Fast and Accurate Fish Detection Design with Deep Learning YOLO-v3 Model and Transfer Learning

Kazim Raza and Zhoushan,

Zhejiang, China

Abstract

Object Detection is one of the difficult computer vision problems with countless applications. We proposed a real-time object detection algorithms based on YOLOv3 for detecting low accuracy objects and slow speed of detection. The demand for monitoring the marine ecosystem is increasing day by day for a vigorous automated system, which could be beneficial for all of the researchers in order to collect information about marine life. This proposed work mainly approached the computer vision technique to detect and classify marine life. Most such systems are already developed that totally based on CNNs where a large amount of training data required. In this paper, we performed object detection on four fish species custom datasets by applying YOLOv3 architecture, where we got 87.56% mAP (mean average precision). We also worked on improving the YOLOv3 baseline model with the help of a novel transfer learning technique, and improvement in loss function to improve the model performance. Moreover, comparing to the experimental analysis of the original YOLOv3 model with the improved one, we observed the mAP increased from 87.56% to 91.30. It showed that improved version outperforms than the original YOLOv3 model.

Keywords:

Deep learning, computer vision, Transfer Learning, Improved YOLOv3, Anchor Box, Custom dataset

1. Introduction

Deep learning is a subfield of Machine learning in Artificial Intelligence, which is based on artificial neural networks that can be unsupervised, semi-supervised, or supervised learning. The methods of deep learning are characterization learning methods which acquired from nonlinear modules that are used to transform raw data representation into a higher level. The core aspect of deep learning is that layers acquired from the given data, unlike humans [1]. Researchers tried hard to train a deep multi-layer network for decades. Still, before 2006, there were not many successful experiments at that time where they passed on effective results with one or two hidden layers. Those results were not producing proper effect due to exploding gradients. Deep learning is like a sensory system where the flow of information is deeply interconnected with all of the neurons. Every neuron helps to process the information to the next one.

Manuscript received January 5, 2026
Manuscript revised January 20, 2026
https://doi.org/10.22937/IJCSNS.2026.26.1.6

There is a massive difference between deep learning and machine learning that machine learning only relies on structured data, whereas deep learning required layers of the Artificial Neural networks. Szeged at el [2] Deformable part based model (DPM) also one of the top techniques for object recognition. Its implementation is established on the decomposition of the object and expressed in graphical mode. DPM has only two layers so this architecture is also not a useful technique for a big dataset. Traditional machine classifiers like SVM, LDA which is insufficient for huge dataset classification. The hierarchical Classification is a little better than SVM because it is 4% better accuracy results than that of a flat SVM classifier [3]. In Previous traditional methods, the researches not used the deep CNN design furthermore as they had been used a tiny dataset that has a low range of images and a restricted number of fish species Also, they use hand crafted ways that the performance was not worthy. The algorithm that had implemented was inadequate for a big dataset and resultantly the popularity accuracy not achieved consequently. In the recent past, the R-CNN, fast R-CNN and faster R-CNN gain significant research performance but these architectures have a very complex execution pipeline to perform recognition tasks. These architectures have less FPS (frame per second) and accuracy as well.

2. Background study

Machine learning which learn from the training data and produced the output. It has two types of supervised learning and unsupervised learning. Supervised learning set out the class of problems using a model. This model is used to learn a mapping between the target variable and the input data in order to make predictions. Unsupervised learning deals with unlabeled data where you do not need to supervise the model - let the model work on its own. This learning is mainly used to find out the hidden patterns from the data or extract the relationships in data only. In the deep ocean, the movement of fish is unpredictably quick and three dimensionally ; therefore, recognition is a difficult task. Fish recognition depicts to identify different types of fish species according to their features. It is essential to

locate for other kinds of reasons, including contour and pattern matching, statistical, quality control, feature extraction, and determination of physical traits [4]. Larsen at el. [5] obtained the shape and texture feature from appearance model and testing on the dataset, which has been containing more than 100 images of three fish species and attaining the accuracy 76%. Helge Balk at el. [6] developed the Sonar5 post-processing program that covered interpretation, analysis, and acquisition stages of hydro acoustic fish detection. The fish-echoes, along with surrounding noise level, can be detected using this program due to its time variation in sonar's detection, so the overall accuracy was high. Fuming Xiang at el. [7] used CNN models pipeline, including VGG16 and SSD on 9 common species of fish in the Missouri river to classify into category and position. They have achieved 87.22% accuracy in the classification of the fish.

Recognizing fish accurately is one of the possibilities that come out with deep learning, which helps you to find the targeted underwater species, i.e., fish. There are hundreds of applications to recognize marine fish, and many practices have already been done to find the right one object, which helps people to solve the problem. Tracking and counting the fish is also crucial for fish industry and conservation purposes as well. Traditionally, marine biologists square up the underwater situation by human underwater observations or by casting nets or throw the light to save up such methods are not automated.

The exact quantity of slaughtering fish is not finalized yet. Still, there is an estimated figure that salmon, sea trout, and migratory char are 27.0% decreased in killing fish from 2017 to 2018, according to Statistics Norway [8]. As per the report, the global river catch has passed to almost 10 million right after the linear growth from the 1950s, which was underreported on collecting the relevant data in the past [9]. There is no certainty on how much river fish caught, released, or slaughtered after catching from the river or ocean, so this thing needs to be automated with an accuracy of data.

Moreover, the caught fish is healthy or not needs some consideration and observation to determine whether the fish is healthy as not all fishes can be healthy. For all of such problems, the CNN does help in the classification of the marine system, observing the behavior of the underwater object, tracking an accurate object, automated, accurate counting of fish caught globally, localization, and controlling the environment.

There almost 20 deep neural networks have been trained for Salmon fish recognition that provides an in-depth discussion of each model with parameter tuning [10]. And, SSD version 2 achieved 84.64% Map, state-of-the-art accuracy with 3.75 FPS for salmon recognition. Background subtraction method was used to detect and track fish in marine life with the help of a video sequence. They get an accurate 73% result from the real type of video though they get the best result by implementing the Viola-Jones method using Haar cascade [11].

Undoubtedly, fish recognition is a complex task where some of the challenges like noise, distortion, overlap, occlusion, and segmentation error needs several techniques to get some accuracy in the result. Some of the techniques have already been applied and one of the Support Vector Machine (SVM) based techniques used on the two training sets on the fish features [12]. One was contained 74 fish testing set, and the other was about 76 fish. The final result based on SVM showed 78.59% accuracy in the fish classification. Dhruv Rathi et al. [13] derived a method based on CNN for the automation classification of fish species, which achieved 96.29% accuracy than other proposed systems.

3. Research Method

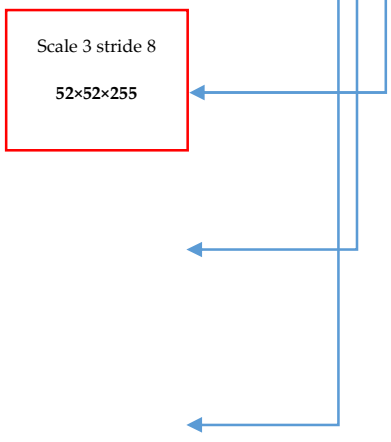
Object recognition and detection are some of the important issues in computer vision problems. On the bases of detection pipeline and backbone architecture, the object detector algorithm classified into two types (1) two-stage object detector such as fast R-CNN [14], faster R-CNN [15], Mask R-CNN [16]. and (2) single-stage object detector such as SSD [17], YOLO [18], YOLOv2 [19], YOLOv3[20]. The two-stage detection algorithm computationally very complex because they have separate backbone architecture. The single-stage object detector models are computationally less complex than that of the two-stage detector. We used the YOLOv3 object detection model in this paper. It is a fast and real-time object detection model. For the feature extraction, we used YOLOv3 darknet-53 as a backbone architecture. The first and second versions of YOLOv3 architecture struggle with small object recognition. As we detect fishes so this 53 convolutional layers' architecture for feature extractor is the best choice. The backbone architecture of YOLOv3 still performs better than ResNet-101 and ResNet-152.

Darknet-53 is the backbone of the YOLOv3 model that holds 23 remaining units, and every such unit restrains the 1×1 and 3×3 convolutional layers. At the end of every residual unit, an element-wise addition carried out between the input and output vectors. Every convolutional layer pursued by the Leaky ReLU activation function, where Batch Normalization being utilized. The downsampling is conducted with a stride of 2 at five separate convolutional layers. YOLOv3 adopts a Feature Pyramid Network (FPN) that used to detect the objects at

different scales that constructs FPN on top of backbone architecture and build a pyramid with downsampling strides, 8, 16, and 32 in order to detect all-sized objects. The Darknet-53 shows the output of the corresponding features in Figure.1 of different reduction dimension module.

	Layers	Filters	Stride	Output
	Convolutional	32	3×3	416×416
	Convolutional	64	3×3/2	208×208
1×	Convolutional	32	1×1	
	Convolutional	64	3×3	
	Residual			208×208
	Convolutional	128	3×3/2	104×104
2×	Convolutional	64	1×1	
	Convolutional	128	3×3	
	Residual			104×104
	Convolutional	256	3×3/2	52×52
8×	Convolutional	128	1×1	
	Convolutional	256	3×3	
	Residual			52×52
	Convolutional	512	3×3/2	26×26
8×	Convolutional	256	1×1	
	Convolutional	512	3×3	
	Residual			26×26
	Convolutional	1024	3×3/2	13×13
4×	Convolutional	512	1×1	
	Convolutional	1024	3×3	
	Residual			13×13

Fig. 1 Network Structure of YOLOv3



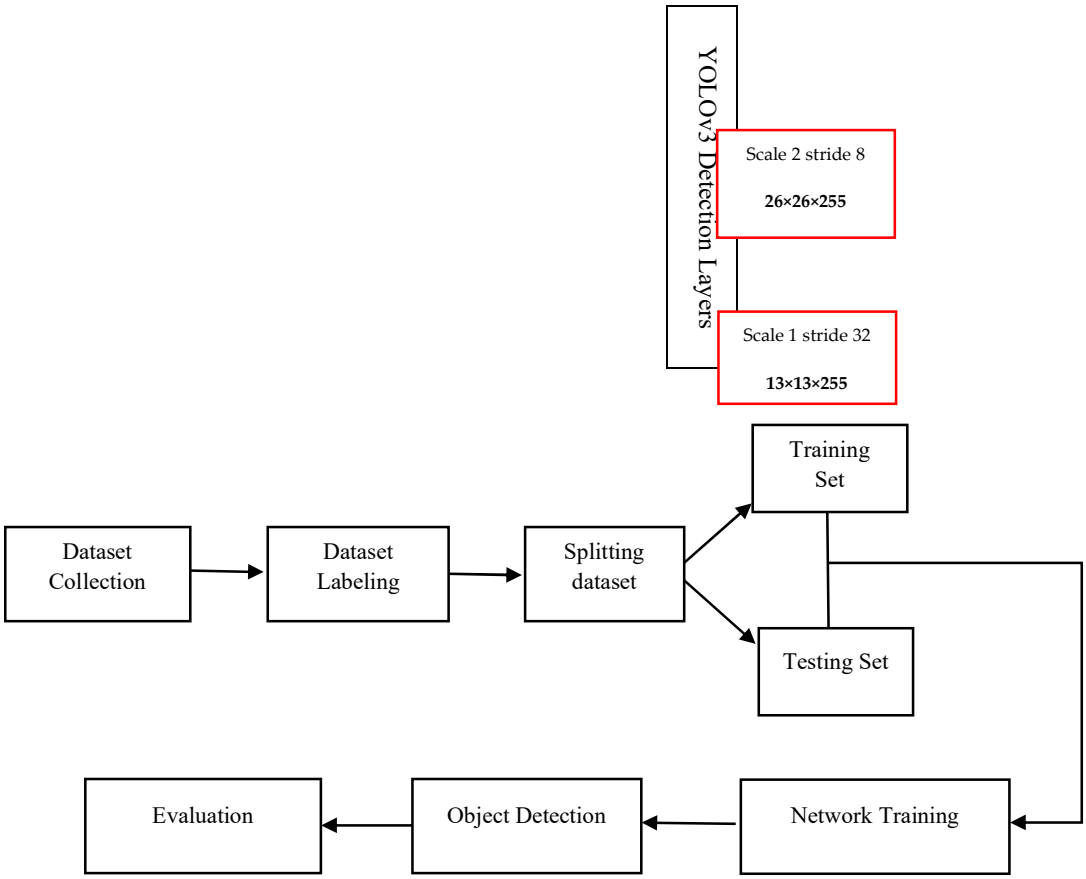


Fig 2. Dataset management and detection Flow Diagram

3.1 Improved Loss Function

In the YOLOv3, logistic regression is used to predict an objectness score for each bounding box that calculated the cost function. The objectness score is one if the anchor box overlaps the ground truth by more than or equal to a specific threshold value. On the other hand, we will ignore the prediction if it still overlaps ground truth by less threshold value that won't be considered the best bounding box. In the Equation (1), we can see that how the network output is changed by bounding box predictions where coordinates **tx, ty, tw, th** are responsible for computing the prediction

$$b_x = \sigma(t_x) + c_x$$
$$b_y = \sigma(t_y) + c_y \quad (1)$$
$$b_w = p_w e^{tw}$$
$$b_h = p_h e^{th}$$

The loss function is responsible for calculating the error between the real values and predicted one in the deep learning network. In the same way, the YOLOv3 loss function is the total sum of coordinate loss, class loss and confidence loss.

$$Loss_{coord} = \lambda_{coord} \sum_{i=0}^{s^2} \sum_{j=0}^{B} 1_{ij}^{obj} \left[(x_i - \hat{x}_i)^2 + (y_i - \hat{y}_i)^2 \right]$$

$$+\lambda_{coord} \sum_{i=0}^{s^2} \sum_{j=0}^{B} 1_{ij}^{obj} \left[\left(\frac{w_i - \hat{w}_i}{\hat{w}_i} \right)^2 + \left(\frac{h_i - \hat{h}_i}{\hat{h}_i} \right)^2 \right], \quad (2)$$

$$Conf_{loss} = \sum_{i=0}^{S^2} \sum_{j=0}^{B} 1_{ij}^{obj} \left(C_i - \hat{C}_i \right)^2 + \lambda_{noobj} \sum_{i=0}^{S^2} \sum_{j=0}^{B} 1_{ij}^{noobj} \left(C_i - \hat{C}_i \right)^2, \quad (3)$$

$$Class_{loss} = \sum_{i=0}^{S^2} 1_i^{obj} \sum_{c \in classes} (p_i(c) - \hat{p}_i(c))^2, \quad (4)$$

Above loss function, x_i, y_i are the center coordinates of the ith box grid cell. h_i, w_i are the height and width and height of the i-th grid cell, respectively. x_i, y_i, w_i and h_i are the real value and $\hat{x_i}$, $\hat{y_i}$, $\hat{w_i}$ and $\hat{h_i}$ are the predicted values. ($p_i(c)$ is the probability of a class and ($p_i\widehat{(c)}$ is the corresponding prediction value. λ_{coord} coordinate loss of weights and λ_{noobj} is the bounding box object loss without weights. 1_{ij}^{obj} denotes that the j-th box predictor in cell i is "responsible" for that prediction. We used $\left(\frac{w_i - \hat{w_i}}{\hat{w_i}}\right)^2$ and $\left(\frac{h_i - \hat{h_i}}{\hat{h_i}}\right)^2$ which helps to reduce the effect of different sizes of an object of the same kind. S^2 Denotes the grid cell B denotes the bounding boxes and 1_i^{obj} denotes the object existence in cell i or not.

Class	***Training images***	***Testing Images***	***Total Images***
Anemone Fish	950	200	1150
Jelly Fish	1005	200	1205
Star Fish	1100	200	1300
Shark	950	200	1150

3.2 Transfer Learning

A transfer learning method was developed to attain better performance with more transferred feature layers. Transfer learning is being used to extract the features from a custom dataset automatically with the help of using pre-trained models. It is a suitable way to apply transfer learning without considering substantial datasets, training, and calculation, which only consumed the time. Transfer learning is an adequate method if one has a small-scale sample dataset. Transfer learning used pre-trained convolutional neural network architecture where almost 1.2 million samples of ImageNet dataset and 1000 classes have been trained with powerful features extraction potential.

We proposed and trained darknet-53 backbone architecture which is pre-trained on ImageNet dataset to extract the features. Then we performed target detection on the COCO dataset by fine tuning. During the fine tuning we adjust several parameters, including the multi scale size of input images learning rate, batch size to boost and enhance the accuracy and performance.

4. Dataset Composition

The dataset is a key for object detection, and the collection of the dataset is an important, challenging milestone for object recognition. We used four kinds of fish including anemone fish, jellyfish, starfish, and shark. The samples of dataset are collected from various resources. All the samples of dataset have varying size such as 320×320, 416×416 and 480×480 The sample of the collected dataset is shown in table 1.

Table 1: The number of Training and testing Dataset of Fish Species

Dataset annotation is a very time-consuming process that takes much time than usual. As we know that the fish postures slightly and haphazardly change due to their free and multiple dimensional rotations, so the bounding box labeling inserts with much care and accurate for (mAP). Fish move freely, so we need to insert bounding box labeling in each direction for precise detection. We use a labelimg tool for dataset labeling.

5. Results and Comparison

The experiment performed by the deep learning open-source library TensorFlow 1.11, OpenCV 4.1.1, and coding is done with the high-level language python 3.5 at Ubuntu 18.04 operating system. Training and testing performed on the system intel core i-7-7700, GPU GTX 1080 with 11 GB of memory. We used the MS-COCO dataset for restoring and initialization of darknet-53 backbone architecture for Fish detection tasks to initialize our backbone Darknet-53 network. We set the resolution of the multi-scale images 544×544, 576×576, and 608×608 during training the model. At the training stage, the initial and end learning rate is set to 1e-4 and 1e-6, respectively, IOU threshold value 0.5, average decay 0.9, and the batch size is 4. And 6 We trained our model to 100 epochs, the batch size change after every 10 epoch. To prevent the model from non-convergence, the learning rate during the training process changed gradually. In the experiment, the custom fish detection dataset is used. The fish detection dataset consists of 4 classes, such as anemone fish, starfish, jellyfish, and shark. The total number of training images is 4005, and images for testing are 800. The mAP of the proposed model increased, with improved loss function and transfer learning technique of YOLOv3 by 3.74% compared to that of baseline YOLOv3, and the detection speed is 39 FPS, which enables real-time detection of YOLOv3. Some state-of-the-art architectures and detectors were chosen for comparisons such as Faster RCNN, YOLO, and YOLOv2 with our improved YOLOv3 model. The mAP with input image sizes of these different network structures is shown in Table 2. The comparison results between YOLOv3 and improved

YOLOv3 are shown in figure 3, figure 4, and figure 5. We draw the curves of model learning loss, confidence loss, and probability loss in Figures 6, 7, 8.

Fig. 3 (a) anemone fish result of original YOLOv3

Fig. 3 (b) Anemone fish result of Improved YOLOv3

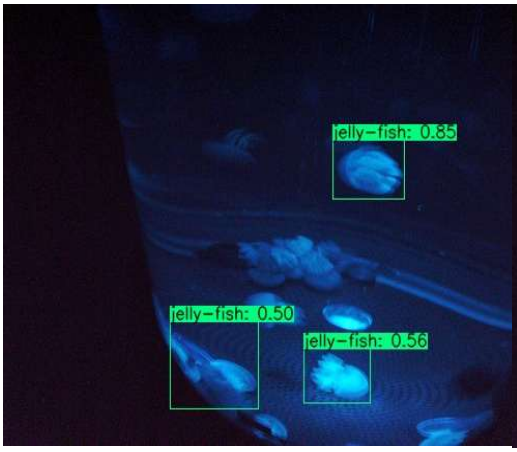


Fig. 4 (a) Jelly fish result of original YOLOv3

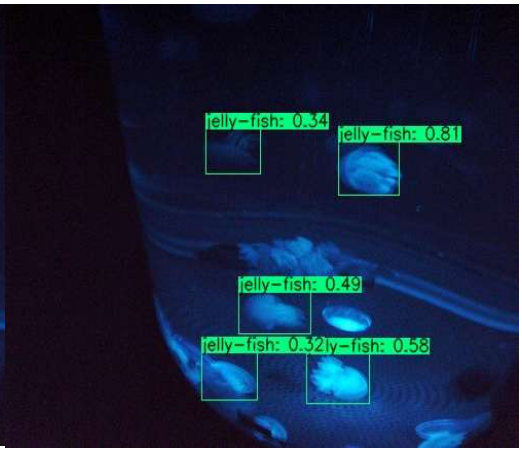


Fig. 4 (b) Jelly fish result of improvedYOLOv

Fig. 5 (a) star-fish result of original YOLOv3

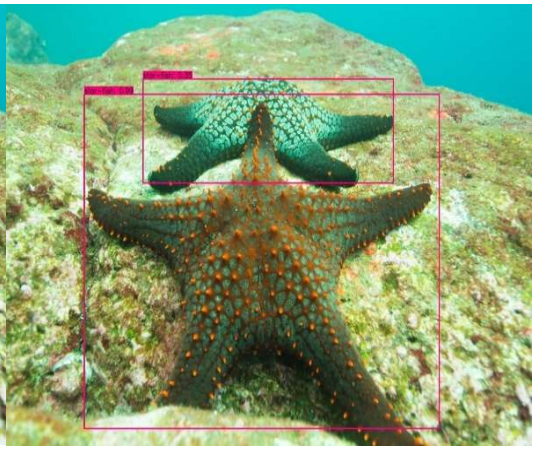
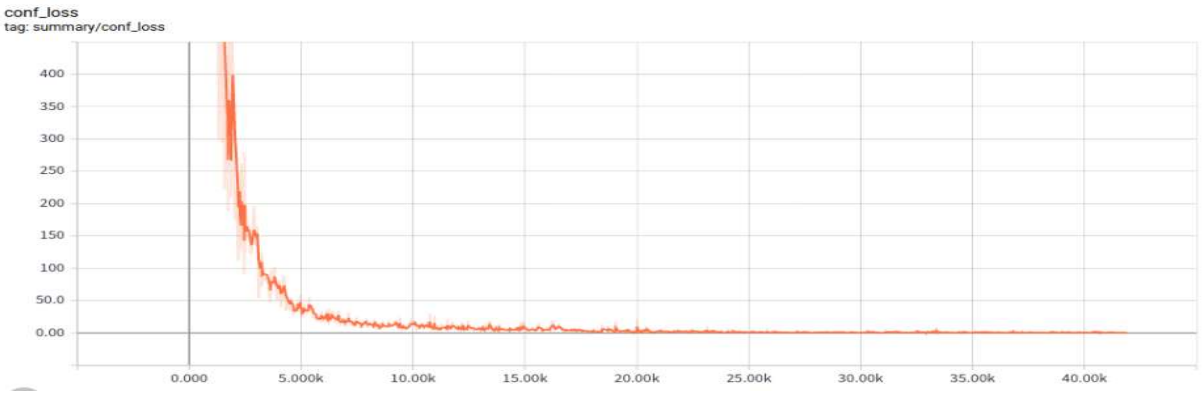


Fig. 5 (b) star-fish result of improved YOLOv3

Fig. 3, 4, 5. Comparison of detection results between the baseline YOLOv3 and improved YOLOv3. In the first column, Figures 3, 4, 5 (a) represents the original YOLOv3 detection results in the second column figures 3, 4, 5 (b) represents the improved YOLOv3 detection results.



Confidence loss cure of trained model

Figure. 6 Confidence loss cure of trained model (a)

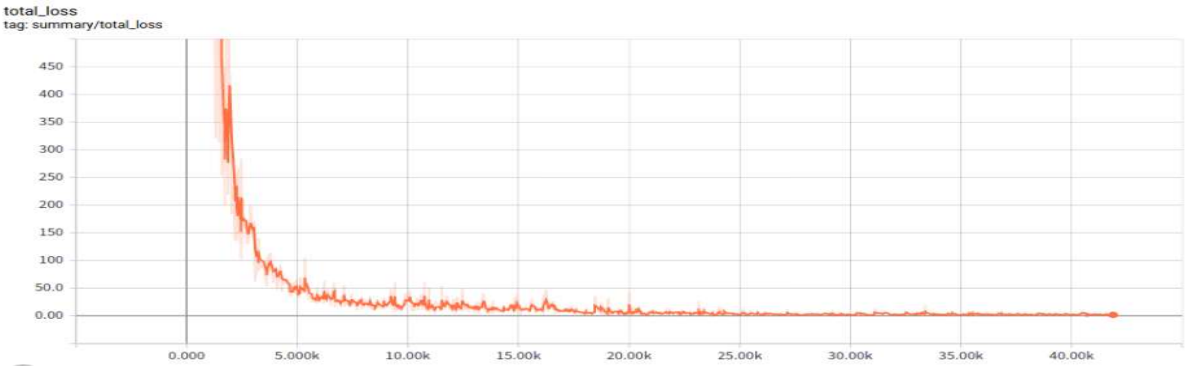


Figure. 7 Confidence loss cure of trained model (b)

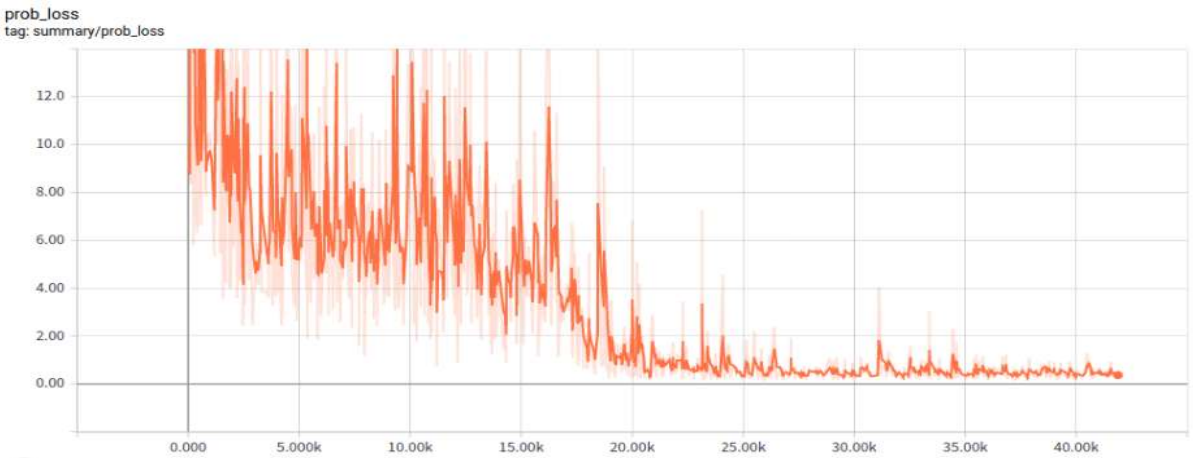


Figure. 8 Total loss cure of trained model

Table 2. YOLOv3 Comparison with others object detector models

Method	*Faster R-CNN*	*YOLO*	*YOLOv2*	*YOLOv3*	*YOLOv3 Improved (our)*
Input Image Size	480	448	416	608	608
mAP	77.4%	69.2%	78.8%	87.56%	91.30%

6. Contribution

We improved the model by using different types of techniques to enhance the accuracy of object detection. (1) We applied a novel transfer learning method to improve the efficiency and (2) improve the loss function for learning and convergence in the model.

7. Conclusion

In this paper, we improved YOLOv3 for fish detection. To obtain better results, we apply the technique of transfer learning and improved loss function as well. The improved YOLOv3 demonstrate that it outperforms than that of the baseline YOLOv3 model by improving the mAP of 3.40%. Mainly, we introduced how deep learning could be beneficial for the underwater species analysis at a large-scale dataset. The object detection shows that deep learning can be achieved on revolutionary results for fish recognition. To wrapping up, deep learning might not be an eventual solution to computer vision techniques, but it could be a realistic solution for the significant dataset in the marine ecosystem.

References

[1] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. “Deep learning.” In: Nature 521.7553 (2015), pages 436–444 (cited on pages 21, 126, 128).

[2] Szegedy, C., Toshev, A., & Erhan, D. (2013). Deep neural networks for object detection. In Advances in neural information processing systems (pp. 2553-2561).

[3] Huang, Phoenix X., Bastiaan J. Boom, and Robert B. Fisher. "Hierarchical classification for live fish recognition." In BMVC student workshop paper. 2012.

[4] Bermejo S. (2007). “Fish age classification based on length, weight, sex, and otolith morphological features.”Fish. Res. 84.

[5] R. Larsen, H. Olafsdottir, B.K. Ersbøll, Shape and texture based classification of fish species, Image Anal., 2009, 745-749.

[6] Helge Balk, Development of hydro acoustic methods for fish detection in shallow water, 2001, pg 28.

[7] Fuming Xiang, Application of Deep Learning to Fish Recognition, 2018, pg 53.

[8] River catch of salmon, sea trout and migratory char”, 2019, Available: https://www.ssb.no/en/elvefiske. Accessed on: Dec. 10, 2019.

[9] D.H Hubel and T.N Wiesel. Receptive Fields, Binocular Interaction, and Functional Architecture in the Cat's Visual Cortex. Pages 151-152, 1961.

[10] Adrian Reithaug, Employing Deep Learning for Fish Recognition, 2018, pg 85

[11] Ekaterina Lantsova, Automatic Recognition of Fish from Video Sequence, 2015, pg 49.

[12] S.O. Ogunlana, O. Olabode , S.A. A. Oluwadare & G. B. Iwasokun, Fish Classification Using Support Vector Machine, 2015, pg 75.

[13] Dhruv Rathi, Sushant Jain, Dr. S. Indu, Underwater Fish Species Classification using Convolutional Neural Network and Deep Learning,

[14] Girshick, R. (2015). Fast r-cnn. In Proceedings of the IEEE international conference on computer vision (pp. 1440-1448).

[15] Ren, S., He, K., Girshick, R., & Sun, J. (2015). Faster r-cnn: Towards real-time object detection with region proposal networks. In Advances in neural information processing systems (pp. 91-99).

[16]He, K., Gkioxari, G., Dollár, P., & Girshick, R. (2017). Mask r-cnn. In Proceedings of the IEEE international conference on computer vision (pp. 2961-2969).

[17] Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C. Y., & Berg, A. C. (2016, October). Ssd: Single shot multibox detector. In European conference on computer vision (pp. 21-37). Springer, Cham.

[18]Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You only look once: Unified, real-time object detection. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 779-788).

[19]. Redmon, J., & Farhadi, A. (2017). YOLO9000: better, faster, stronger. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 7263-7271).

[20]. Redmon, J., & Farhadi, A. (2018). Yolov3: An incremental improvement. arXiv preprint arXiv:1804.02767.