



# Fast Accurate Fish Recognition with Deep Learning Based on a Domain-Specific Large-Scale Fish Dataset

Yuan Lin<sup>1</sup>, Zhaoqi Chu<sup>2</sup>, Jari Korhonen<sup>3</sup>, Jiayi Xu<sup>4</sup>, Xiangrong Liu<sup>4</sup>,  
Juan Liu<sup>2</sup>(✉), Min Liu<sup>5</sup>, Lvping Fang<sup>5</sup>, Weidi Yang<sup>5</sup>, Debasish Ghose<sup>1</sup>,  
and Junyong You<sup>6</sup>

<sup>1</sup> School of Economics, Innovation, and Technology, Kristiania University College,  
Oslo, Norway

{yuan.lin,debasish.ghose}@kristiania.no

<sup>2</sup> School of Aerospace Engineering, Xiamen University, Xiamen, China

23020220157220@stu.xmu.edu.cn, cecyliu@xmu.edu.cn

<sup>3</sup> School of Natural and Computing Sciences, University of Aberdeen, Aberdeen, UK

jari.korhonen@abdn.ac.uk

<sup>4</sup> School of Information Science and Technology, Xiamen University, Xiamen, China

xrliu@xmu.edu.cn

<sup>5</sup> School of Ocean and Earth, Xiamen University, Xiamen, China

{minliuxm,lpfang,wdwang}@xmu.edu.cn

<sup>6</sup> Norwegian Research Centre (NORCE), Bergen, Norway

juyo@norceresearch.no

**Abstract.** Fish species recognition is an integral part of sustainable marine biodiversity and aquaculture. The rapid emergence of deep learning methods has shown great potential on classification and recognition tasks when trained on a large scale dataset. Nevertheless, some practical challenges remain for automating the task, e.g., the lack of appropriate methods applied to a complicated fish habitat. In addition, most publicly accessible fish datasets have small-scale and low resolution, imbalanced data distributions, or limited labels and annotations, etc. In this work, we aim to overcome the aforementioned challenges. First, we construct the OceanFish database with higher image quality and resolution that covers a large scale and diversity of marine-domain fish species in East China sea. The current version covers 63,622 pictures of 136 fine-grained fish species. Accompanying the dataset, we propose a fish recognition testbed by incorporating two widely applied deep neural network based object detection models to exploit the facility of the enlarged dataset, which achieves a convincing performance in detection precision and speed. The scale and hierarchy of OceanFish can be further enlarged by enrolling new fish species and annotations. Interested readers may ask for access and re-use this benchmark datasets for their own classification tasks upon inquiries. We hope that the OceanFish database and the fish recognition testbed can serve as a generalized benchmark that motivates further development in related research communities.

**Keywords:** Fish recognition · Deep learning · Data augmentation · Convolutional neural networks

## 1 Introduction

The rapid emergence of computer vision and machine learning technologies is expected to offer new tools for exploring the ocean. Observation of the population and distribution of marine species are among the important tasks which give valuable insights to address marine biodiversity in terms of the variety, distinctiveness, and complexity of marine life [1]. Reliable algorithms that assist and automatize the recognition of marine species are in great demand. Fish species classification and recognition is a compelling research field with potential applications that cover a broad range of industries, including but not limited to fishery.

Traditional methods of identifying fish species are in general using shape and texture feature extraction [2–4]. However, the main drawbacks of feature based approach comes from its sensitivity to background noise, lack of generalization and difficulties of finding discriminating features, especially when the task deals with recognizing sub-ordinate object classes. In recent years, deep convolutional neural networks (CNN) have shown impressive results and large potential for identifying fish species. A deep-learning architecture that is composed of two principal component analysis (PCA) based convolutional layers, spatial pyramid pooling, and a linear SVM classifier was proposed in [5] to recognize fish from ocean surveillance videos, and it achieved 98.64% accuracy. In [6], AlexNet [7] was trained via transfer learning for automatic fish species classification from underwater video source, achieving an accuracy of 99.45%. Deep learning methods have been applied to fish recognition competition on the Kaggle challenge named “The Nature Conservancy Fisheries Monitoring” [8], whereas the deployed fish benchmarks are, however, small-scale and of poor image quality. There are some practical challenges remaining for identifying fish species. First, fish recognition is hindered by the poor image quality, uncontrolled objects and unconstrained natural habitat especially in wild circumstances. Second, the number of fish species in marine ecology is very large, and the distribution of fish species is highly characterized by regions or territories. Third, in real-world applications, data assessment, e.g. labeling and annotation, is expensive as it involves time-consuming and labour-intensive process.

The success of deep learning in classification and recognition tasks lies in the enormous explosion of data, as proven by the significant achievements leveraged by large-scale open datasets, such as ImageNet [9], COCO [10], and VOC [11] before being fine-tuned to new applications. However, most large-scale visual databases are not dedicated to fish species. For example, ImageNet provides a hierarchical framework that contains over 14 million hand-annotated images and 20 thousand ambiguous categories. However, it contains little fishery data, and the hierarchical architecture and label annotations do not align with our study. Similar limitation was addressed by Microsoft COCO, which shows the limitations of dataset in terms of volume and category. Other widely used datasets,

e.g. TinyImage [12], ESPdataset [13] either only provide low-quality fish subsets, or most of the dataset is not publicly accessible.

To the best of our knowledge, the scale, accuracy and diversity of most open fish image datasets is insufficient. One popular fish database is Fish4Knowledge [14], where the fish data is acquired from a live video dataset resulting in 27,370 verified fish images covering 23 representative species. The dataset has been used by the 2015 Sea Clef contest [15] for fish species classification [24]. However, the video quality ( $640 \times 480$  pixels) and the number of species (mainly coral reef fishes in Southern China Sea) remain quite limited. QUT fish dataset, mostly collected in a constrained and plain environment [16], consists of only 3,960 labelled images of 468 species. DeepFish dataset, proposed in [17], contains approximately 40,000 underwater images from 20 habitats in the marine-environments of tropical Australia. However, the main purpose of the dataset is to assess fish-habitat associations in challenging, even inaccessible environments, thus with limited generalization. Another open fishery images database lately proposed by [18] contains 86,029 images of 34 object classes, making it the largest and most diverse public dataset of fisheries EM imagery to-date, as claimed by the author. However, the resolution restriction of EM images can inhibit further comprehensive progress. Therefore, there is a need of constructing a reliable fish dataset of large scale, accuracy and diversity of species.

In this work, we introduce a novel domain-specific dataset, named OceanFish, with 63,622 high resolution ( $1088 \times 816$  pixels) PASCAL VOC formatted images [11] covering 136 categories mainly inhabiting in East China Sea. Interested readers may ask for access and re-use this benchmark datasets for their own classification tasks upon inquiries. We hope that the scale, accuracy and diversity of the fish dataset can offer unparalleled opportunities to researchers in a variety of research communities and beyond. Accompanying the dataset, we also implement a fine-tuned fish recognition model by incorporating two of the most widely used object recognition models, Faster R-CNN [21] and Single Shot MultiBox Detector (SSD) [23], and investigate their performance on OceanFish. As a result, we obtain a comprehensive testbed, which can be further tailored to include new algorithms for specialized problem setups.

In summary, the main contribution of this work lie in two folds: a large scale, high quality dataset OceanFish, and a deep learning based framework to recognize fish species. Specifically, the contribution of OceanFish is highlighted in the following aspects: the dataset is domain specific, large scale, the image quality is enhanced, and the application aspects clarified.

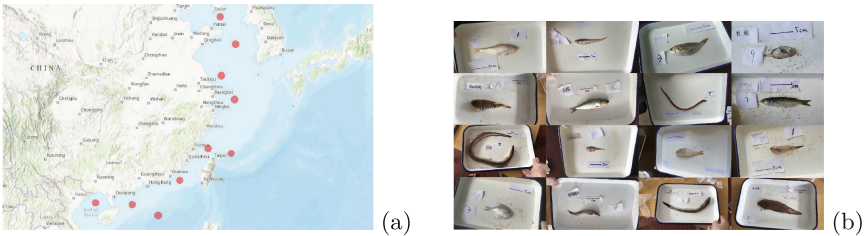
The remainder of the paper is structured as follows: We present OceanFish in Sect. 2, where data augmentation mechanisms are elaborated as well. In Sect. 3, we present our generalized fish recognition framework, and explain the experimental results. Finally, we conclude in Sect. 4.

## 2 OceanFish

### 2.1 Constructing the Dataset

One ambition of this work is to construct a high quality benchmark dataset that covers a large scale and diversity of marine-domain fish species, and its scale and hierarchy can be further enlarged by enrolling new fish species and annotations. We believe that the dataset will facilitate developing advanced, large-scale deep learning algorithms. We build our own customerized dataset OceanFish by shooting video clips for collected fish samples from different angels to create variation, with implementation details described as below:

1. Specify the photographing conditions and device for image acquisition. A light background which is contrast to the fish color is required to ensure that fish's outline is clearly visible. We use D65 light source with a color temperature of 6500K.
2. Collect fish characteristics by shooting from various angles. The video clip is taken from carefully chosen angles i.e., 30 or 45°C, with the camera rotating for about 60 s. Subsequent acquisition of images are extracted from the video frames.
3. Categorize and annotate images according to the chosen naming convention.
4. Split the dataset into training, test, and validation parts. 60% of the entire dataset is for training, 20% for testing, and 20% for validation.



**Fig. 1.** (a)Locations and territory where OceanFish categories are acquired (b)16 representative fish categories in OceanFish

Images for OceanFish dataset were collected for 136 fish species, mainly inhabiting in the East China Sea, and the total volume is 63,622 pictures of  $1088 \times 816$  pixels in PASCAL format. The locations from where the fish were acquired and 16 different fish species as examples are shown in Fig. 1, and the distribution of OceanFish images among categories is shown in Fig. 2.

A comparison between different fish datasets is shown in Table 1. Note that as well as Fish4Knowledge and Rockfish, the images of OceanFish are taken in a controlled environment which eliminate background noise, allowing a reliable and comprehensive analysis. Overall, the OceanFish dataset achieves an overall

upgrade in comparison to the previous fish datasets in terms of scale, accuracy, and diversity of species.

PASCAL VOC2007 was applied to create the dataset, and LabelImg software was used to manually label and annotate images, and to create XML files automatically. XML files contain recorded information about the fish sample locations and classes.

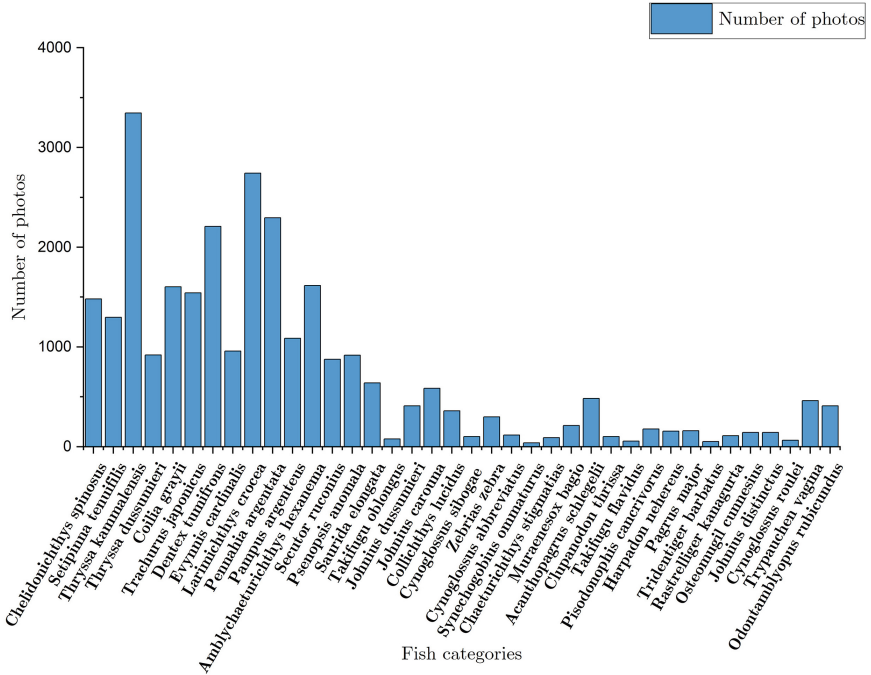


Fig. 2. OceanFish image distribution among different categories

## 2.2 Data Augmentation

Data augmentation, used to increase the number of training samples by applying a set of transformation to the images, has proven to improve the performance of the neural network models, especially when learning from small scale datasets. It can also be beneficial in yielding faster convergence, decreasing the risk of overfitting, and enhancing generalization [26]. In this work, most images are shot in the same setup environment. The similarity of image layout, background noise, spatial properties, texture, or overall shapes can lead to a classification model with a compromised generalization capability. To reduce the risk of overfitting, we applied label-preserving transformations to artificially enlarge the dataset.

**Table 1.** Comparison between fish dataset statistics

Dataset	No. of Images	Tasks	Resolution	Environment Type	Has background
OceanFish (ours)	63,622	Detection	1088 × 816	Controlled	No
QUT	3,960	Classification	480 × 360	Controlled	No
Fish4Knowledge	1.6B	Detection	352 × 240	On site	No
RockFish	4,307	Detection	1280 × 720	On site	Yes

In this work, we employed three data augmentation methods: color jitter, random cropping, and affine transformation. Color jitter randomly changes the brightness, contrast, and saturation of an image instead of the RGB channels. We adopt the color jitter mechanism employed by Facebook AI Research for the reimplementation of ResNet-101 [27]. Random cropping prevents a CNN from overfitting to specific features by changing the apparent features in an image. In this work, we applied Scale Jittering that is used by VGG and ResNet-101 networks [29]. Affine transformations generate duplicate images that are shifted, zoomed in/out, rotated, flipped, and distorted can be generated for image augmentation. We have applied a set of affine transformations for image augmentation, and both the original images and the duplicates were used as input to the neural networks.

Most existing learning algorithms produce inductive learning bias towards the frequent (majority) classes if training data are not balanced, resulting in poor minority class recognition performance. Data augmentation also aims at alleviating this problem and improving data generalizability. In this work, we chose a data equalization mechanism inspired by the implementation of AlexNet [7]. The procedure is described as follows:

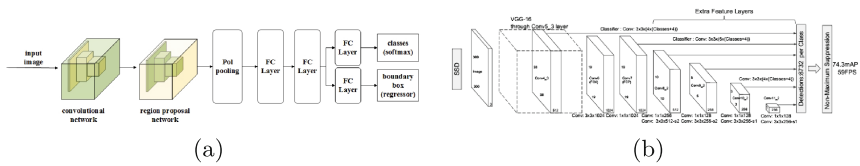
1. Get a list of the original images, sorted by label.
2. Calculate the number of samples in each category and save the largest number as  $M$ .
3. Generated a new list of size  $M$  for each category, and fill in the new list by looping the original content repeatedly, until the number of samples in the new list reaches  $M$ .
4. Random shuffle the new list with  $M$  samples.
5. To avoid overfitting, the aforementioned color jitter, scale jittering, and affine transformations are applied for image augmentation.

### 3 Methods and Experiments

#### 3.1 CNN Based Object Detection

Since the 2010s, there has been a trend towards utilizing and continuously improving CNNs in object detection tasks. CNN based object detection algorithms can be divided into two major categories: region proposal based approaches such as R-CNN [19], Fast R-CNN [20], and Faster R-CNN [21], which

yield high detection accuracy in terms of mean Average Precision (mAP) in different object detection scenarios; and regression based end-to-end approaches such as You Only Look Once (YOLO) [22] and Single Shot MultiBox Detector (SSD) [23], which predict bounding box coordinates and class probabilities straight from image pixels, thus achieving fast detection at the cost of decreased accuracy. Faster R-CNN and SSD are applied in this work, and their concepts are depicted in Fig. 3. Note that we have focused on the original meta-architecture for clarity and simplicity, although many improvements have been introduced to the original architectures since their release.



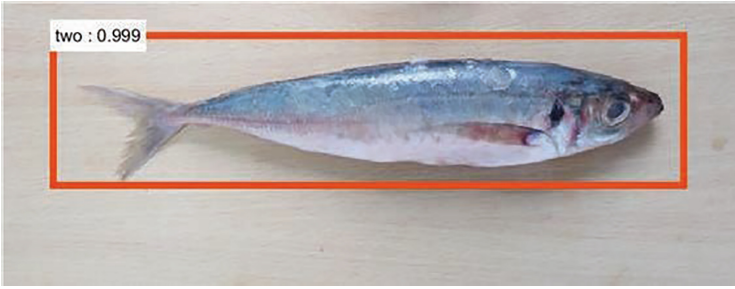
**Fig. 3.** Two representative CNN based object detection framework (a) Faster R-CNN (b) Single Shot MultiBox Detector (SSD)

### 3.2 Experiments and Performance Evaluation

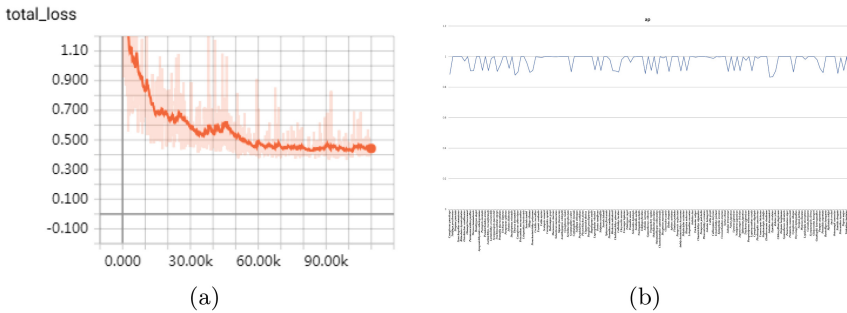
In this section, we carry out a series of experiments to evaluate the performance of Faster R-CNN and SSD on OceanFish. Both models contain a deep learning backbone network as feature extraction module. We perform fish recognition on OceanFish by fine-tuning the pre-trained model. TensorFlow is used for all the experiments. Models are trained on NVIDIA GeForce GTX 1080Ti 11G GPUs with a total batch size of four. Dataset is divided into training, validation, and test sets in a proportion 3 : 1 : 1. The learning rate is initialized to  $5 \cdot 10^{-4}$ . Figure 4 shows a successful detection of *Decapterus maruadsi*, with both classification and localization information.

First, we choose ResNet-101, ResNet-50 [27], and VGG-16 [29] as backbone feature extractors for Faster R-CNN and evaluate their performance on a subset of OceanFish with 20 fish species and 400 images in each category. The experimental results show that ResNet-101 achieves the highest accuracy with the mAP of 0.994, compared to ResNet-50 with the mAP of 0.988, and VGG-16 with the mAP of 0.986. Therefore, we select ResNet-101 backbone as the feature extractor for Faster R-CNN and it applies to the following experiments.

In Fig. 5(a), we depict the total loss in respect with training iterations for Faster R-CNN. The total loss is gradually decreasing and converging as the number of iterations increases. The Ap value distribution of a subset of fish species is shown in Fig. 5(b), where the average mAP for the complete dataset is approximately 0.9712.



**Fig. 4.** A successful detection of fish specie *Decapterus maruadsi*



**Fig. 5.** Faster R-CNN: (a) Total loss v.s. Number of training iterations, (b) Ap value distribution of a subset of fish species

The results in Table 2 show that Faster R-CNN has relatively low miss rate for all categories, which means that the framework in general detects and localizes the target properly. We apply both false positive rate (FPR) and true positive rate (TPR) in the confusion matrix to evaluate the classification performance. Relatively large false positive error rate is observed for several fish species, indicating that the proportion of misclassified items in the whole set cannot be ignored. For example, among the 669 images of *Thryssa kammalensis* in OceanFish, 38 images are falsely detected as *Thryssa dussumieri*, five images as *Setipinna tenuifilis*, and two images as *Pampus argenteus*.

In the second part of the experiments, we apply VGG-16 as SSD backbone network. Figure 6(a) shows that the total loss is gradually decreasing and converging as the number of iterations increases. After 70,000 iterations, the SSD model yields the recognition accuracy with mAP of 0.8156 over the complete OceanFish dataset, which indicates a slight performance degradation in comparison with Faster R-CNN. Moreover, the Ap value of fish species demonstrates more variance among categories, as shown in Fig. 6(b).

As shown in Table 3, SSD and Faster R-CNN achieve satisfactory localization accuracy, since both detection algorithms have utilized bounding box regression for fine-tuning of object locations. It has also been observed that SSD and Faster

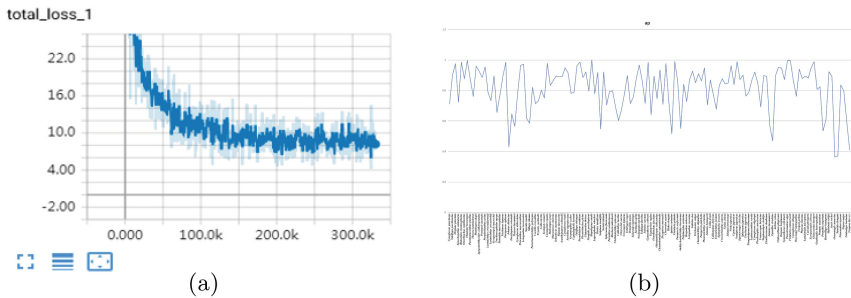


**Table 2.** Fish species with top 10 highest error rate classified by Faster R-CNN

Category	Name	No. of Images	Miss Rate	False Positive
19	Johnius carouna	313	0	16.6%
3	Thryssa kammalensis	669	0.89%	7.3%
120	Johnius trewavasae	269	0.37%	15.2%
10	Pennahia argentatus	460	1.3%	8.9%
9	Larimichthys crocea	549	1.4%	4.9%
61	Saurida tumbil	116	0%	18.1%
20	Collichthys lucidus	181	0.55%	7.7%
16	Saurida elongata	223	0.9%	4.9%
24	Cynoglossus abbreviatus	238	1.3%	4.6%
22	Cynoglossus sibogae	148	1.4%	7.4%

**Table 3.** Fish species with top 10 highest error rate classified by SSD

Category	Name	Total Images	Miss Rate	False Positive
19	Thryssa kammalensis	669	0.9%	33%
3	Trachinocephalus myops	389	0.77%	49.9%
120	Pennahia argentatus	460	1.3%	39.6%
10	Odontamblyopus lacepedii	433	0.46%	31.2%
9	Saurida elongata	223	2.2%	130%
61	Johnius carouna	313	0%	58.2%
20	Lagocephalus wheeleri	406	0.49%	28.8%
16	Chelidonichthys kumu	297	%	37.4%
24	Upeneus bensasi	431	0.46%	23.2%
22	Decapterus maruadsi	257	0%	36.6%

**Fig. 6.** SSD: (a) Total loss v.s. Number of training iterations (b) Ap value distribution among selected fish species

R-CNN present the accuracy/speed tradeoff due to the essential architecture difference. Although SSD shows faster speed in comparison with Faster R-CNN, it introduces a certain performance degradation. For example, among the total 669 images of *Thryssa kammalensis* in OceanFish, 221 images were falsely detected as other fish species, yielding false positive error rate exceeding 30%.

Table 4 shows the performance of our testbed in terms of mAP value. Note that Faster R-CNN and SSD has achieved mAP of 97.12% and 81.56% respectively when applied to OceanFish dataset. It can be seen that the overall detection performance improves significantly, noting that previous models, i.e., AlexNet and CIFAR-10 [28] only accomplish mAP of 60.90% and 71.10% for classifying the fish subset of ImageNet. The result shows that our proposed testbed together with fine-grained dataset yield satisfactory results, matching our observed trends that large-scale fine-grained datasets play a vital role in deep learning tasks.

**Table 4.** Performance of our testbed in terms of mAP value based on OceanFish

Model	Dataset	Pre-processing	Backbone network	Precision rate
Faster R-CNN (ours)	63622 images 136 categories	Data augmentation	ResNet-101	97.12%
SSD (ours)	63622 images 136 categories	Data augmentation	VGG-16	81.56%

## 4 Conclusion

In this work, we introduced a novel domain-specific dataset named OceanFish and tailored to marine domain fish classification and recognition tasks. The dataset consists of 63,622 high resolution images covering 136 clusters, and it is therefore much larger than the earlier publicly accessible fish datasets. The dataset can be further enlarged by enrolling new fish species and annotations. We hope that the OceanFish database can serve as a benchmark that motivates further development in related research communities.

We implemented a deep learning based fish recognition testbed by incorporating Faster R-CNN and SSD, and investigated their performance on OceanFish dataset comprehensively. We demonstrated through extensive experiments that the model achieves satisfactory classification accuracy and speed. It can be further tailored and used to create specialized algorithms in different problem setups, and thus it has provided us with potential of applying the generalized approach together with domain specific large scale dataset to smart farming and aquaculture production in terms of productivity, food security, and sustainability.

Experimental results show that our model may introduce false detections in some fish categories, and the Ap values demonstrate certain variance among categories. The reason behind might lie in the model architecture and inadequate examples of some categories. To improve the performance further, we may improve model architecture, or refine training procedure, including changes

in loss functions, data preprocessing, and optimization methods etc., in our future work. Last while not least, we encourage researchers to enlarge OceanFish dataset, in both scale and hierarchy, to provide upgraded capability.

## References

1. Gouletque, P., et al.: The importance of marine biodiversity. *Biodiversity in the Marine Environment*, pp 1–13 (2014)
2. Yi-Haur, S., et al.: Fish observation, detection, recognition and verification in the real world. In: *Proceedings of the International Conference on Image Processing, Computer Vision, and Pattern Recognition (IPCV)*, p. 1, (2012)
3. Katy, B., et al.: Fish species recognition from video using SVM classifier. In: *Proceedings of the 3rd ACM International Workshop on Multimedia Analysis for Ecological Data*, pp. 1–6 (2014)
4. Mehdi, R., et al.: Automated fish detection in underwater images using shape based level sets. *Photogram. Record.* **30**(149), 46–62 (2015)
5. Qin, H.W., et al.: DeepFish: accurate underwater live fish recognition with a deep architecture. *Neurocomputing* **187**, 49–58 (2016)
6. Tamou, A.B., et al.: Underwater live fish recognition by deep learning. In: *International Conference on Image and Signal Processing*, pp. 275–283 (2018)
7. Krizhevsky, A., et al.: ImageNet classification with deep convolutional neural networks. In: *Proceedings of the 25th International Conference on Neural Information Processing Systems*, vol. 1, pp. 1097–1105 (2012)
8. <https://www.kaggle.com/c/the-nature-conservancy-fisheries-monitoring> . Kaggle Competition. The Nature Conservancy Fisheries Monitoring (2017)
9. Deng, J., et al.: ImageNet: a large-scale hierarchical image database. In: *IEEE Conference on Computer Vision and Pattern Recognition*, 2009. CVPR (2009)
10. Lin, T.-Y., et al.: Microsoft COCO: common objects in context. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) *ECCV 2014*. LNCS, vol. 8693, pp. 740–755. Springer, Cham (2014). [https://doi.org/10.1007/978-3-319-10602-1\\_48](https://doi.org/10.1007/978-3-319-10602-1_48)
11. Everingham, M., et al.: 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>
12. Torralba, A., et al.: 80 million tiny images: a large dataset for nonparametric object and scene recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **30**(11), 1958–1970 (2008)
13. Ahn, L.V., et al.: Labeling images with a computer game. In: *CHI04 Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pp. 319–326 (2004)
14. Fisher, R. et al.: Overview of the Fish4Knowledge project. In: *Fish4Knowledge: Collecting and Analyzing Massive Coral Reef Fish Video Data*, pp. 1–17 (2016)
15. Alexis, J., et al.: Life CLEF 2015: multimedia life species identification challenges. In: *Experimental IR Meets Multilinguality, Multimodality, and Interaction*. Springer International Publishing, pp. 462–483 (2015)
16. Anantharajah, K., et al.: Local inter-session variability modelling for object classification. In: *IEEE Winter Conference on Applications of Computer Vision*, pp 309–316 (2014)
17. Saleh, A., et al.: A realistic fish-habitat dataset to evaluate algorithms for underwater visual analysis. *Sci. Rep.* **10**, 14671 (2020)

18. J. Key, et al.: The fishnet open images database: a dataset for fish detection and fine-grained categorization in fisheries. In: 8th Workshop on Fine-Grained Visual Categorization at CVPR (2021)
19. Girshick, R., et al.: Feature hierarchies for accurate object detection and semantic segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, vol. 2014, pp. 580–587 (2014)
20. Girshick, R.: Fast R-CNN. In: Proceedings of the IEEE International Conference on Computer Vision, vol. 2015, pp. 1440–1448 (2015)
21. Ren, S.Q., et al.: Faster R-CNN: towards real-time object detection with region proposal networks. In: Advances in Neural Information Processing Systems, vol. 2015, pp. 91–99 (2015)
22. Redmon, J., et al.: You only look once: unified, real-time object detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 779–788 (2016)
23. 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](https://doi.org/10.1007/978-3-319-46448-0_2)
24. Villon, S., et al.: Coral reef fish detection and recognition in underwater videos by supervised machine learning: comparison between deep learning and HOG + SVM methods. In: International Conference on Advanced Concepts for Intelligent Vision Systems, ACIVS, pp. 160–171 (2016)
25. Deng, J., et al.: <http://www.image-net.org/challenges/LSVRC/2012/>. In: ILSVRC-2012 (2012)
26. Takahashi, R., Matsubara, T.: Data augmentation using random image cropping and patching for deep CNNs. In: arXiv (2018)
27. He, K., et al.: Deep residual learning for image recognition. In: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR2016), pp. 770–778 (2016)
28. Krizhevsky, A.: Learning multiple layers of features from tiny images. Technical report, University of Toronto, pp. 1–60 (2009)
29. Simonyan, K., et al.: Very deep convolutional networks for large-scale image recognition. In: [arXiv:1409.1556](https://arxiv.org/abs/1409.1556) (2014)