Hindawi
Complexity
Volume 2021, Article ID 4761670, 8 pages
https://doi.org/10.1155/2021/4761670

Research Article

High-Accuracy Real-Time Fish Detection Based on Self-Build Dataset and RIRD-YOLOv3

Wenkai Wang, Bingwei He, and Liwei Zhang

School of Mechanical Engineering and Automation, Fuzhou University, Fuzhou 350000, China

Correspondence should be addressed to Bingwei He; mebwhe@fzu.edu.cn

Received 19 July 2020; Revised 5 January 2021; Accepted 4 March 2021; Published 8 April 2021

Academic Editor: Xin Dong

To better detect fish in an aquaculture environment, a high-accuracy real-time detection model is proposed. An experimental dataset was collected for fish detection in laboratory aquaculture environments using remotely operated vehicles. To overcome the inaccuracy of the You Only Look Once v3 (YOLOv3) algorithm in underwater farming environment, a suitable set of hyperparameters was obtained through multiple sets of experiments. Then, a real-time image recovery algorithm is applied before YOLOv3 to reduce the effects of both noise and light on images whilst keeping the real-time capability, leading to a mean average precision of 0.85 and frame rate of 17.6 fps, respectively. Finally, compared with the base detection model using only the YOLOv3 algorithm, the enhanced detection model presented results in a reduction of miss detection rate from 23% to only 9% across different environments and with the detection accuracy of the target in different environments being improved from 8% to 37%.

1. Introduction

Recently, ocean engineering and research have increasingly relied on underwater images captured by autonomous underwater vehicles (AUVs) and remotely operated vehicles (ROVs) [1]. However, since the collection of underwater datasets is more difficult than that for onshore datasets, there are few generally accessible datasets for underwater creatures, and public datasets for freshwater creatures are even rarer. In addition, underwater images usually suffer from various types of degeneration, such as low contrast, color casts, and noise, due to wavelength-dependent light absorption and scattering as well as the effects of low-end optical imaging devices [2]. To obtain much higher quality underwater images, a number of advanced methods have been designed and used. For example, Gray World [3] and White Patch [4] are used in color correction. Fang et al. proposed a single image enhancement approach based on image fusion strategy to enhance the underwater image [5]. Li et al. presented a systematic underwater image enhancement method including underwater image dehazing algorithms and a contrast enhancement algorithm for high-quality underwater images [6], and Hitam et al. utilized the contrast limit adaptive histogram equalization (CLAHE) to enhance the contrast [7]. Recently, Peng and Cosman proposed a depth and background light estimation method for underwater scenes based on image blurriness and light absorption, which can be used to restore and enhance underwater images [8]. Besides, many studies try to address the issue from the physical level. Typically, Schechner and Karpel employed a polarizer in front of their camera [9]. These methods work well for underwater image processing, but few of them took the degeneration model into account or the proposed models are too complex to work in real-time. Moreover, most existing algorithms are lacking in the capability of self-adaption and self-adjustment, which are important for a robot working in a changing and complex underwater environment.

Instead of traditional target detection, artificial neural networks (ANNs) can be used to detect fish in images, and some methods have shown promise for real-time performance, such as Faster R-CNN [10], R-FCN [11], SSD [12], and YOLO series [13–15], amongst others. Among them, YOLOv3 performs well in both real-time and in terms of mean average precision (mAP). However, YOLOv3 just performs well in clear waters. When in dim and turbid

waters, YOLOv3 loses almost all its original land-based advantages.

In this paper, an experimental dataset was collected, which solves the problem of the lack of datasets for fish in aquaculture environments. Furthermore, a set of suitable hyperparameters were obtained for the dataset through multiple sets of experiments, reducing training time and improving the detection accuracy. To improve the performance of YOLOv3, a high-accuracy real-time fish detection algorithm was proposed, named RIRD-YOLOv3. This paper is organized as follows: firstly, this paper introduces the laboratory acquisition of the dataset, the RIRD-YOLOv3 algorithm, and the matching of hyperparameters; secondly, we discuss the analytical results of the experiments; finally, we present the conclusion. In this paper, the proposed algorithm is tested, and it performs well.

2. Dataset Collection and RIRD-YOLOv3 Algorithm

2.1. Acquiring the Dataset. Deep learning [16] requires a large amount of training samples, and the amount of data used will directly affect the detection accuracy of fish for this application. However, the problem faced by the fish dataset is that its open source dataset is very scarce and does not meet the training needs of grass carp detection models.

To solve the problem of the lack of grass carp dataset in the breeding environment, in this paper, through a field investigation, a simulated grass carp breeding environment is established in the laboratory. Based on the growth environment of grass carp, the length, width, and height of the pool are set to 600 cm, 450 cm, and 250 cm. The pond can simulate the real grass carp breeding environment. The experiment site is shown in Figure 1.

The content of the sample of the dataset includes adult grass carp and robot fish. Among them, the robotic fish is a bionic robot purchased in the laboratory. It matches the shape and characteristics of the real fish. The purpose of placing it in the dataset is to verify whether the classification performance of the model obtained by the algorithm will be affected. Each of the sample image contains zero or more instances of fish. As a result, each image could contain from zero to multiple annotations. This method enriches the types of dataset samples and can be used to verify the classification characteristics of the detection model. The sample content of the dataset is shown in Figure 2.

To fully simulate the impact of light on grass carp in the breeding environment and to fully collect images of grass carp under different light environments, this paper accomplishes this by changing the lighting conditions at different time periods. The specific implementation is shown in Table 1.

According to Table 1, after using ROV to collect grass carp images, through sorting out, it is found that three types of images are useless, as shown in Figure 3. The three types of images shown in Figure 3 do not contain much in the dataset, but they still affect the accuracy of the subsequent detection model. Therefore, in order to achieve the purpose of improving the accuracy of the detection model, this paper removes these three types of images manually.

Figure 1: The experiment site.

A standard dataset should include a training set and a testing set. The training set and the testing set are mutually exclusive. The training set is used to obtain an excellent detection model, and the testing set is used to test the performance of the model. After removing the above three types of useless samples, the dataset contains 3069 images. In order to ensure the performance of the detection model, this paper uses the 'reserve method' to randomly divide the images of the dataset into a training set and a testing set according to 7 : 3 ratio. After the division of "reserve method," the training set contains 2148 images and the testing set 921 images. Besides, in this paper, the training set is classified according to the three types of lighting conditions shown in Table 1, and the number of the three types of samples is NL 443, NOL 1228, and 477 NORL. The dataset named the Grass Carp Dataset before Restoration (GCDBR) is composed of original images. The examples of GCDBR are shown in Figure 4.

Besides, to improve the detection accuracy, it is necessary to separate out the fish from the environment. Usually fish have a similar color to the environment to protect themselves. Therefore, a large number of optical images of the underwater environment were collected and labelled as 'negative sample.'

2.2. Dataset Labelling. The dataset needs to be annotated to accomplish and validate the goal of classification and detection on the images. In this paper, labelling software is used to create annotations of respective classes for the images in the dataset based on the PASCAL VOC [17] standard labelling format. Each annotation is created by drawing a bounding box around the object of interest belonging to one of the classes and assigning the bounding box and the class label associated with it. For simplicity, in this paper, axis-aligned bounding boxes are used as described in the PASCAL VOC dataset paper [17]. The examples of annotated images are shown in Figure 5.

2.3. The RIRD-YOLOv3 Algorithm. The water quality of the grass carp farming environment is turbid, and due to the absorption of light by the water, the scattering effect, and

(a) (b)

Figure 2: The sample content of the dataset: (a) robot fish; (b) grass carp.

Table 1: Different light conditions and implementation methods of grass carp growth.

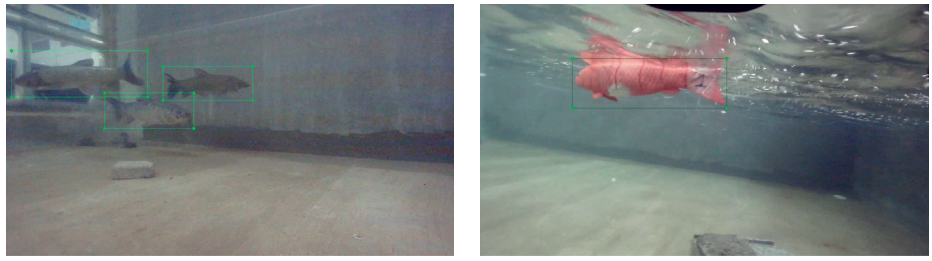
Lighting conditions Reasons and method of implementation	Natural light (NL)	Natural and outdoor light (NOL)	Natural, outdoor, and ROV light (NORL)
Reason	Sufficient sunlight, ROV without ROV light can clearly collect grass carp images	With sunlight but weak, such as cloudy, rainy, and evening	Late at night or without any light
Method of implementation	Just having sunlight, without any additional auxiliary light source	Sunlight is the main light source; outdoor light source is the supplement	Outdoor light and ROV light are main light sources

(a) (b) (c)

Figure 3: Cull images: (a) sports afterimage; (b) no target; (c) too many bubbles.

(a) (b) (c)

Figure 4: The examples of GCDBR: (a) NL; (b) NOL; (c) NORL.



(a) (b)

Figure 5: Example images within the labelled dataset: (a) grass carp; (b) robot fish.

the uneven illumination of the ROV, the quality of the image will deteriorate and the grass carp cannot be distinguished by the naked eyes. The examples of images of grass carp sample of low quality are shown in Figure 6.

To overcome this problem, Chen et al. provided three parameters, related to underwater image degradation and color correction, by presearching in the first frame of image sequences using an artificial fish school algorithm [18]. The core of image restoration is a Wiener Filter in frequency domain as follows:

$$V_{\text{orig},C}(u,v) = \left[\frac{H(u,v)}{H(u,v)^2 + R}\right] V_{\text{deg},C}(u,v), \tag{1}$$

where $V_{\text{orig},C}$ represents one channel of the original image; $V_{\text{deg},C}$ represents one channel of the degraded image due to underwater scattering and abortion; R is the reciprocal of signal-to-noise ratio and was implemented to restrict scattering; $H(u,v)$ is originated as a general image degradation model in turbulent media [18] expressed by

$$H(u,v) = e^{-k\left(u^2+v^2\right)^{(5/6)}}, \tag{2}$$

where k is a crucial parameter related to the depth of water and the distance from the camera.

After Wiener Filter is applied, color correction is implemented on the image by gamma factor as follows:

$$I_{\text{corrected},C} = I^{\gamma}. \tag{3}$$

At this point, R, K, and γ have been introduced. To obtain a reliable combination of these three parameters, we employ a quality index of the restored image expressed as follows:

$$Q = \frac{\alpha\beta}{1+\eta}, \tag{4}$$

where α is a haze indicator, describing the level of haze by gradient computed by the modified Tenegrad evaluation, given as follows:

$$\alpha = \frac{1}{W}\sum_{i=0}^{M}\sum_{j=0}^{N}\sum_{k=0}^{7}\left|\text{Grandient}\left(V_g(i,j),k\right)\right|^2, \tag{5}$$

where $M \times N$ is the size of an input image; V_g is a grayscale map, and orientations of gradient are regulated as $k \times 45^{\circ}$. This indicator takes the textural feature and edge feature into consideration. Generally, a higher value of α reflects a clearer restored image.

β is a contrast indicator, which is calculated by histogram distribution in RGB channels, representing the image contrast as defined in the following equation:

$$\beta = \frac{1}{MN}\sum_{C\in\{R,G,B\}}\sqrt{\sum_{i=0}^{255}\left(h_C(i)\times i - \mu_C\right)^2}, \tag{6}$$

where $h_C(i)$ stands for the data of histogram curves at gray level i for channel C and μ_C shows the average of histogram curves of channel C. Theoretically, objects can be distinguished more easily with a higher value of β.

η is an imbalance indicator, which denotes the level of color correction as follows:

$$\eta = \left|\mu_r - \mu_b\right| + \left|\mu_r - \mu_g\right| + \left|\mu_b - \mu_g\right|. \tag{7}$$

Clearly, η diminishes along with a better color correction.

A test result of a deep sea image is shown in Figure 7. Clearly, the method is effective in contrast and color correction, and it takes only 17.5 milliseconds for each frame on average. In addition, the amount of relevant information in the restored image has been retained to a large degree, such as color information, texture and edge information, and illumination information.

2.4. Matching of Hyperparameters. To train the ANN efficiently and well to predict the desired outcome, the hyperparameters of the network should be properly determined. For the various values of number of epochs, momentum, learning rate, and batch size, a grid search was performed to optimize the hyperparameters. All possible sets of values described in Table 2 were tested to train the network. Then, after training the network using each set of values, the sensitivity was assessed using 100 test images. Then, the values that maximize the quality of the network were adopted for the hyperparameters.

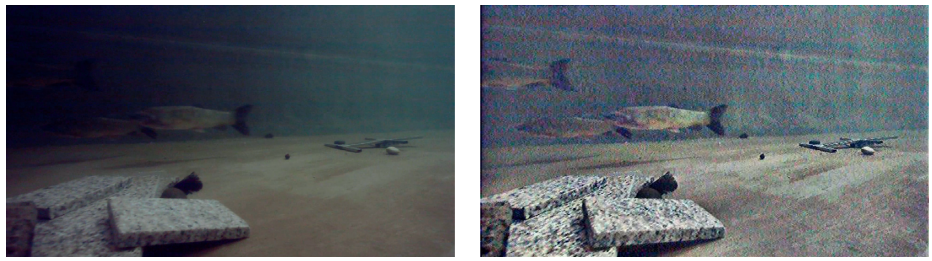
3. Results and Discussion

3.1. Experimental Platform and GCDAR. First of all, the experiment in this paper starts with image restoration. The specific implementation method is to restore the sample image of GCDBR to obtain the grass carp dataset after restoration (GCDAR). The comparison of sample images of grass carp dataset before and after restoration is shown in Figure 8. The appearance of the target in the image is different due to the influence of the light. In Figure 8(a), since the color of the target is similar to the background environment, it becomes difficult for the human eye to detect the target. In Figure 8(b), under the action of the outdoor auxiliary light, the underwater image halo is enhanced, and the target is very blurred due to the presence of water mist. In Figure 8(c), when natural light and ambient light do not work, the self-contained light source of the ROV is used, but the observation of the underwater target is still difficult due to the limited light strength. After the image is restored, the target in the restored image in Figure 8 becomes distinctly clear.

When training the model, our device is NVIDIA Tesla M40 with graphics card of 12 GB. When performing image recovery and detection, the actual ranges of K, R, and γ are $[10^{-7}, 1.5\times10^{-4}]$, $[0.01, 15]$, and $[0.4, 1]$ all of which are normalized as $[0.1, 150]$. The time-related data are obtained with $[640\times360]$ pixel-size images, and the processor was a Core i5-7300HQ CPU with the main frequency increased up to 2.5 GHz.

(a) (b)

Figure 6: Examples of image of grass carp sample of low quality.



(a) (b)

Figure 7: The performance of image restoration: (a) before restoration; (b) after restoration.

3.2. Training Result. The best hyperparameters for training are shown in Table 3. Figure 9 is a graph showing the relationship between batches and average loss. For each batch, 64 images are randomly selected and used to train the ANN. Since the number of samples is limited, each image is used multiple times. The graph shows that the average loss is almost reduced to 0 as batches progress. A total of 30200 epochs were run, and it took 48 hours to complete the training. Compared with the initial training parameters, the best hyperparameters reduce the training time by at least 48 hours.

To meet real-time requirements, through testing, the frame rate of GCDAR's model is shown in Table 4.

In addition, evaluation of the trained network is performed by taking our validation dataset consisting of 300 images and executing detection on it using the trained model.

The metrics used to evaluate the object detection are as follows:

(a) *mAP*. This is the mean of the interpolated average precision across all the classes in the dataset used for object detection.

(b) *IOU*. This is the ratio of the area of intersection to the area of union of the predicted bounding box and the corresponding maximally matched ground truth box as defined in the following equation:

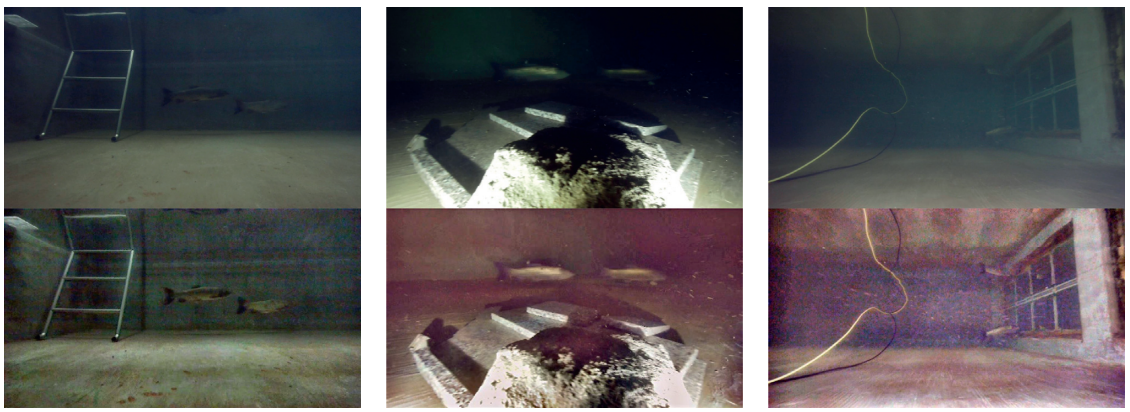
$$\text{IOU} = \frac{\text{Area}\left(\text{PredictedBox} \cap \text{GroundtruthBox}\right)}{\text{Area}\left(\text{PredictedBox} \cup \text{GroundtruthBox}\right)}. \quad (8)$$

Table 2: Tested values in grid search of hyperparameters.

Number of epochs	1000, 2000, . . ., 40000
Momentum	0.6, 0.7, 0.8, 0.9
Learning rate	0.00001, 0.00002, 0.00003, 0.00004
Batch size	8, 16, 32, 64

3.3. Comparison of Unrecovered and Recovered Images. To verify the validity of RIRD-YOLOv3 in different environments, two new evaluation parameters were proposed, namely, missed detection rate (MDR) and target detection accuracy (TDA) in a single environment. In addition, to demonstrate the advantages of the method, a comparative experiment of image detection capability before and after restoration is proposed and conducted.

The MDR and the TDA were also tested (as defined in equations (9) and (10)). The results are shown in Table 5. As seen from Table 5, the restored image's MDR is reduced from 8.9% to 21.7% compared with the image before restoration. The restored image's TDA is increased from 7.8% to 36.8% compared with the image before restoration. The large reduction in the rate of missed detection indicates the effectiveness of the RIRD-YOLOv3 algorithm. The improvement of detection accuracy in different environments shows that the model has generally excellent performance. Figure 10 shows the IOU contrast between the prerecovery image and the restored image. In the original image, the target detection showed missed detection and false detection. However, in the restored image, both missed detection



(a) (b) (c)

Figure 8: Example of the comparison of sample images of grass carp dataset before and after restoration: (a) NL; (b) NOL; (c) NORL.

Table 3: Hyperparameters to train.

Parameter	Number of epochs	Momentum	Batch size	Learning rate
Numerical value	30200	0.9	64	0.00001

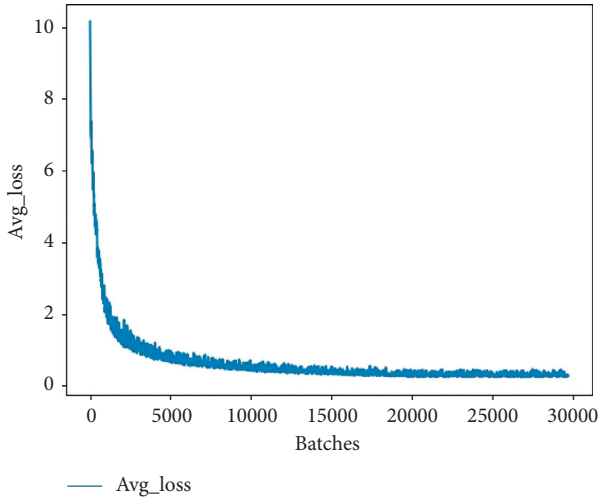


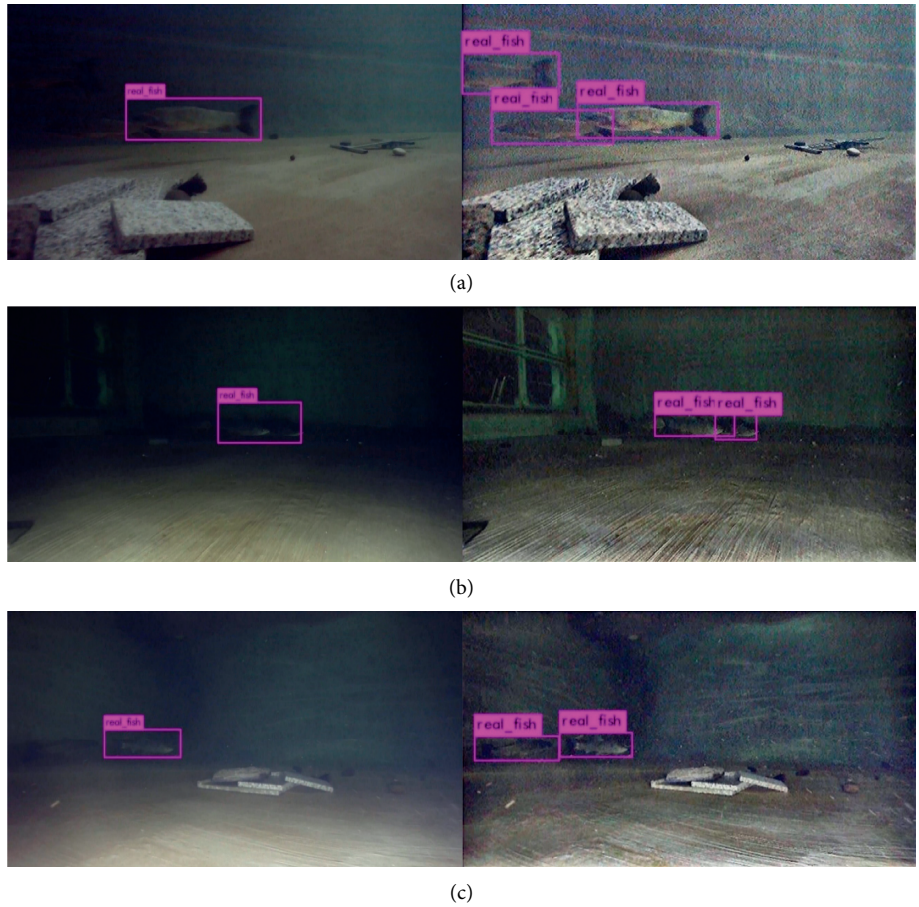
Figure 9: The average loss function of training.

Table 4: Performance of the test model based on the RIRD-YOLOv3 algorithm.

Metric	Frame rate	mAP	IOU
Numerical value	17.6	0.85	0.82

Table 5: Comparison table of missed detection rate and target detection accuracy in three environments.

	Miss detection rate		Detection accuracy	
	Before recovery (%)	After recovery (%)	Before recovery (%)	After recovery (%)
NL	12.1	3.2	89.3	97.1
NOL	26.4	12.5	62.9	99.71
NORL	33.3	11.6	73.1	86.7



(a)

(b)

(c)

Figure 10: Comparison of fish detection between original image and restored image in three environments: (a) NL; (b) NOL; (c) NORL.

and false detection are reduced.

$$\mathrm{MDR} = \frac{M}{M+N} \times 100\%, \tag{9}$$

where M is the missing detection image and $N$is the no missing detection image.

$$\mathrm{TDA} = \frac{t_1 + t_2 + t_3 +, \cdots, +t_n}{n} \times 100\%, \tag{10}$$

where $t$is the detection accuracy of a single target in test images and n is the number of targets.

4. Conclusions

In this paper, we presented a high-accuracy real-time fish detection algorithm, called RIRD-YOLOv3. It is able to solve the problem of image blur and noise caused by processing in an underwater environment. In addition, a set of suitable hyperparameters is provided for laboratory freshwater aquaculture environmental dataset. When using the hyperparameters is discovered, the experimental results show that the training time for the dataset is reduced by 48 hours. During testing, the frame rate of RIRD-YOLOv3 was 17.6 FPS and the model's mAP is 0.85. Prerecovery and postrecovery images were contrasted in three environments, and the miss detection rate and detection accuracy are reduced from 23% to 9% and increased from 13% to 37%, respectively. Therefore, overall RIRD-YOLOv3 has demonstrated excellent performance for this type of environment.

The RIRD-YOLOv3 algorithm is of important significance for underwater target detection applications. It can be applied to underwater submersibles such as ROV and AUV. It has potential for further contribution to the exploration of underwater resources.

Data Availability

This paper proposes the laboratory acquisition of the dataset, which is published on CSDN, and data can be obtained from the following links: https://download.csdn.net/download/qie123zi456/12328847, https://download.csdn.net/download/qie123zi456/12328855, https://download.csdn.net/download/qie123zi456/12328879, https://download.csdn.net/download/qie123zi456/12328894, and https://download.csdn.net/download/qie123zi456/12328901, https://download.csdn.net/download/qie123zi456/12328911.

Conflicts of Interest

The authors declare that they have no conflicts of interest regarding the publication of this paper.

References

[1] M. Bryson, M. Johnson-Roberson, O. Pizarro, and S. B. Williams, "True color correction of autonomous underwater vehicle imagery," *Journal of Field Robotics*, vol. 33, no. 6, pp. 853–874, 2016.

[2] C. Li, J. Guo, and C. Guo, "Emerging from water: underwater image color correction based on weakly supervised color transfer," *IEEE Signal Processing Letters*, vol. 25, no. 3, pp. 323–327, 2018.

[3] G. Buchsbaum, "A spatial processor model for object colour perception," *Journal of the Franklin Institute*, vol. 310, no. 1, pp. 1–26, 1980.

[4] E. Provenzi, C. Gatta, M. Fierro, and A. Rizzi, "A spatially variant white-patch and gray-world method for color image enhancement driven by local contrast," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, no. 10, pp. 1757–1770, 2008.

[5] S. Fang, R. Deng, Y. Cao, and C. Fang, "Effective single underwater image enhancement by fusion," *Journal of Computers*, vol. 8, no. 4, pp. 904–911, 2013.

[6] C.-Y. Li, J.-C. Guo, R.-M. Cong, Y.-W. Pang, and B. Wang, "Underwater image enhancement by dehazing with minimum information loss and histogram distribution prior," *IEEE Transactions on Image Processing*, vol. 25, no. 12, pp. 5664–5677, 2016.

[7] M. S. Hitam, E. A. Awalludin, W. N. J. H. W. Yussof, and Z. Bachok, "Mixture Contrast Limited Adaptive Histogram Equalization for Underwater Image Enhancement," in *Proceedings of the 2013 International Conference on Computer Applications Technology (ICCAT)*, pp. 1–5, Sousse, Tunisia, January 2013.

[8] Y.-T. Peng and P. C. Cosman, "Underwater image restoration based on image blurriness and light absorption," *IEEE Transactions on Image Processing*, vol. 26, no. 4, pp. 1579–1594, 2017.

[9] Y. Y. Schechner and N. Karpel, "Clear underwater vision," in *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Washington, DC, USA, June 2004.

[10] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: towards real-time object detection with region proposal networks," *Advances in Neural Information Processing Systems*, vol. 39, pp. 91–99, 2015.

[11] J. Dai, Y. Li, K. He, and J. Sun, "R-FCN: object detection via region-based fully convolutional networks," *Advances in Neural Information Processing Systems*, vol. 29, pp. 379–387, 2016.

[12] W. Liu, D. Anguelov, D. Erhan et al., "SSD.: single shot MultiBox detector," in *Proceedings of the European Conference on Computer Vision*, pp. 21–37, Cham, Switzerland, October 2016.

[13] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: unified, real-time object detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 779–788, Las Vegas, NV, USA, June 2016.

[14] J. Redmon and A. Farhadi, "YOLO9000: better, faster, stronger," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 7263–7271, Honolulu, HI, USA, July 2017.

[15] J. Redmon and A. Farhadi, "YOLOv3: an incremental improvement," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Salt Lake City, UT, USA, June 2018.

[16] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[17] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The PASCAL visual object classes (VOC) challenge," *International Journal of Computer Vision*, vol. 88, no. 2, pp. 303–338, 2010.

[18] X. Chen, Z. Wu, J. Yu, and L. Wen, "A real-time and unsupervised advancement scheme for underwater machine vision," in *Proceedings of the 7th Annual IEEE International Conference on Cyber Technology in Automation, Control, and Intelligent Systems (CYBER)*, pp. 271–276, Hawaii, USA, July 2017.