidate regions (two-stage detector), and the other is the foundation of the regression way (one-stage detector). R-CNN series are the typical networks based on candidate regions. R-CNN was first proposed by Girshick et al. in 2014, in which CNN was used to extract features and SVM for classification. For better object detection performance, Girshick presented Fast R-CNN in 2015. Later the same year, Ren et al. proposed Faster R-CNN to improve the performance of Fast R-CNN. In 2016, Lin et al. [7] proposed FPN to further improve the performance of Faster R-CNN. He et al. proposed Mask R-CNN, and enlarging work to Faster R-CNN mainly, for instance, segmentation assignment in 2017, and Mask R-CNN can be a more accurate object detector.

Redmon adjusted the anchor box and presented YOLOv2 with improved multi-scale target detection. In
2017, Lin et al. [8] proposed RetinaNet to imbalance positive and negative samples in the dataset. In 2018, YOLOv3 with higher APS (small object average precision) performance on the small size objects were released, which has greatly improved the problems of the previous two versions, and it has a higher detection speed and accuracy.

Recently, Bochkovskiy et al. [9] introduced the latest version, YOLOv4, which is powerful than previous versions in AP (Average Precision) and FPS (Frames Per Second), a backbone network, neck network, and activation function improves with optimizations. In this paper, YOLOv4 will be employed.

This paper aims to develop a recognition system for Autonomous Underwater Vehicles to make it possible to recognize the surrounding environment and marine organisms. Specifically, using a normal RGB camera to obtain marine organisms images in the aquaculture ponds and object detection are carried out under different conditions of illumination and occlusion. The main contribution of this work includes:

(1) Based on the improvement of the newest detection algorithm YOLOv4, rapid and precise detection for marine organisms can be realized under various environmental conditions;

(2) Different neural network algorithms are compared and discussed the results of fish detection to verify the applicability, recommended a method in marine organisms detection.

Related work on fish and marine organisms detection

In recent years, deep learning-based CV techniques and object detection algorithms have been widely exploited in aquaculture, such as fish size measuring, body analysis, quality calculation, illness diagnosis, etc. As a kind of contactless method, high-precision CV techniques can monitor the size, the fabric, and the physical condition of the farmed organisms and became an important monitoring way in aquaculture. As mentioned before, CNN has been widely used in CV and particularly made a breakthrough in abstract cognitive problems. For example, based on the Fish4 Knowledge dataset, Rathi designed a 3-level CNN to classify 21 types of tropic fish.

By combining the feature selection framework and image segmentation, Marini implemented classification for fish species and assessed the abundance of fish on the collected data. Mandal combined Faster R-CNN with three classification networks (ZFNet, CNN-M, and VGG16) to realized regional prediction for fish and crustaceans collected from Queensland beaches. Konovalov designed an Xception CNN-based fish detector for a fish group and realized underwater fish detection in multiple water areas. The recent study that is based on YOLOv3 show that it can improve the detection performance for marine organisms by using the color and texture features of target objects.

However, the detection of marine organisms used CPU to train images in early work, the training speed was slower. Moreover, if the scattered background noise is not eliminated in the image, the detection ratio may also be reduced in marine organisms. Meanwhile, GPU processing to apply in images dataset makes training more effective
and faster.

In this work, the algorithm YOLOv4-embedding is present to realize fast detection for marine organisms. Compared with other arithmetics, YOLOv4-embedding is fast and better accurate. Thus, YOLOv4-embedding can balance high accuracy and fast speed for marine organisms recognition tasks, especially for underwater robots to perform fishing tasks. Another advantage of our work is that using a normal RGB camera instead of perplexed sensors greatly decreases the cost of collecting marine organisms’ images under shallow sea conditions.

Rest of this research is organized and listed below. In Section 3, the data and methods are defined. There is also briefly review the properties of the YOLOv4-embedding arithmetic. Different object detection arithmetic techniques are then compared in Section 4. Finally, experimental analysis is consulted and established in Section 5 and Section 6.

Architecture design of YOLOv4-embedding

Data and relevant methods

To validate the proposed method, using the data provided by The Institute of Oceanology, Chinese Academy of Sciences (IOCAS), which is collected in the real aquaculture environment. Specifically, 1557 valid marine organism images cover 4 types of marine organisms, i.e., Abalone, Echinoidea, Holothuroidea, and Thamnaconus modestus. A digital color camera (GoPro CHDHX) with a resolution of 1280 x 720 pixels is used to retrieve data, and the shooting angle was set to the front and side of the aquaculture ponds. Furthermore, some photos with an elevation angle of 45 were adopted for comparison.

FIG. 1. The overall network architecture of YOLOv4-embedding.
The training set, validation set, and test set include 1307, 100, and 150 images separately, and the mosaic image enhancement method is used to enhance the image set. Implement the algorithm with Python on a desktop computer (Intel(R)Core(TM) i7 10700 @2.9 GHz, 16.0 GB RAM, NVIDIA GeForce RTX 3090). Labeling, an open-source and free labeling tool, is used to tag each image. Once the marine organisms are labeled, an Extensible Markup Language (XML) document contains the labels and the coordinates of boxes that bound the target marine organisms in the image.

Detection procedure

In this subsection, the inside architecture of YOLOv4-embedding is presented in detail. YOLOv4 mainly includes 4 basic components, i.e., the input layer, backbone network (BackBone), neck network (Neck), and output layer. The input layer accepts fixed-size images extracted through the backbone network and sent to the neck network for feature fusion. The output layer outputs 3 different scale prediction anchor boxes called YOLO Head. Fig. 1 is the architecture of marine organisms detection based on the YOLOv4-embedding algorithm. The detection process is summarized as follows:

Step 1: Feed Marine organisms image into the network.

Step 2: The backbone is a CSPDarknet53 model, which keeps the Darknet53 framework, but uses the CSP mechanism. The Mish activation function and Leaky activation function are employed that abstracts the info from the image.

Step 3: Assemble SPP (Spatial Pyramid Pooling) module and FPN (Feature Pyramid Networks) + PAN (Path Aggregation Network) module to make use of the feature extracted by the backbone.
PAN uses feature pyramid and path aggregation technology to propagate low-level information to the top-level easier. With these components, the multi-scale prediction for three types of targets, i.e., large ones, medium ones, and small ones, can be performed.

Step 4: Embedding the convolution layer and linear activation function at the end of the YOLOv4 Neck. CBLR is employed in the network. The architecture is shown in Fig. 2. Concat task is the addition of tensors and dimensionality, which add the features of the two CBLR.

Step 5: The YOLOv4 head performs predicting, which outputs the final detection results.

Now, clarifying the concrete building blocks. The backbone is CSPDarknet53 architecture that comprises 5 CSP (Cross Stage Partial connections) modules, 11 CBM (Convolutional+ Batch normalization +Mish) modules and 10 CBL (Convolutional+ Batch normalization +Leaky). Particularly, the CBM module fulfills the convolution task by using Batch Normalization and Mish activation functions. In contrast, the CBL module fulfills the convolution task by using Batch Normalization and Leaky Relu activation functions.

![The Mish function and the Leaky Relu function.](image)

Leaky Relu activation function is prevalent in deep learning, whereas the average performance of the Mish function is better. Leaky Relu activation function and Mish activation function are combined in the YOLOv4-embedding backbone, which increases the accuracy of detection is an innovation of this work. Leaky Relu activation function is used in the rest part of the network. Specifically, Mish function is

\[ f(x) = x \cdot \tanh(\text{softplus}(x)) \]  \hspace{1cm} (1)

Leaky relu function is

\[ f(x) = \begin{cases} x & \text{if } x > 0 \\ \lambda x & \text{if } x \leq 0 \end{cases} \]  \hspace{1cm} (2)

The diagrams of the Mish function and Leaky Relu function are shown in Fig.3.

CSPx1 implies one Resnet part in the YOLOv4 backbone; CSPx4 implies 4 Resnet parts. After passing through the backbone, the input image size is decreased from 512 to 16. According to Fig.1, the CSP module maps features
into two sections for two convolution operations and then combines the results, decreasing the memory and increasing the detection accuracy. The Resnet allows a deeper network, and higher-level features can be extracted.

The appearance of marine organisms is irregular, and marine organisms’ color is often very close to the surroundings. The background and attenuation have great interference on marine organisms detection in the ocean environment. Therefore, there is a great significance to extract effective features to detect and recognize marine organisms. The neck architecture employs the SPP building block (purple) and the FPN+PAN building block in the network. In the SPP building block, the attribute size of max-pooling adopts respectively 5x5, 9x9, and 13x13, and the stride is set to be 1 to keep the image size unchanged.

In addition to the effect of traditional max-pooling, the SPP building block can extend the undertaking range of backbone network characteristics. That also improves the detection accuracy with lower calculation costs. In YOLOv4 Neck, FPN magnifies the size of the feature map through the up-sampling task to merge dimension and tensor with the feature map after the CSP task in the backbone network and transmit the object semantic message. After down-sampled the merged feature map through the convolution task, the PAN architecture is merged with the feature map of the relevant dimension in FPN to extract positioning features.

FPN+PAN merges different detection layers and trunk layers repeatedly and uses many kinds of dimensions to extract more deep positioning messages and semantic messages to find more slender objects of different sizes. As a consequence, the detection of tiny objects has been highly beneficial. For example, in marine organisms detection, the fish sizes of different breeds very extreme. When marine organisms of different sizes show in one image, the generality capability of the detection arithmetic is very significant, which will be interpreted in the discussion part. The head architecture in the network is the prediction section. The three-layer dimension feature maps (64 x64; 32 x 32; 16 x16) are output received from the CBL building block and convolution task at the tail end of YOLOv4-embedding Neck.

Each dimension predicts three anchor boxes, and there are five values per anchor (four box coordinates + one object confidence); the filter is equal to 4 class + 5 values and multiplied by the prior box that contained 3 anchors. Therefore, the prediction section has 27 outputs. In addition, the confidence and bounding box of the detected marine organisms can be acquired under the output message.

Then, the bounding box whose confidence is lower than the threshold will be deleted, and the best of all candidate boxes would be chosen under the DIOUT_nms arithmetic. Finally, CIoU loss is brought forward by forcing the conformity of the aspect ratio. The loss function can be defined as:

$$L_{CIoU} = 1 - IoU + \frac{\rho^2(b,b_d)}{c^2} + \alpha v$$  \hspace{1cm} (3)$$

Where \(\alpha\) represents the weight function, and \(v\) represents the similarity of the aspect ratio. After filtering by CIoU, the detection consequence is output, and the detection task is completed. Since CIoU takes the distance, overlap rate, scale, and penalty factors into account, the prediction frame can quickly approach the aspect ratio of
the real frame during training, avoiding divergence in the training process of the YOLOv3 version.

**Experimental Results**

This paragraph illustrates the consequence of marine organisms detection in the training period and detection period. In addition, the evaluation target, training parameters, and detection effects on different occasions are depicted.

**Experimental setup**

In marine organisms graphic classification tests, the tolerate hyper-parameters are listed: the training steps are 30938, the batch measurement and the mini-batch measurement are 8 and 1 separately; the polynomial decay learning ratio timetable outline is adopted with a first learning ratio of 0.0013. The iteration steps are 1000; the momentum and weight parameter are separately set as 0.949 and 0.0005.

**Assess training models**

In the training section, which assesses the recapitulation capability and progressively optimizes the model, precision, recall, and the mAP score were employed as an assessment target.

\[
\text{Precision} = \frac{TP}{TP + FP} \times 100\% \quad (4)
\]

\[
\text{Recall} = \frac{TP}{TP + FN} \times 100\% \quad (5)
\]

\[
mAP = \frac{(AP_1 + AP_2 + \ldots + AP_n)}{n} \quad (6)
\]

Where precision represents positive predictive value, TP represents real positive, FP represents fake positive, and FN represents fake negative.

In training, the batch size was subscribed to 8. That 8 images were fetched in each iteration, and overall, 1557 images were trained. Therefore, one epoch required 103 iterations. The weight consequence of each epoch was verified in the validation set. Because of a threshold, a set of precision and recall of the pattern could be received. When different thresholds are subscribed for the pattern, numerous groups of precision and recall would be received. The area of the Precision-Recall region is the average precision. Three groups of training stages epoch were subscribed to, with the largest 100, 200, and 300.
FIG 4 Evaluation index curve in mAP

<table>
<thead>
<tr>
<th>No.</th>
<th>I</th>
<th>II</th>
<th>III</th>
</tr>
</thead>
<tbody>
<tr>
<td>Epoch</td>
<td>100</td>
<td>200</td>
<td>300</td>
</tr>
<tr>
<td>Image size</td>
<td>512x512</td>
<td>512x512</td>
<td>512x512</td>
</tr>
<tr>
<td>Precision</td>
<td>0.87</td>
<td>0.86</td>
<td>0.86</td>
</tr>
<tr>
<td>Recall</td>
<td>0.92</td>
<td>0.95</td>
<td>0.95</td>
</tr>
<tr>
<td>mAP_{50}</td>
<td>0.9676</td>
<td>0.9717</td>
<td>0.9709</td>
</tr>
</tbody>
</table>

TABLE 2
Performance of the YOLOv4-embedding in mAP_{50}

<table>
<thead>
<tr>
<th>No.</th>
<th>I</th>
<th>II</th>
<th>III</th>
</tr>
</thead>
<tbody>
<tr>
<td>Epoch</td>
<td>100</td>
<td>200</td>
<td>300</td>
</tr>
<tr>
<td>Image size</td>
<td>512x512</td>
<td>512x512</td>
<td>512x512</td>
</tr>
<tr>
<td>Precision</td>
<td>0.78</td>
<td>0.85</td>
<td>0.86</td>
</tr>
<tr>
<td>Recall</td>
<td>0.97</td>
<td>0.95</td>
<td>0.95</td>
</tr>
<tr>
<td>mAP_{50}</td>
<td>0.9673</td>
<td>0.9675</td>
<td>0.9719</td>
</tr>
</tbody>
</table>

TABLE 3
Performance of the YOLOv4-embedding in mAP_{75}

<table>
<thead>
<tr>
<th>No.</th>
<th>I</th>
<th>II</th>
<th>III</th>
</tr>
</thead>
<tbody>
<tr>
<td>Epoch</td>
<td>100</td>
<td>200</td>
<td>300</td>
</tr>
<tr>
<td>Image size</td>
<td>512x512</td>
<td>512x512</td>
<td>512x512</td>
</tr>
<tr>
<td>Precision</td>
<td>0.63</td>
<td>0.67</td>
<td>0.73</td>
</tr>
<tr>
<td>Recall</td>
<td>0.66</td>
<td>0.74</td>
<td>0.81</td>
</tr>
<tr>
<td>mAP_{75}</td>
<td>0.625</td>
<td>0.7209</td>
<td>0.8268</td>
</tr>
</tbody>
</table>

TABLE 4
Performance of the YOLOv4-embedding in mAP_{75}

<table>
<thead>
<tr>
<th>No.</th>
<th>I</th>
<th>II</th>
<th>III</th>
</tr>
</thead>
<tbody>
<tr>
<td>Epoch</td>
<td>100</td>
<td>200</td>
<td>300</td>
</tr>
<tr>
<td>Image size</td>
<td>512x512</td>
<td>512x512</td>
<td>512x512</td>
</tr>
<tr>
<td>Precision</td>
<td>0.60</td>
<td>0.68</td>
<td>0.74</td>
</tr>
<tr>
<td>Recall</td>
<td>0.74</td>
<td>0.76</td>
<td>0.82</td>
</tr>
<tr>
<td>mAP_{75}</td>
<td>0.7085</td>
<td>0.7651</td>
<td>0.856</td>
</tr>
</tbody>
</table>

The weight conforming to the maximum mAP in each training was selected, the precision and recall were produced when the threshold was 0.5 and 0.75, compare the capability of the three epochs of training, the
performance of the YOLOv4 as shown in Table 1 and Table 3, and the performance of the YOLOv4-embedding in Table 2 and Table 4.

As the epoch and the number of iterations raised, the mAP turned higher and applied more time. At the set of 300 epochs in mAP, the highest mAP reached 82.68% in YOLOv4, whereas the mAP of 300 epochs in YOLOv4-embedding is enough to get a high mAP value of 85.6%. Therefore, the assess indexes in the training process of 300 epochs in YOLOv4-embedding are explained. In the epoch training, the numerical curves of mAP are shown in Fig. 4. It can be seen that the mAP figure can converge swiftly and near to the value of 0.9. The precision figure is more steady after the 150th epoch. Therefore, the optimal weight pattern in the epoch 300 training is chosen as the marine organisms detection model based on the YOLOv4-embedding arithmetic.

![Fig. 5(a)](image1)

![Fig. 5(b)](image2)
**Fig. 5(c)**

**FIG. 5.** Frame detection results close to marine organisms

*Detection consequence*

The trained marine organisms detection pattern was examined in different environmental conditions. Choose three cases in each condition. Fig. 5(a), Fig. 5(b), and Fig. 5(c) show the detection consequences of the camera close to marine organisms. Fig. 5(c) is closer to marine organisms than Fig. 5(b) and Fig. 5(a), there are more marine organisms be detected.

Fig. 5(a), Fig. 5(b), and Fig. 5(c) show that 8, 12, and 15 marine organisms were detected, respectively. Thus, marine organisms could be detected, no matter whether the marine organisms are completely irradiated or partially irradiated by the light source. If the light source is sufficient in the aquaculture ponds, the detection confidence of marine organisms is higher.
Fig. 6(a)

Fig. 6(b)

Fig. 6(c)

FIG. 6. Detection results far away from the light source condition.

Fig. 6 demonstrates the detection consequences far away from the light source condition. Fig. 6(a) is farther away from the light source than Fig. 6(b), Fig. 6(b) is farther away from the light source than Fig. 6(c). The orange line indicates the distance from the light source to the detection object. The object detection confidence in Fig. 6(a) is 60%, The object detection confidence in Fig. 6(b) is 98%, The object detection confidence in Fig. 6(c) is 100%. If the light source is insufficient, the detection confidence of marine organisms is lower. Otherwise, the light source is close to the detection object, the detection confidence of marine organisms is higher.
Fig. 7(a) large zone occlusion by water column; (orange box)

Fig. 7(b) less occlusion by water column; (orange box)

Fig. 7(c) no occlusion by water column; (orange box)

FIG. 7. Detection results of marine organisms under different occlusion conditions
Marine organism's detection consequences were different under different occlusion extent, which is also classified as an occlusion condition.

That is why examining the trained detection pattern in separately occlusion extent one by one, as shown in Fig. 7(a) is large zone occlusion (orange box), that did affect the detect result, because the information of the marine organism was too little due to occlusion; Fig. 7(b) shows the accurate detection when the occlusion zone decreased, but the pattern still detected the marine organisms’ confidence is 65%; The information of the marine organism in Fig. 7(c) was almost completely, the marine organism’s confidence is 91%. Occlusion in Fig. 7(a) and Fig. 7(b) often happens in successive detection. Accurate detection of all marine organisms in succeeding frames has great meaning for solving duplicated detection, while deep learning arithmetic has stronger environmental conditions.

![Fig. 8(a) The precision curve of the three deep learning architectures.](image)

![Fig. 8(b) The recall curve of the three deep learning architectures](image)
Discussion

Other deep learning arithmetic is compared in this division to examine the capability of the marine organism detection pattern in the aquaculture ponds. At the same time, marine organism detection consequences given deep learning arithmetic are compared and analyzed.

Comparison of YOLOv3 and YOLOv4-based in marine organism detection

The marine organism dataset in the YOLOv4-embedding neural network are trained and detected. The epoch was subscribed as 300, and the optimal training model was selected for validation, with mAP\textsubscript{50} of 97.19%. Fig. 8 shows the P-R curve of the three YOLOv4-based neural networks on the validation set. As referred above, YOLOv4 uses FPN+PAN architecture to collect the features of the trunk layer and detection layer again and again via multi-scales, which is of great meaning for improving small objects network detection, and YOLOv4-embedding changes the architecture of CBM to DEPCU (deep convolution), that consists of 3 building blocks(CBL+CBM+CBL).

Fig. 9(a) marine organism detected by YOLOv4
While YOLOv4 and YOLOv4-embedding apply the real-time Ocean environment as the background. Similarly, Haliotis is detected in Fig. 9. Both arithmetics have carried out precise detection on marine organisms, and YOLOv4-embedding gets higher confidence in the same frame. Compare to YOLOv3 and YOLOv4-embedding, YOLOv3 will misjudge the measurement of small objects, the human eye may have to make a distinction carefully to see the position of the marine organism. While YOLOv4-embedding made a precise detection. Because of FPN+PAN architecture and the use of tricks. Small marine organism object detection has great meaning to the operation management of aquaculture ponds. First, YOLOv4-embedding object detection can reasonably decide different varieties; Moreover, the accurate detection of the small marine organism can offer a beneficial message for successive detection.
Finally, the marine organism image at a different angle are caught. Though most marine organisms can be caught horizontally, and some marine organisms may not swim horizontally. Executing experiments on the marine organism images with an angle of 45 to see whether the marine organism can be detected. Fig.10 shows the detection consequences of YOLOv3 and YOLOv4-embedding in the 45 angles, it can be seen YOLOv4-embedding exactly detected the marine organisms, but YOLOv3 detected fewer numbers of marine organisms. This is because YOLOv4-embedding has more generalization capability in different angles of marine organisms detection.

**TABLE 5.**
Detection indexes of the four arithmetics.

<table>
<thead>
<tr>
<th>Arithmetic</th>
<th>EfficientDet-D3</th>
<th>YOLOv4</th>
<th>YOLOv4-embedding</th>
<th>YOLOv4-DEPCU</th>
</tr>
</thead>
<tbody>
<tr>
<td>mAP_{50}</td>
<td>50.0</td>
<td>71.2%</td>
<td>97.09%</td>
<td>97.19%</td>
</tr>
<tr>
<td>mAP_{75}</td>
<td>75.0</td>
<td>61.6%</td>
<td>82.68%</td>
<td>85.6%</td>
</tr>
<tr>
<td>Average</td>
<td></td>
<td>72ms</td>
<td>19.31ms</td>
<td>19.46ms</td>
</tr>
<tr>
<td>detection time</td>
<td></td>
<td></td>
<td></td>
<td>20.20ms</td>
</tr>
<tr>
<td>Weight size</td>
<td></td>
<td>46.33MB</td>
<td>244.22MB</td>
<td>244.55MB</td>
</tr>
<tr>
<td>Training time</td>
<td></td>
<td>5.47h</td>
<td>5.98 h</td>
<td>6.08 h</td>
</tr>
<tr>
<td>(300 epoch)</td>
<td></td>
<td></td>
<td></td>
<td>8.40 h</td>
</tr>
</tbody>
</table>
Comparison between different deep learning arithmetics

Machine learning arithmetic has a lower running expense, shorter training time, and smaller weight measurement. That can be achieved on the CPU, does not demand GPU, but longer detection time and lower detection ratio. So marine organism targets in deep learning arithmetics are compared.

Table 3 compares the crux target of the four arithmetic in the same test set. YOLOv4-series require GPU to obtain the optimal model, which required 300 epochs for training. The training time of YOLOv4
was shorter than YOLOv4-embedding, and the weight measurement was smaller than YOLOv4-embedding. YOLOv4 had the shortest detection time for a single image, but mAP$\text{75}$ was not as high as YOLOv4-embedding.

The mAP$\text{75}$ of YOLOv4 was 82.68%, which was smaller than YOLOv4-embedding arithmetic and YOLOv4-DEPCU arithmetic, YOLOv4-embedding detected the marine organism with mAP$\text{75}$ 85.6%, YOLOv4-DEPCU detected the marine organism with a detection ratio of 84.75%. The average detection time of YOLOv4 was 19.31ms, and the YOLOv4-DEPCU detection time was 20.20ms. Since the network of YOLOv4-DEPCU is deeper than YOLOv4, the detection time was also increased. On the whole, YOLOv4-series could obtain the optimal weight model with iterations in the training level, and superior to the traditional machine learning arithmetic and deep learning arithmetic like EfficientDet-D3, YOLOv4-series have high confidence and high detection ratio in the detection level.

Conclusion

Accurate marine organisms detection is a major meaning to the clever administration of the aquaculture ponds. This paper proposed a detection modus given the latest YOLOv4 neural network for marine organism detection in natural conditions. Besides, the performance of the EfficientDet-D3 arithmetic and YOLOv4-based arithmetic in marine organism detection are analyzed. According to the experimental results, the following conclusions can be summarized:

(1) Deep learning arithmetic for marine organism detection in the aquaculture ponds is suitable. The structural features of the YOLOv4-embedding neural network and the crux issues of aquaculture ponds detection were analyzed. In the network, CSPDarknet53 deepens the network that could extract more deep marine organism features and reduce the interference of background;

The SPP architecture increases the acceptance range of network characteristics with less computational expense. In addition, FPN+PAN architecture merges multi-scale features to extract more profound marine organisms’ semantic information and positioning information, which detect marine organisms more accurately. However, accurate detection can still be achieved when the marine organism sizes in the same image are highly different; The CIOU arithmetic improves the confidence of marine organism detection results, and YOLOv4-embedding arithmetic achieves 60.8% AP$\text{50}$ for the MS COCO dataset in Fig. 11.

(2) The marine organism detection arithmetic in the aquaculture ponds, based on the YOLOv4-embedding neural network, can detect precisely when under different occlusion and illumination for different breeds and matureness, information for the marine organism intelligent management and underwater machine.

(3) YOLOv4-embedding neural networks are constructed to realize the fast differentiation of marine organisms. The intermediate data of CNN and image feature extraction by deep learning when the network training is stable are
studied. The influence of convolution embedded strategy, sample number, and training rounds on network training speed and recognition accuracy are also discussed.

As shown from the effect of experiments, different marine organisms can be recognized and classified effectively by using improved CSPNet[10] architecture into the neck architecture of YOLOv4. Furthermore, optimizing the gradient backpropagation path improves the network’s learning ability and greatly reduces the amount of calculation under the premise of ensuring accuracy.

The detection performance of YOLOv4-based arithmetic is better than EfficientDet-D3 arithmetic for marine organism detection in the aquaculture ponds. Compared with YOLOv4, YOLOv4-embedding, and YOLOv4-DEPCU, the average detection time of the three arithmetic was 19.31ms, 19.46ms and 20.20ms. The mAP<sub>75</sub> of marine organisms was 82.68%, 85.6%, and 84.75%, respectively. In the training section, YOLOv4-embedding obtained the optimal weight model with 300 epochs iterations. Moreover, higher FPS is still the characteristic of YOLOv4-embedding.

In the detection section, YOLOv4-embedding was superior to other arithmetic with its high confidence and high mAP. In conclusion, the proposed architecture is suitable for marine organism detection in aquaculture ponds. Future work will mainly obtain the coordinate value of marine organisms in the real world, achieve the localization of marine organisms, calculate the picking dot’s location, deploy a marine organism detection model in tiny terminals, develop a moving mechanically in the research. Increase the identification ability of marine biological species characteristics in the aquaculture pond and reduce background noise interference. In addition, this algorithm can cope with the problem of reduced marine biological recognition ability due to different depths and different light intensities. After increasing training data and training times, it can effectively improve the accuracy of recognition. Thus, this research provides a new solution to the problem of species identification in aquaculture ponds and will more accurately grasp the status of individual fish bodies and species in the aquaculture ponds to quickly discover and solve the problems in the aquaculture ponds and make specific contributions to reducing the risks of the aquaculture ponds.

**Authorship contributions:**

*Latest deep learning arithmetic does higher domain-general strategy knowledge
*Performance of the mean average precision during the detection of marine organisms
*Monitoring in real-time keeps the performance of the strategy up during marine organisms detection
*Performance of a strategy after improving the neural architecture domain-general strategy knowledge
*Improving and monitoring by arithmetic combined foster domain-general strategy knowledge

I. Ethical approval:

All applicable international, national, and/or institutional guidelines for the care and use of animals were followed.

II. Funding details:
Funding: This study was funded by the National Natural Science Foundation of China (grant number 61773415).

III. Conflict of interest:

Author Jinde Zhu declares that he has no conflict of interest.

IV. Informed Consent:

Informed consent was obtained from all individual participants included in the study.

References


